

IT UNIVERSITY OF COPENHAGEN

PHD THESIS

---

# Procedural Personas for Player Decision Modeling and Procedural Content Generation

---

*Author:*

Christoffer Holmgård

*Supervisors:*

Prof. Georgios N. Yannakakis

Prof. Julian Togelius

*A thesis submitted in partial fulfillment of the requirements  
for the degree of Ph.D.*

*in the*

Center for Computer Games Research  
IT University of Copenhagen

August 2015

*“Games are a series of interesting decisions.”*

Sid Meier

*“This is the essence of intuitive heuristics: when faced with a difficult question, we often answer an easier one instead, usually without noticing the substitution.”*

Daniel Kahneman

IT UNIVERSITY OF COPENHAGEN

## *Abstract*

### **Procedural Personas for Player Decision Modeling and Procedural Content Generation**

by Christoffer Holmgård

How can player models and artificially intelligent (AI) agents be useful in early-stage iterative game and simulation design? One answer may be as ways of generating synthetic play-test data, before a game or level has ever seen a player, or when the sampled amount of play test data is very low.

This thesis explores methods for creating low-complexity, easily interpretable, generative AI agents for use in game and simulation design. Based on insights from decision theory and behavioral economics, the thesis investigates how player decision making styles may be defined, operationalised, and measured in specific games.

It further explores how simple utility functions, easily defined and changed by game designers, can be used to construct agents expressing a variety of decision making styles within a game, using a variety of contemporary AI approaches, naming the resulting agents “Procedural Personas.”

These methods for constructing procedural personas are then integrated with existing procedural content generation systems, acting as critics that shape the output of these systems, optimizing generated content for different personas and by extension, different kinds of players and their decision making styles.

Finally, the thesis compares the top-down theory driven definition of utility functions with bottom-up play trace driven learning of utility functions and proposes methods for hybridizing the two approaches.

## *Abstract*

### **Procedural Personas for Player Decision Modeling and Procedural Content Generation**

by Christoffer Holmgård

Hvordan kan modeller af spilleradfærd og agenter baseret på kunstig intelligens bruges i de tidligere stadier af iterative spiludviklingsprocesser? Et muligt svar er, at de kan bruges til at generere data fra syntetiske spillere, før et spil eller en bane er afprøvet af menneskelige spillere, eller når mængden af indsamlet information er forholdsvis lille.

Denne afhandling udforsker metoder til at skabe kunstigt intelligente agenter, der har en lav kompleksitet og hvis adfærd er let at fortolke, til brug i design af spil og simulationer. Funderet i beslutningstagningsteori undersøger afhandlingen, hvordan spilleres beslutningstagningsstilarter kan defineres, operationaliseres og måles i specifikke spil.

Desuden undersøger afhandlingen, hvordan simple funktioner, der beskriver nytteværdi, som er lette for spildesignere at definere og ændre, kan bruges til at styre agenter, som udtrykker en række forskellige beslutningstagningsstilarter i en række spil. Agenterne skabes ved hjælp af moderne metoder inden for kunstig intelligens til computerspil og kaldes under en samlende betegnelse for "Procedurale Personaer."

Metoder til at skabe procedurale personaer integreres med eksisterende systemer til procedural generering af spilindhold, hvor personaerne fungerer som kritikere, der former systemernes output, optimerer genereret indhold til forskellige personaer og i forlængelse heraf forskellige typer af spillere og deres beslutningstagningsstilarter.

Til sidst sammenligner afhandlingen to tilgange til at definere modeller af spilleradfærd og agenter: Én baseret på teori-dreven definering af nytteværdifunktioner og én baseret på data-dreven maskinlæring af nytteværdifunktioner. Afhandlingen afsluttes med et forslag til, hvordan de to tilgange kan kombineres i en hybridtilgang til modellering af spilleres beslutningstagningsstilarter.



# *Acknowledgements*

Writing this thesis was a large project that I could not have accomplished on my own. I am indebted to many people whom I try to thank and remember below. Great thanks go to Georgios N. Yannakakis and Julian Togelius for bringing me into the academic community. The thesis and my work is shaped by their guidance and ideals and would never have happened without their help and trust in my abilities and potential. I thank Antonios Liapis who has been an inspiring, creative, and meticulous collaborator, and a good friend, throughout my PhD studies. The procedural persona concept is strongly shaped by our discussions. I also thank Rilla Khaled who encouraged me to apply for the research assistant position that led up to me doing a PhD. My thanks also go to Héctor P. Martínez for his help in relation to stress detection and signal treatment, as well as all the other driving forces in the project related to the StartleMart studies, especially Karen-Inge Karstoft, Henrik Steen Andersen, Lars Henriksen, and Agne Gediminskaite. For developing the framework which this thesis partially rests on thanks go to Alessandro Canossa who also was kind enough to host me at Northeastern University along with the rest of the Playable Innovative Technologies Lab, including Casper Harteveltdt and Steven Sutherland whom I greatly enjoyed working with. Also many thanks to my friends and colleagues at the University of Malta, Institute of Digital Games, for hosting Benedikte and myself ensuring a great stay: Ashley Davis, Costantino Oliva, Gordon Calleja, Pippin Barr, Ken Hullett, Phil Lopes, and Mirjam Eladhari. Likewise, thanks go out to the whole Game Innovation Lab at New York University Polytechnic School of Engineering for hosting me during my visit. I also thank the rest of my research group at the IT University, whom I had great fun with and learned so much from. Many thanks go out to (in no particular order) Corrado Grappiolo, Miguel Sicart, Rune Lundedal Kristensen, Sebastian Risi, Dan Lessin, Espen Aarseth, Noor Shaker, Martin Pichlmair, Daniel Vella, Daniel Cermak-Sassenrath, Sebastian Möring, Hans-Joachim Backe, Henrike Lode, Gabriella Barros, Marco Scirea, Florian Berger, Yun-Gyung Cheong, Byung Chull Bae, Emma Witkowski, Sheng-Yi Hsu, T.L. Taylor, Tobias Mahlmann, Paolo Burelli, and Mark Nelson. Every one of you impacted my thinking and helped me have an amazing time at the IT University. Very special thanks to my friends and colleagues Julie Houlberg Michaelsen, Morten Mygind, Nils Deneken, Nicklas Nygren, and Douglas Wilson who not only were great and supportive friends, but also kept me grounded in and connected to the practice of game design and development while I was doing my thesis. My gratitude also goes to the Stibo Foundation which funded my stay abroad under their Global IT Talents program, allowing me to visit not one, but two, internationally outstanding research groups within games and technology, the European Union's FP7 program which funded parts of my PhD work, and the Danish Council for Technology and Innovation which also funded parts of my research. I also thank the IT University's PhD-school for taking me on as a PhD-student and offering stellar conditions and support. Mostly importantly, my gratitude and warmest thanks go to Benedikte Mikkelsen without whom the unlikely series of events that led to me writing this thesis would never have occurred and without whose encouragement I would not have pursued nor completed it. No doubt I am guilty of omitting someone who should have been mentioned in the list above for which I offer my apologies. In many ways our lives and work become the sum and product of those around us, so thank you everyone who helped make mine what they are.

# Preface

The research reported in this thesis was conducted from 2012 to 2015 at the IT University of Copenhagen, University of Malta, Northeastern University, and New York University, as part of my enrollment in the three-year PhD program at the IT University of Copenhagen's PhD School.

The PhD training and research that I conducted was continuously submitted to academic venues for review and, in some instances, published throughout the course of my enrollment. For that reason, this thesis takes the form of a collection of papers.

However, not all the publications that were published during my enrollment fit neatly together under the same main topic. For the sake of focus and brevity, this thesis focuses on the development and application of the concept of *procedural personas* for player modeling and automatic play-testing. The main text contains a summary and integration of the papers that were published under this theme.

In parallel with pursuing and developing the concept of procedural personas, I also conducted work in another direction within player modeling: Detecting stress from physiological signals in response to games and simulations and building models of user affect; in particular for supporting the diagnosis and treatment of post-traumatic stress disorder. This secondary research track never achieved integration with the overarching agenda of decision modeling for automatic play-testing, though a number of possible avenues for integration were identified and tentatively explored. Other work, such as Ahn (2010)'s highly interesting PhD thesis on affective decision making, points to how such integration could be pursued. I briefly touch upon this in the chapters on decision making and player modeling. Though integrating psychophysiology into the procedural personas concept turned out to be out of scope for a three-year PhD thesis, it is certainly a direction worthy of additional research. The papers that resulted from these endeavors are interesting in their own right, but are not integrated in the main thesis, since they belong to a different narrative. Instead, they are included in the appendices for the interested reader.

Below are given four lists together containing all papers and book chapters published during my PhD training: Those which are integrated into the main thesis introductory chapters, those included in the main thesis in their original form, those included in appendices, and a few that do not fit cleanly into any of these categories, or where my involvement in writing the paper or chapter was minor. These texts are omitted from this thesis, but listed for completeness.

## Papers Integrated in Introduction

- Holmgård, Christoffer (2013). “Decision Making in Computer Games”. In: *Foundations of Digital Games, Doctoral Consortium*.
- Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis (2015). “MiniDungeons 2: An Experimental Game for Capturing and Modeling Player Decisions”. In: *10th International Conference on Foundations of Digital Games*.

## Papers Included in the Main Thesis

- Holmgård, Christoffer, Julian Togelius, and Georgios N. Yannakakis (2013). “Decision Making Styles as Deviation from Rational Action. A Super Mario Case Study”. In: *Proceedings of the Ninth Annual AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.
- Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis (2014a). “Evolving Personas for Player Decision Modeling”. In: *IEEE Conference on Computational Intelligence and Games*.
- (2014b). “Generative Agents for Player Decision Modeling in Games”. In: *Foundations of Digital Games*.
- Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis (2014c). “Personas versus Clones for Player Decision Modeling”. In: *13th International Conference on Entertainment Computing (ICEC 2014)*. Springer, pp. 159–166.
- Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis (2015). “Monte-Carlo Tree Search for Persona Based Player Modeling”. In: *Workshop Technical Reports from the Eleventh AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (Player Modeling Workshop at AIIDE-15)*.
- Liapis, Antonios, Christoffer Holmgård, Georgios N. Yannakakis, and Julian Togelius (2015). “Procedural Personas as Critics for Dungeon Generation”. In: *Applications of Evolutionary Computation*. Springer, pp. 331–343.

## Papers Included in the Appendices

- Holmgård, Christoffer, Georgios N. Yannakakis, Karen-Inge Karstoft, and Henrik Steen Andersen (2013a). “Stress Detection for PTSD via the StartleMart Game”. In: *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*. IEEE, pp. 523–528.
- Holmgård, Christoffer, Georgios N. Yannakakis, and Rilla Khaled (2013b). “The Games for Health Prototype”. In: *Foundations of Digital Games, Research and Experimental Games Festival*, pp. 396–399.
- Holmgård, Christoffer, Georgios N. Yannakakis, Héctor P. Martínez, Karen-Inge Karstoft, and Henrik Steen Andersen (2014). “Multimodal PTSD Characterization via the StartleMart Game”. In: *Journal on Multimodal User Interfaces*, pp. 1–13.
- Holmgård, Christoffer, Georgios N. Yannakakis, Héctor P. Martínez, and Karen-Inge Karstoft (2015). “To Rank or to Classify? Annotating Stress for Reliable PTSD Profiling”. In: *Affective Computing and Intelligent Interaction (ACII), 2015 Humaine Association Conference on*. IEEE.
- Pedersen, Christoffer Holmgård and Georgios N. Yannakakis (2012). “Epistemological Challenges for Operationalism in Player Experience Research”. In: *Foundations of Digital Games, Play Experience Workshop - Reconciling PX Methods*.
- Pedersen, Christoffer Holmgård, Rilla Khaled, and Georgios N. Yannakakis (2012). “Ethical Considerations in Designing Adaptive Persuasive Games”. In: *Persuasive Technology*.

## Papers and Book Chapters not Included in this Thesis

- Cheong, Yun-Gyung, Corrado Grappiolo, Christoffer Holmgård, Florian Berger, Rilla Khaled, and Georgios N. Yannakakis (2013). “Towards Validating Game Scenarios for Teaching Conflict Resolution”. In: *Foundations of Digital Games, Games for Learning Workshop*.
- Cheong, Yun-Gyung, Rilla Khaled, Christoffer Holmgård, and Georgios N. Yannakakis (2015). “Serious Games for Teaching Conflict Resolution: Modeling Conflict Dynamics”. In: *Conflict and Multimodal Communication: Social Research and Machine Intelligence*. Springer.
- Gudnadottir, Elin Rut, Alaina K Jensen, Yun-Gyung Cheong, Julian Togelius, Byung Chull Bae, and Christoffer Holmgård Pedersen (2013). “Detecting Predatory Behaviour in Online Game Chats”. In: *The 6th International Conference on Interactive Digital Storytelling (ICIDS), The 2nd Workshop on Games and NLP (GAMNLP-13)*.

# Contents

<b>Abstract</b>	<b>ii</b>
<b>Acknowledgements</b>	<b>iv</b>
<b>Preface</b>	<b>v</b>
<b>Contents</b>	<b>vii</b>
<b>List of Figures</b>	<b>xv</b>
<b>List of Tables</b>	<b>xix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Primary Contributions . . . . .	4
1.2 Secondary Contributions . . . . .	5
1.3 Thesis Overview . . . . .	5
1.4 Chapter Summary . . . . .	8
<b>2 Decision Making</b>	<b>9</b>
2.1 Decision Theory . . . . .	9
2.2 Decision Making Styles . . . . .	11
2.3 Bounded and Adaptive Rationality . . . . .	12
2.4 Naturalistic and Recognition Primed Decision Making . . . . .	13
2.5 Somatic Markers of Decision Making . . . . .	14
2.6 Chapter Summary . . . . .	15
<b>3 Player Modeling and Procedural Content Generation</b>	<b>16</b>
3.1 Player Modeling . . . . .	16
3.2 Procedural Content Generation in Games . . . . .	19
3.3 Chapter Summary . . . . .	20
<b>4 Decision Making in Games</b>	<b>22</b>
4.1 Play Personas and Procedural Personas . . . . .	22
4.2 Identifying Decisions in Games . . . . .	25
4.3 Three Pragmatic Levels of Decision Making in Games . . . . .	27
4.4 Metrics of Decision Similarity . . . . .	28
4.4.1 Action Agreement Ratio . . . . .	28

4.4.2	Tactical Agreement Ratio . . . . .	29
4.4.3	Strategic Agreement Ratio . . . . .	29
4.5	Agent-Control Methods for Procedural Personas . . . . .	32
4.6	Chapter Summary . . . . .	33
<b>5</b>	<b>Domains</b>	<b>35</b>
5.1	The Mario AI Benchmark . . . . .	35
5.2	MiniDungeons 1 . . . . .	37
5.3	MiniDungeons 2 . . . . .	38
5.4	Chapter Summary . . . . .	41
<b>6</b>	<b>Decision Making Styles as Deviation from Rational Action</b>	<b>44</b>
6.1	Abstract . . . . .	44
6.2	Introduction . . . . .	45
6.3	Related Work . . . . .	45
6.3.1	Latent Trait Models of Behavior . . . . .	46
6.3.2	Irrational Play . . . . .	46
6.3.2.1	The Compliant-Constructive Continuum. . . . .	47
6.4	Belief-Desire-Intention Agents . . . . .	47
6.4.1	Decision Making Styles . . . . .	49
6.5	Decision Making in the MAIB . . . . .	49
6.6	Method . . . . .	50
6.6.1	Feature Extraction and Clustering . . . . .	51
6.6.2	Data Collection . . . . .	52
6.7	Results . . . . .	52
6.8	Discussion . . . . .	55
6.9	Conclusion . . . . .	57
6.10	Acknowledgments . . . . .	57
<b>7</b>	<b>Generative Agents for Player Decision Modeling</b>	<b>59</b>
7.1	Abstract . . . . .	59
7.2	Introduction . . . . .	60
7.3	Related Work . . . . .	61
7.3.1	Decision Theory . . . . .	61
7.3.2	Validity in Latent Trait Modeling . . . . .	62
7.3.3	Player Modeling . . . . .	63
7.4	Testbed . . . . .	64
7.4.1	Game Environment . . . . .	64
7.4.2	Generative Agents . . . . .	66
7.5	Data Collection . . . . .	67
7.5.1	Human Playtraces . . . . .	69
7.5.2	Artificial Playtraces . . . . .	70
7.6	Results . . . . .	72
7.7	Discussion . . . . .	74
7.8	Conclusion . . . . .	77
7.9	Acknowledgements . . . . .	77

<b>8</b>	<b>Evolving Personas for Player Decision Modeling</b>	<b>78</b>
8.1	Abstract . . . . .	78
8.2	Introduction . . . . .	79
8.3	Related Work . . . . .	81
8.3.1	Decision Theory . . . . .	81
8.3.2	Player Modeling . . . . .	82
8.3.3	Procedural Content Generation . . . . .	83
8.4	Previous Work . . . . .	83
8.4.1	MiniDungeons . . . . .	83
8.4.2	Previous Experiments with Q-learning . . . . .	84
8.5	Methods . . . . .	86
8.5.1	Evolved Controllers . . . . .	86
8.5.2	Evolutionary algorithm . . . . .	87
8.5.3	Fitness Functions for Evolving Personas . . . . .	88
8.5.4	Decision-level playtrace comparison . . . . .	89
8.5.5	Baseline Agents . . . . .	89
8.6	Results . . . . .	89
8.6.1	Agreements between Q-learning personas and evolved personas . . . . .	91
8.6.2	Agreements between personas and human players . . . . .	93
8.7	Discussion . . . . .	96
8.8	Conclusion . . . . .	96
8.9	Acknowledgements . . . . .	97
<b>9</b>	<b>Personas versus Clones for Player Decision Modeling</b>	<b>98</b>
9.1	Abstract . . . . .	98
9.2	Introduction . . . . .	99
9.2.1	Research Questions . . . . .	100
9.2.2	Prior Work . . . . .	100
9.2.3	Metrics and Methods for Comparing Agents to Human Players . . . . .	101
9.2.4	Modeling Bounded Rationality . . . . .	101
9.3	Related Work . . . . .	102
9.3.1	Decision Theory and Utility . . . . .	102
9.3.2	Personas . . . . .	103
9.3.3	Player Modeling . . . . .	104
9.4	MiniDungeons . . . . .	105
9.5	Agreement Ratios for Evaluating Player Models . . . . .	105
9.5.1	Action Agreement Ratio . . . . .	106
9.5.2	Tactical Agreement Ratio . . . . .	106
9.5.3	Strategic Agreement Ratio . . . . .	107
9.6	Generative Agents . . . . .	107
9.6.1	Evolving Agent Controllers . . . . .	108
9.6.2	Personas . . . . .	108
9.6.3	Clones . . . . .	110
9.6.4	Specialized Agents . . . . .	110
9.7	Results . . . . .	110
9.8	Discussion . . . . .	114
9.8.1	AAR, TAR, and SAR metrics . . . . .	116

9.8.2	Controller Architecture Performance . . . . .	117
9.8.3	Personas and Clones for MiniDungeons and Beyond . . . . .	117
9.8.4	Combining Personas and Clones . . . . .	118
9.9	Conclusion . . . . .	119
9.10	Acknowledgments . . . . .	120
<b>10</b>	<b>Procedural Personas as Critics for Dungeon Generation</b>	<b>121</b>
10.1	Abstract . . . . .	121
10.2	Introduction . . . . .	121
10.3	Previous Work . . . . .	122
10.3.1	MiniDungeons game . . . . .	122
10.3.2	Procedural Personas . . . . .	124
10.3.3	Constrained Optimization of Game Levels . . . . .	125
10.4	Methodology . . . . .	126
10.4.1	Evolving MiniDungeon levels . . . . .	126
10.4.2	Assessing Playability with Personas . . . . .	127
10.4.3	Assessing Level Quality with Personas . . . . .	128
10.5	Experiments . . . . .	130
10.5.1	Discovery of feasible content . . . . .	130
10.5.2	Quality of feasible content . . . . .	131
10.6	Discussion . . . . .	135
10.7	Conclusion . . . . .	136
10.8	Acknowledgements . . . . .	136
<b>11</b>	<b>Monte-Carlo Tree Search for Persona Based Player Modeling</b>	<b>137</b>
11.1	Abstract . . . . .	137
11.2	Introduction . . . . .	138
11.3	Related Work . . . . .	139
11.4	MiniDungeons 1 Game Design . . . . .	140
11.5	MiniDungeons 2 Game Design . . . . .	140
11.6	Persona Design for MiniDungeons 2 . . . . .	142
11.7	Monte Carlo Tree Search for Persona Control . . . . .	142
11.8	Metrics . . . . .	143
11.8.1	Action Agreement Ratio . . . . .	143
11.8.2	Summary Statistics and Heat-Maps for Persona Identification and Skill Evaluation . . . . .	144
11.9	Results . . . . .	144
11.9.1	Action Agreement Ratios . . . . .	145
11.9.2	Summary Statistics and Heat-Maps . . . . .	145
11.10	Discussion . . . . .	148
11.11	Conclusion . . . . .	149
11.12	Acknowledgments . . . . .	149
<b>12</b>	<b>Discussion</b>	<b>151</b>
12.1	Establishing Reference Processes in State Spaces . . . . .	152
12.2	Validity: Do Procedural Personas Represent Decision Making Styles? . . .	153
12.3	Metrics and Operationalization . . . . .	155



12.4	Extensibility and Usefulness . . . . .	156
12.5	Chapter Summary . . . . .	157
<b>13</b>	<b>Future Work</b>	<b>159</b>
13.1	Learning Utility Functions for New Personas . . . . .	160
13.2	Alternative Agent Control Architectures . . . . .	161
13.2.1	Emphasizing Model-Free Aspects of Procedural Personas . . . . .	161
13.2.2	Emphasizing Model-Based Aspects of Procedural Personas . . . . .	163
13.3	Correspondence with Other Measures . . . . .	164
13.4	Future Implementations and Studies with Procedural Personas . . . . .	164
13.5	Chapter Summary . . . . .	165
<b>14</b>	<b>Conclusion</b>	<b>166</b>
<b>A</b>	<b>Epistemological Challenges for Operationalism in Player Experience Research</b>	<b>167</b>
<b>B</b>	<b>Ethical Considerations in Designing Adaptive Persuasive Games</b>	<b>169</b>
B.1	Abstract . . . . .	169
B.2	Introduction . . . . .	169
B.3	The ATG prototype. . . . .	170
B.4	Discussion . . . . .	171
B.5	Conclusion and Future Work . . . . .	173
<b>C</b>	<b>The Games for Health Prototype</b>	<b>174</b>
C.1	Abstract . . . . .	174
C.2	Introduction . . . . .	174
C.3	Background . . . . .	175
C.4	Game Design Considerations . . . . .	176
C.5	Playing the Game . . . . .	178
C.5.1	The Game Environment . . . . .	179
C.5.2	Hardware and Setup . . . . .	180
C.6	Clinical Trials . . . . .	181
C.7	Future Work . . . . .	181
C.8	Conclusion . . . . .	181
C.9	Acknowledgments . . . . .	182
<b>D</b>	<b>Stress Detection for PTSD via the StartleMart Game</b>	<b>183</b>
D.1	Abstract . . . . .	183
D.2	Introduction . . . . .	184
D.3	Background: Stress Detection and PTSD Treatment . . . . .	185
D.3.1	Stress Detection . . . . .	185
D.3.2	Physiology of PTSD . . . . .	186
D.3.3	Treating Stress Disorders . . . . .	186
D.3.4	Games for Mental Health . . . . .	186
D.4	The StartleMart Game for PTSD Treatment . . . . .	187
D.4.1	Game Stressors . . . . .	189

D.5	Experimental Protocol and Data Collection . . . . .	190
D.5.1	Physiological Sensors and Setup . . . . .	191
D.6	User Data Features . . . . .	191
D.6.1	PTSD profile . . . . .	191
D.6.2	Self-Reports of Stress . . . . .	192
D.6.3	Features extracted from Skin Conductance . . . . .	192
D.7	Results . . . . .	193
D.7.1	Correlation analysis between PTSD profile and SC features . . . . .	193
D.7.2	Correlation analysis between self-reports and SC features . . . . .	194
D.8	Discussion . . . . .	195
D.9	Conclusion . . . . .	196
<b>E</b>	<b>Multimodal PTSD Characterization via the StartleMart Game</b>	<b>197</b>
E.1	Abstract . . . . .	197
E.2	Introduction . . . . .	198
E.3	Stress Detection . . . . .	200
E.3.1	Physiology of PTSD . . . . .	201
E.4	The StartleMart Game for PTSD Treatment . . . . .	201
E.5	Experimental Protocol and Data Collection . . . . .	202
E.5.1	Physiological Sensors and Setup . . . . .	202
E.6	User Data Features . . . . .	204
E.6.1	PTSD Profile . . . . .	204
E.6.2	Self-Reports of Stress . . . . .	205
E.7	Features Extracted from Physiological Signals . . . . .	205
E.7.1	Features Extracted from Skin Conductance . . . . .	205
E.7.2	Features Extracted from Blood Volume Pulse . . . . .	206
E.8	Results . . . . .	207
E.8.1	Correlations Between PTSD Profile and SC Features . . . . .	209
E.8.2	Correlations Between Self-Reports and SC Features . . . . .	211
E.8.3	Correlations between PTSD Profile and HR/BVP Features . . . . .	211
E.8.4	Correlations between Self-Reports and HR/BVP Features . . . . .	213
E.8.5	Principal Component Analysis of Physiological Features and their Relations to PTSD symptom severity . . . . .	213
E.9	Discussion . . . . .	215
E.10	Conclusion . . . . .	218
<b>F</b>	<b>To Rank or to Classify? Annotating Stress for Reliable PTSD Profiling</b>	<b>220</b>
F.1	Abstract . . . . .	220
F.2	Introduction . . . . .	221
F.3	Background: PTSD, Games, and Affect . . . . .	222
F.3.1	Post Traumatic Stress Disorder . . . . .	222
F.3.2	Games for PTSD treatment . . . . .	222
F.3.3	Physiology of PTSD . . . . .	223
F.4	The StartleMart Game for PTSD Treatment . . . . .	224
F.5	Experimental Protocol and Data Collection . . . . .	226
F.5.1	Physiological Sensors and Setup . . . . .	226
F.5.2	Game Logging . . . . .	227

---

F.6	Self-reports and SC Feature Extraction . . . . .	227
F.6.1	Stress Self-reports . . . . .	227
F.6.2	Skin Conductance Event Response Features . . . . .	228
F.7	Results . . . . .	229
F.7.1	Classifying Stress . . . . .	230
F.7.2	Ranking Stress . . . . .	231
F.8	Discussion and Conclusions . . . . .	232

**Bibliography****234**

# List of Figures

1.1	The three fields that procedural personas integrate and the three primary applications for procedural personas. . . . .	4
4.1	An illustration of how the Action Agreement Ratio is calculated. . . . .	28
4.2	An illustration of how the Tactical Agreement Ratio is calculated. . . . .	30
4.3	An illustration of how the Strategic Agreement Ratio is calculated. . . . .	31
5.1	A screen-shot from the Mario AI Benchmark Framework, showing a few different potential paths form the left-hand side of the screen to the right-hand side. . . . .	36
5.2	A level from MiniDungeons 1. The only moving character is the player’s. The main challenge is to decide what affordances to pursue in the level under the risk that each monster may deal from 5 to 14 HP worth of damage. . . . .	39
5.3	A typical MiniDungeons 2 level at start (left panel) and after three turns (right panel). . . . .	42
6.1	Dendrogram from the agglomerative hierarchical clustering of the extracted features from 400 observations. The tree was cut at the level of four clusters. . . . .	52
6.2	Players’ individual cluster expressions across all play throughs. Each column represents a player and each shading within each column represents how many of the individual player’s 40 play sessions were grouped into clusters $C_1$ , $C_2$ , $C_3$ , and $C_4$ respectively. . . . .	53
6.3	Expressed play style clusters mapped to levels. Each shading represents a cluster. . . . .	54
6.4	Traces from three levels. Each graph represents a trace of an individual player or the agent. Note that the high-performing $C_2$ traces bear resemblance to the agent traces. . . . .	58
7.1	The game environment on one of the levels used in the experimental protocol. The hero, shown in gray armor, moves around the level collecting treasures (brown closed chests), potions (red bottles) and killing enemies (green goblins). The hero starts at the entrance (stairway leading up, left of the screen) and the level ends when the exit is reached (stairway going down, right of the screen). The hero’s hitpoints are shown at the bottom, along with the number of treasures collected and the most recent event. . . . .	65
7.2	The levels used for the data collections experiment. The tutorial level is hand-crafted, and could be played multiple times. The “real” levels (1-10) were played only once (no retries if the hero died) and were created in a mixed-initiative fashion. . . . .	68

---

7.3	Heatmaps of selected players in Level 2. Some acted as “completionists”; others rushed to the exit. Many players only collected guarded items if the risk was limited while others took excessive risks and died. The heatmaps indicate that a single level allows for different decision making styles in spite of the apparent simplicity of the testbed. . . . .	71
7.4	Some indicative heatmaps of trained agents on Level 2 (Fig. 7.4a–7.4c) and Level 8 (Fig. 7.4d–7.4f). The different playstyles of the agent-personas are showcased, although in Level 2 agents B, R and S all share the heatmap of Fig. 7.4a. . . . .	72
7.5	The best agreeing agent for each level. The dotted line indicates the mean agreement of the best fitting agent-personas across all players on each level.	75
8.1	The intended designer use of procedural personas. . . . .	80
8.2	The levels included in MiniDungeons. The tutorial level is hand-crafted, and could be played multiple times. The “real” levels (1-10) were played only once (no retries if the hero died) and were created in a mixed-initiative fashion. . . . .	84
8.3	The MiniDungeons interface while a game is played. . . . .	85
8.4	The controller network which was evolved to generate the five personas. .	87
8.5	Heatmaps of all personas and baseline agents on level 7 of MiniDungeons.	92
9.1	Heat-maps of six selected human play-traces in Level 2 of MiniDungeons, showing a diversity of player decision making styles. Note that in two heat-maps, top center and bottom right, the player died before completing the level. . . . .	106
9.2	The controller network used for all agents: personas, clones, and specialized agents. The weights of the connections are determined through evolution as described in Section 9.6.1. For every action it takes in the game, a controller uses the current state of the game, represented by the 8 input nodes (disregarding the bias node), and the weights to select which of the 7 strategies represented by the output nodes to pursue. . . . .	109
9.3	Heat-maps from one player and all best matching personas, all clones, and all specialized agents on Level 2. For each agent the maximally attained agreement on each metric is indicated in the order AAR; TAR; SAR. The metric used to drive the evolution of each agent, excepting the persona, is indicated in bold. The best matching persona, by all metrics, was the Treasure Collector. . . . .	115
10.1	The levels used for collecting player data and for evolving procedural personas. . . . .	123
10.2	Playtraces of human players and evolved procedural personas of MiniDungeons. . . . .	124
10.3	Best evolved levels. . . . .	133
11.1	Map 7 from MiniDungeons 2. . . . .	141

11.2	Heat-maps of end game states exemplifying differences in the personas' behaviors, taken from personas given 1s of decision making time. Different maps are chosen to best showcase the personas' decision making styles. Green indicates tiles a persona moved to and red indicates a javelin attack. In sub-figures (a) and (b) the Exit and Runner personas manage to finish the level, while in sub-figures (c), (d), and (e) the personas die before reaching the end of the level. The Survivalist pursues two potions, but ultimately dies, and the Monster Killer manages to kill all monsters in the level, but does not survive. Finally, the Treasure Collector gets all the treasure, but dies in the process. . . . .	150
B.1	Screenshots from the three modes of ATG . . . . .	171
C.1	The supermarket. A man is walking rapidly and angrily down the aisle. .	177
C.2	A flashback scene presented immediately after the game state of Figure C.1. The angrily walking man bled into a vision of a man running towards the player. . . . .	178
D.1	The three flashbacks of the game (b, d, f) and the immediately preceding supermarket scenes (a, c, e). Elements of the supermarket bleed into the flashbacks, simulating re-experience. . . . .	188
D.2	An example of an SC signal recorded from a single game session: Subject no. 5, game session 3 (high stress intensity). . . . .	190
E.1	The three traumatic experience cues of the game (b, d, f) and the immediately preceding stressful scenes from everyday life (a, c, e). Elements of the everyday life scenes bleed into the cue scenes, referencing re-experience, a symptom typical for PTSD. . . . .	203
E.2	An example of an SC and a BVP signal recorded from a single game session: Subject no. 5, game session 3. . . . .	204
E.3	Scree plot of all 45 principal components derived from the original feature set. The scree plot shows two components explaining most of the variance in the data with the remaining components explain relatively less. . . .	214
F.1	The three flashbacks of the game (b, d, f) and the immediately preceding supermarket scenes (a, c, e). Elements of the supermarket bleed into the flashbacks, simulating re-experience. . . . .	225
F.2	Continuous Decomposition Analysis of SC of Player 5, Session 3. The top graph shows the full game session whereas the bottom graph shows a detailed view of an excerpt from that session. Both graphs depict three components extracted from the raw SC signal: Phasic activity (yellow), tonic activity (orange), and the phasic driver (red) of SC. SC features are extracted within the event response windows as illustrated in the bottom graph. . . . .	229

---

F.3 Individual event response patterns: Patients and event types are cross tabulated. The color of each cell represents the patient’s mean normalized  $S_p$  response to the event type, across all sessions, extracted using Continuous Decomposition Analysis (Benedek and Kaernbach, 2010a). The size of each green dot indicates the number of times the patient picks the corresponding type of event as the most stressful. For fair comparison purposes, both the event response levels and the event preference frequencies are normalized into  $[0, 1]$  within each individual, across all sessions. . . . . 231

# List of Tables

6.1	Spearman correlation between cluster membership frequency, features, and performance measures. Values significant at the $p < 0.05$ level are in <b>bold</b> . Significance values are subjected to Holm-Bonferroni correction for multiple tests. . . . .	53
6.2	Kruskal-Wallis H tests for differences between clusters. $C1-C4$ contains the Hodges-Lehmann Estimator of Location (Lehmann and D’Abrera, 1975), as a measure of centrality, for each cluster for each feature to allow for comparison. Note that for some features, e.g. <i>Speed</i> , all clusters are different from one another, while for others, e.g. <i>Jump</i> , one cluster deviates. 55	
7.1	Description of agents. . . . .	67
7.2	Rewards $r$ for specific game events. . . . .	67
7.3	Game statistics of each artificial agent for the entire playthrough of 10 levels. With the exception of Times Died, values are averaged across 20 test runs; Times Died includes all 200 playthroughs tested. . . . .	72
7.4	Frequencies of agent-persona best fit across levels. . . . .	74
7.5	Statistics of the individual agent-personas. All agent-personas attain high maximal values. This indicates that all agents, except for the random controller, are relevant approximations of some players. . . . .	74
8.1	All personas tested in the experiments. . . . .	90
8.2	Baseline agents used in experiments. . . . .	90
8.3	Average statistics of each persona’s play traces across all levels in MiniDungeons. M stands for monsters killed, T for treasures collected, P for potions drunk, Ex for tiles explored, and D for times died. With the exception of D, values are averaged across 20 test runs; D includes all playthroughs tested. . . . .	91
8.4	Agreements between individual personas based on human players’ play traces. . . . .	93
8.5	Agreements between personas/baseline agents and human players. Evolved personas which were generalized by holding out the test level are marked with a $\Sigma$ . . . . .	93
8.6	Frequencies of personas and baseline agents being the best match for individual human play traces. . . . .	94
8.7	Mean agreement ratios for personas and baseline agents when they were selected as best matches. . . . .	95
8.8	Mean agreement ratio across all traces from all levels where personas or agents were selected as best matches. . . . .	95
8.9	Wilcoxon Rank Sum Test for differences between persona pairs, with personas created either via Q-learning or evolution. . . . .	96



9.1	Utility weights for the five designed personas. . . . .	109
9.2	Means and standard deviations (SD) of agreement ratios attained for best matching personas, clones, and agents evolved using the described fitness functions. It is worth noting that across all three metrics, personas generally exhibit performance close to that of the clones. Additionally, it is worth noting that while specialized agents evolved using TAR and SAR as fitness functions perform best in the metric they were evolved from, this is not the case for agents evolved using the AAR metric as a fitness function. Using the AAR metric, the TAR-evolved specialized agents exhibit the best performance. . . . .	111
9.3	Best persona matches based on Action Agreement Ratio (AAR), Tactical Agreement Ratio (TAR), and Strategic Agreement Ratio (SAR), respectively. . . . .	113
10.1	Simulation metrics of the best final levels. . . . .	134
11.1	All personas tested in the experiments. The five different personas are awarded utility from seven affordances in MiniDungeons 2: Taking a turn (Tu), reducing the distance to the exit (Di), killing a monster (M), collecting a treasure (Tr), drinking a potion (P), dying (D) or reaching the exit of a level (E). . . . .	142
11.2	Action Agreement Ratios (AAR) between all personas. The AARs range from 0 to 1 and indicate the extent to which two personas agree on atomic in-game decisions. The results indicate that personas agree with themselves to a larger extent than with other personas. However, it is also evident that some variation happens within personas, likely due to the stochasticity of the MCTS algorithm. All AARs were calculated with 1s of decision making time per action. . . . .	145
11.3	Play summary statistics summed across the 10 maps under different computational budgets: 10ms, 100ms, 1s, 10s, and 60s. Each condition totaled 200 runs per persona, 20 per map. . . . .	147
D.1	PTSD profile features . . . . .	192
D.2	Correlations $\rho$ between SC signal and PTSD profile features in the left section of the table. Correlations $c(z)$ between SC signal and self-reported stress in the right section of the table. Statistically significant correlations appear in bold (p-value < 0.05) and italics (p-value < 0.10). . . . .	194
E.1	PTSD profile features . . . . .	205
E.2	Overview of features extracted from SC and BVP. . . . .	208
E.3	Correlations $\rho$ between physiological signal features and PTSD profile features are in the left section of the table. Correlations $c(z)$ between physiological signal features and self-reported stress are in the right section of the table. Statistically significant correlations appear in bold (p < 0.05) and italics (p < 0.10). . . . .	210
E.4	Standard deviation and explained variance for each of the two principal components extracted from the feature set. . . . .	213

---

E.5	Correlations between individual features and the two extracted principal components. For each component, the features listed are sorted according to the absolute value of the correlation coefficient in order to allow for a straightforward identification of the features with the strongest relation to the component. . . . .	216
E.6	Correlations $\rho$ and $c(z)$ between the two principal components and Age, PCL, $N_{day}$ , and $N_{dep}$ , and self-reports. Statistically significant correlations appear in bold ( $p < 0.05$ ) and italics ( $p < 0.10$ ). . . . .	217
F.1	Pairwise rank-correlations ( $c(z)$ ) and corresponding p-values ( $p$ ) between the most stressful event reported (class) or the reported rankings of stressful events (preference) and the four skin conductance features. . . . .	232

# Chapter 1

## Introduction

This thesis addresses the topic of modeling player decision making in computer games. It introduces the concept of *procedural personas*: Game playing agents that codify player decision making styles, either from the designer’s holistic representation of these or from observations of players collected directly from the game. These procedural personas, in turn, become proxies for actual human players, acting as stand-ins that play the game in place of human players, enabling *procedural play-testing* by generating synthetic play-traces.

Extending the *play persona* concept (Canossa and Drachen, 2009), procedural personas represent archetypal ways of playing the particular game and by interfacing with the game through the same operations as the human players, it allows for the application of the analytical and visualization tools that game designers would otherwise use to make sense of human play. For reasons that will be explicated later in this chapter, the procedural personas in this thesis are constructed around the notion of simulating human *decision making styles*.

The fundamental research question for this thesis is: “How can we computationally model and simulate human decision making to facilitate the exploration of (digital) games?”

While other perspectives onto characterizing and simulating behavior in games could be equally valid, centering on decision making offers a number of advantages that this work seeks to exploit. The study of human decision making has a long history in psychology which provides a substantial literature on which to draw when defining decisions and understanding player decision making. It provides a theoretical basis from which to operationalize decision making into observable in-game behavior.

Insights into human decision making from these perspectives are often based on reductive models of human cognition and motivation, and their empirical data is often collected from highly structured decision making experiments that limit participant's decision making freedom (see e.g. Schulte-Mecklenbeck et al., 2010).

While this approach is sometimes criticized for lacking ecological validity (Gregory, 2004), particularly within psychology's naturalistic decision making tradition (Kahneman and Klein, 2009), it is well-suited for understanding decision making in games. Many games, and digital games in particular, are exactly characterized by being highly structured environments that limit the decision making freedom of their players, creating situations that make decisions clear and meaningful and not wholly unlike the experimental paradigms found in decision science.

This thesis argues that psychological decision science provides a useful theoretical foundation for the computational modeling of player decision making and supports this claim through a number of demonstrations in specific game domains.

The purpose of the computational models of decision making presented in this thesis is to develop new methods for understanding how players choose to engage with games as decision spaces. While this problem is interesting in itself, modeling decision making also has direct applicability in relation to play-testing and procedural content generation in games.

Play-testing is a crucial stage of game development, analog or digital (Fullerton et al., 2004). Some might even argue that it is an integral part of imagining new games: A game designer happens upon an idea for a game: a mechanic, a set of rules, a setting, or an ambiance. Instantly, in her mind, she imagines how the game plays. The first play-test has already happened, and the game has not seen a single player yet. The designer mentally simulated her players, extrapolating their behavior from her experience across many other games, participating in, observing, designing for play. From these experiences she already has a range of archetypal player behaviors distilled from the many individual examples. These individual archetypes represent play-styles; various ways that different kinds of people like to interact with games. A mental design-test-evaluate loop has occurred, simulating and evaluating a low-fidelity version of the first design of the game.

Play-testing as a concept and practice can be broken down along many different dimensions (Fullerton et al., 2004; Elias et al., 2012). Three possible dimensions are *when* the play-test occurs, *what* is observed, and *who* participates.

Modern play-testing of digital games may be conceptualized as ranging from the low-fidelity idea generating process over the fully instrumented laboratory runs of large game

development studios to the telemetric measurement, storing, and data mining of player behavior from released games that are already in the market. This is one dimension of play-testing: When in the game's life-cycle it takes place.

A second dimension is what classes of information are gathered from the players participating in the play-test: A non-exhaustive list of options from multiple modalities includes in-game behavioral data, video data, emotional responses, verbal reactions, subjective reports from players or a designer's impression of player responses. They range from the atomic and strictly operationalised, e.g. a single action at a single point in time, to the holistic and phenomenological, e.g. a free-form narrative from a play-tester describing her play experience (Fullerton et al., 2004; El-Nasr et al., 2013).

A third dimension of play-testing is the number of individuals that take part in the play-testing process. From the example above, which includes only the designer of the game, to a released, telemetrically enabled game which in practice engages every player with an Internet connection in a continuous play-test. Typically, the more complete the game is, the easier it becomes to gather atomic play-test data at a larger scale enabling fine tuning via small changes (El-Nasr et al., 2013).

Earlier in a game's development cycle, play-testing requires bringing players and developers together, often in the same room. This puts a limitation on the amount of play-testing that is logistically and financially feasible to conduct for most games. Minor changes to game rules or pieces of content such as level designs may not be significant enough to warrant new play-tests with players. As a consequence, the designer making these individually small, but collectively important, changes relies heavily on the imaginal play-testing loop, supported by various analytical tools, when she is designing or changing a piece of content for her game.

When the designer opts to create a game reliant on procedural content generation, whole parts of this iterative production and testing loops are embedded into the game itself as an artifact. Here, automated play-testing would allow designers to incorporate models of player decision making as strictly encoded representations of their imagined players, controlling and driving the procedural content generation process.

This thesis explores how to build technology that may support the designer in the play-testing process, creating an analytical tool that helps designers explicate and codify assumptions about player motivations. It focuses on the stage in a game's life-cycle when it is *still under development, but the rules are well-defined*, where it would be useful to observe *individual player actions in response to specific pieces of content* as they are being created, and when *play-testers* external to the development team are *infrequently available or not available at all*. This could be when a game development

team is expanding the content of a game by adding puzzles and levels, or it could be when a procedural content generation system is generating content for a game.

## 1.1 Primary Contributions

The thesis provides a highly non-exhaustive answer to the research question through a number of specific contributions to the literature on player modeling, listed below and illustrated in Figure 1.1.

- It extends play persona theory by adding a generative dimension to the preexisting descriptive and prescriptive dimensions in the form of procedural personas.
- It forges a connection between contemporary decision science, player modeling and a number of modern AI agent control methods.
- It argues that procedural personas can provide dynamic reference points for understanding possible trajectories in games.
- It shows that procedural personas may be used to model archetypal decision making styles in three test-beds with four different agent control architectures.
- It demonstrates that procedural personas can be used as computational critics shaping the output search-based procedural content generation systems.
- It provides two novel test-beds for studying and modeling decision making in games to the community, one centered on decision making under risk and the other centered on bounded rationality.

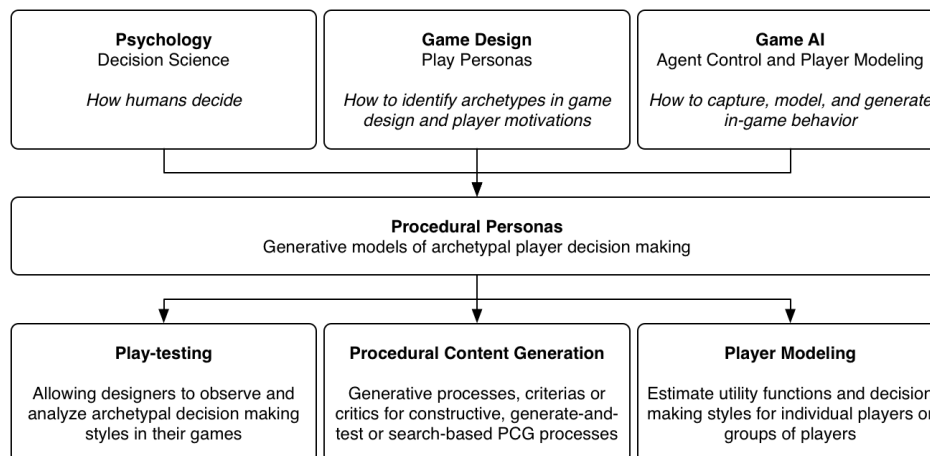


FIGURE 1.1: The three fields that procedural personas integrate and the three primary applications for procedural personas.

## 1.2 Secondary Contributions

In addition to the primary contributions presented in the main narrative of this thesis, the thesis work provides a number of secondary contributions to the literature on player modeling in relation to stress detection in sufferers of post-traumatic stress disorder in relation to the *StartleMart* game. The secondary contributions are not included in the main narrative of this thesis, as they are only tangentially related to the main theme, but the produced papers are included in the appendices for the sake of completeness. This secondary strain of research brings the following contributions to the literature on applied games and player modeling:

- It demonstrates how modeling responses to games and simulations from physiological signals may be used to characterize the symptoms of patients suffering from post-traumatic stress disorder.
- It introduces a novel paradigm for combining games, multi-modal physiological player modeling, and stress-inoculation therapy: *StartleMart*.
- It provides evidence toward preference based ranking being a more stable paradigm for self-reporting experiences of stress than numerical rating.

Below, the following section provides a brief overview of how the primary contributions will be presented in this thesis.

## 1.3 Thesis Overview

The remainder of this thesis is organized into 13 chapters.

In Chapters 2 to 5, a general summary of the theoretical background and position of the thesis is provided along with a review of the methods and domains used for the empirical work.

The following six chapters, Chapters 6 to 11, each contain a paper contributing to developing the procedural persona concept. Each paper is included in its published form, adapted to the layout of this thesis. The papers are ordered chronologically by their date of publication. This order also matches the development of the procedural persona concept and thus is well suited to trace the theoretical development and empirical exploration of the concept.

This development starts at the general notion of using agents to represent certain decision making styles in games and gradually develops into the persona concept, which

ultimately is tested in the two MiniDungeons test-beds. For five of the six papers, I was main author and conducted the bulk of the work pertaining to persona development, implementation, and testing. The fifth paper “Personas as Critics in MiniDungeons” was written by Antonios Liapis as first author and he conducted the main parts of the work and research. The personas enabling the experiments in the paper, however, are the ones developed in the earlier study reported in “Evolving Personas from Utility Configurations”. The complete author lists and venues for the papers may be found in the Preface of this thesis and in the chapters reproducing the papers.

After presenting the six papers, the thesis moves on to summarize the findings from this research agenda so far, suggesting future directions of research for developing the procedural persona concept, and concludes in the final chapter.

**Chapter 2: Decision Making**, gives an introduction to the work that underpins the theoretical position of the thesis. It starts out with a description of three perspectives from the decision sciences in psychology and examines the question of what defines rationality and how human decision makers exhibit rationality. It also extracts a number of key requirements to agents simulating human decision making from each perspective presented.

**Chapter 3: Player Modeling and Procedural Content Generation**, provides a short review of player modeling and search-based procedural content generation in games in order to contextualize the concept of procedural personas.

**Chapter 4: Decision Making in Games**, links the review of general theories on decision making to decision making in games. The chapter further applies decision science to identify features of games that are relevant to studying human decision making. Given the many different perspectives that can be taken on games as systems in which players make decisions, how do we choose a general approach for identifying decision points in games? A pragmatic solution for the purpose of developing the procedural persona concept, building on affordances, is proposed. Future avenues for formalizing and improving this process, working from design patterns, are suggested. An introduction to play persona theory, the primary foundation for the procedural persona method, is given and the procedural persona concept is introduced as an extension of the play persona concept. Three metrics for evaluating decision making likeness between decisions made by humans and simulated decision makers are introduced. Finally, the specific agent control methods used to control the procedural personas are presented.



**Chapter 5: Domains**, introduces the games that were used to develop and test the procedural persona concept: The Mario AI Benchmark Framework, MiniDungeons 1, and MiniDungeons 2.

**Chapter 6: Decision Making Styles as Deviation from Rational Action**, the first paper, uses the Mario AI Benchmark to understand how human play behavior can be compared to agent behavior and how different kinds of deviations from a reference agent can define archetypal ways of playing a game; in this case the Mario AI Benchmark.

**Chapter 7: Generative Agents for Player Decision Modeling**, the second paper, focuses on using Q-learning to both generate and identify archetypal behavior, this time in the MiniDungeons 1 game.

**Chapter 8: Evolving Personas from Utility Configurations**, the third paper, follows up on the results from Chapter 7 by addressing the problem that the personas created via Q-learning did not generalize to unseen content. To solve this problem, this study evolves sets of linear perceptrons to control the personas and arrives at a more generalizable solution.

**Chapter 9: Personas versus Clones for Player Decision Modeling**, the fourth paper<sup>1</sup> returns to the question of relating persona behavior to human player behavior. Again using MiniDungeons 1, this paper asks how the use of personas, representing designer intuition and expertise, performs in comparison to simply learning by observing human players. The paper also investigates the properties of the controller architecture developed in Chapter 8 in greater detail.

**Chapter 10: Procedural Personas as Critics for Dungeon Generation**, the fifth paper, brings the study of procedural personas in MiniDungeons 1 full circle, by integrating the personas in Antonios Liapis' (2014) mixed-initiative level design tool, *Sentient Sketchbook*. Here, the personas are used to not only test and evaluate human-created levels, but also to come up with new solutions presented to human designers. This study represents the first full-circle proof-of-concept of the procedural persona method.

**Chapter 11: Monte-Carlo Tree Search for Persona Based Player Modeling**, the sixth paper included in this thesis, moves on to the domain of MiniDungeons 2. In contrast to MiniDungeons 1, MiniDungeons 2 contains no hidden information or stochasticity and thus does not represent an environment where decisions must be

---

<sup>1</sup>The version of the paper reported here was expanded from the published version with additional theoretical considerations and more comprehensive results. This expanded version is, at the time of writing, under review with the journal *Entertainment Computing*.

made under risk. Instead, the fully deterministic game represents a decision space with a high branching factor and many interacting parts that players cannot solve analytically. They must rely on a combination of analytic and heuristic reasoning. This paper uses on-line search based agents, moving away from off-line training. It uses an architecture that combines tree-search with heuristics to create procedural personas that simulate this two-step analytic and heuristic decision making process in human players.

**Chapter 12: Discussion**, provides a recapitulation of the findings from the six studies and a discussion of the performance, validity, intelligibility, usefulness, and extensibility of the procedural persona concept.

**Chapter 13: Future Work**, continues where the discussion of Chapter 12 left off, identifying future potentially fruitful directions of research into the theoretical and practical aspects of the procedural personas concept.

**Chapter 14: Conclusion**, summarizes the findings and contributions of the thesis.

**Appendices A to F**, reproduce six papers published in relation to the *StartleMart* game, a game developed under the *Games for Health* project with the purpose of investigating the use of games and simulations as diagnosis and treatment tools for post-traumatic stress disorder.

## 1.4 Chapter Summary

In this chapter an introduction was given to the thesis and its research question, pertaining to the modeling and simulation of player decision making styles. A list of the main contributions, which form the main narrative of the thesis, was provided along with a list of secondary contributions which are included in the appendices of the thesis.

The following chapter reviews three main psychological perspectives on decision making and connects them to computational modeling and simulation of decision making.

## Chapter 2

# Decision Making

The procedural personas presented in this thesis are constructed around modeling and simulating human decision making, building on a combination of *psychological decision science* and *play persona theory*. This chapter introduces three psychological perspectives and relates them to agent based simulation of decision making: *Decision Theory*, *Bounded and Adaptive Rationality*, and *Recognition Primed Decision Making*. From each perspective, key design requirements to decision simulating agents are extracted which are later related to the procedural persona concept. Chapter 3 provides an overview of player modeling and procedural content generation and relates these to decision science. Chapter 4, in turn, introduces play personas and extends them into procedural personas, a form of decision science based player modeling, via agent based simulation.

### 2.1 Decision Theory

Decision theory as a field was founded by Tversky and Kahneman (1974). Their work built on and presented a theoretical and empirical critique of expected utility theory which adhered to the idea of humans as perfectly rational decision makers; *homo economicus*. Tversky and Kahneman documented a vast number of identifiable heuristic processes that guide, shorten or bypass conscious analytical decision making. They further documented that humans sometimes apply heuristics to the extent that they become biases; systematically skewed evaluations in spite of evidence that, subjected to detailed analysis, would have yielded different conclusions.

Decision theory did, however, retain the notion of the personal *utility function*. The personal utility function describes an individual decision maker's valuation of different potential outcomes of a particular decision, in a given moment, in a given context.

Rather than assume that decision makers were rationally optimizing for utility, decision theory documented how most decision processes in humans were only partially analytic and relied heavily on biases and heuristics shaping decision outcomes (Kahneman, 2011).

This two-step decision making process is described by decision theory as the outcome of the interaction of two psychological systems:

**System 1:** A quickly classifying, motivating, impulsive, and highly parallel emotional process. Capable of rapidly making decisions, but prone to bias and misclassification (the plausible neuro-psychological basis of system is extensively studied in the work of Damasio, see section 2.5).

**System 2:** A slower, laborious, attention demanding, memory limited, sequential process that delivers precise responses and inhibits motivations and biased outputs from system 1.

Their interaction produces results that are systematically shaped by the biases of system 1 and the performance capabilities of system 2, both properties that are somewhat stable within the individual (Kahneman, 2011). These stable tendencies, or decision making styles, should be expressed through actions in game-play in the same way that they are expressed across other activities in everyday life.

Since biases and heuristics, and utility functions themselves, are highly responsive to context, much decision theoretical research has been carried out in strict experiments under laboratory conditions. As a consequence, decision theory is sometimes criticized for being based on too reductive models making unwarranted generalizations about human decision making from experiments carried out in reduced, structured environments. For instance, the naturalistic decision making (Lipshitz et al., 2001) perspective, which focuses on individual case studies and a qualitative approach, is often positioned in contrast to decision theory (Kahneman and Klein, 2009).

While this criticism is relevant, and even acknowledged by decision theory, the fact that most decision theoretical research is conducted in highly structured environments makes the perspective well suited for many games, digital and otherwise. Understood as ergodic artifacts that shape the behavior of their players as they work their way through its structures (Aarseth, 1997; Juul, 2011) many games derive their appeal exactly from being limited decision making spaces where each decision can be analyzed and the consequences predicted with either deterministic accuracy or within some stochastic range, depending on whether the game is focused on luck, skill, or a combination thereof (Elias et al., 2012)<sup>1</sup>.

---

<sup>1</sup>While the classical categories of Caillois (2001) also include simulation (mimicry) and vertigo (ilinx), we consider these out of scope for the study of decision making in this thesis.

An important point of decision theory is that the biases and heuristics applied by human decision makers in all manners of decisions may in fact often be chosen inappropriately for reasons that may not be consciously available to the decision maker (Kahneman, 2011).

For the case of many games this may mean that in-game decisions that appear to be a consequence of lack of skill or understanding of the game may in fact be the result of shortcuts learned from unrelated, and hence potentially inappropriate, games or other domains. For the objective of constructing agents simulating human decision making in games, this means that agent control architectures should support the inclusion of biases and/or heuristics that may not always be optimal for the individual decision. Determining the influences of these biases or the considerations of the heuristics in human players will, however, be a challenging task, as they are not necessarily consciously available to players and hence must be observed from game-play behavior. Taken together, decision theory informs us that to model human decision makers in agents, we need agents that can be configured to act in accordance with utility functions and with biases and heuristics shaping the way they pursue utility.

## 2.2 Decision Making Styles

A second part of decision science of relevance to this work is the finding that people tend to exhibit patterns or tendencies in decision making. Individuals seem to hold a disposition or trait toward one or more types of general decision making strategies that have an impact on their performance on specific tasks. From the decision theory perspective, we may say that they are systematically expressing biases in their heuristic evaluations through specific decisions (Grigorenko and Sternberg, 1995) and that these biases are stable over time. The concept has been operationalized into a psychological instrument that has starting to see some use and attain evidence for reliability and validity (Loo, 2000; Scott and Bruce, 1995; Thunholm, 2004). This may suggest a potential relation between the notion of a player exhibiting a certain play style (Canossa and Drachen, 2009), as often claimed by player modeling (Yannakakis et al., 2013), and players exhibiting a decision making style. Or, put differently, a player's decision making style may be a partial determinant of a player's play style, as captured by player modeling. As such, modeling a player's decision making style may be understood as a sub-problem of the more general problem of player modeling.

If we can model the heuristic tendency of the player's system 1, and combine this with knowledge about a) the capacity/performance of the player's system 2 and b) contextual

knowledge about possible solutions for achieving the player's current goal in the game, we may be able to predict the player's decisions with some accuracy.

To the extent that we choose to model the player's decision making process in a generative agent after this double system, it may be reasonable to look for decision making algorithms that incorporate a combination of analytic and heuristic aspects, maintaining an abstract isomorphism to the process observed in human decision makers. We need agents that decide both analytically and heuristically, support stable biases, and are characterized by how they balance these two systems.

## 2.3 Bounded and Adaptive Rationality

A third, related, perspective on human decision making is given by the direction of psychology studying human rationality as being *bounded* and *adaptive*. Early aspects of this perspective even predate prospect and decision theory's contribution to the study of decision making in the work of Simon (1955). Later efforts, notably those of Gigerenzer and Selten (2002) and Gigerenzer and Gaissmaier (2011), have developed, to a certain extent, in dialog with decision theory.

The bounded and adaptive rationality perspective sees the human decision making process as taking place in an environmentally responsive and adaptive balance between analytic and heuristic processing. Proponents of the adaptive perspective on decision making will argue that human decision makers often operate with limited resources available for attention and cognition in constantly changing and partially unpredictable environments. As such, the application of biases and heuristics that in the perspective of decision theory may be seen as leading to cognitive fallacies, may in the adaptive perspective be seen as strategies for making decisions that are good enough, rather than optimal. In this perspective, the rational decision then becomes the decision that leads to an acceptable outcome under an acceptable amount of effort within an acceptable time-frame. This necessarily means that the motivation to make a good decision and the time available for decision making become important contextual parameters when evaluating decision making quality. Considered in the context of games, we may imagine that a decision may seem inconsequential or uninteresting to a player, even if the game rules emphasize the decision. Players playing irrationally or seemingly without skill may simply be unmotivated by individual decisions, the game as such, or distracted, and may hence be playing carelessly and devoting few resources to optimizing their decision making.

If we aim to simulate human decision making styles in games, our strategies for doing so should be able to scale the amount of information that players take into consideration and the amount of effort they dedicate to making optimal (as defined by the game rules) decisions. We need agents that care about different decision problems to different degrees and can represent different amounts of decision making effort.

## 2.4 Naturalistic and Recognition Primed Decision Making

Related to the notion of adaptive rationality and decision making under limited resources (such as time for reasoning or attention) is a perspective from naturalistic decision making studies: recognition primed decision making (Lipshitz et al., 2001; Klein, 1993; Flin, 1997; Ross et al., 2004). Recognition primed decision making is concerned with describing situations where experts decide within complex situations over short time spans.

Studies of naturalistic decision making in fire fighters, among other professions, have shown that expertly trained professionals may make decisions under stress by quickly iterating through all courses of action retrieved from memory in response to a particular perceived pattern and enact the first strategy that is deemed submissible, rather than comparing all submissible courses of action and selecting the best one. The first retrieved submissible course of action retrieved from memory will depend on the pattern in question and can be shaped through training. The shorter the time span for deciding, or in general the more limited the cognitive resources available for decision making, the more likely it is that expert decision makers will make recognition primed decisions. In the context of games this becomes relevant when we consider game mechanics that are common across games within particular genres or when players develop strong expertise within a single game. In e.g. a first person shooter game players may respond immediately to enemies by strafing or taking cover, without any rational thought preceding the behavior. Any player that has moved between superficially similar, but different, games such as from *Quake Live* (id Software, 2010) to *Counter-Strike* (Hidden Path Entertainment and Valve Corporation, 2012) will probably have experienced the effect of making recognition primed decisions out-of-context: While strafing is an effective way of counter-acting an enemy's ranged attack in *Quake Live*, due to the movement speed of the player and the low rate at which the player takes damage, the same behavior is ill-advised in *Counter-Strike*, where movement is slower and shots are far more deadly. Still, the superficial likeness of the two games may make the expert *Quake Live* player, who starts playing *Counter-Strike*, make these inappropriate tactical decisions until a new set of behaviors can be linked to the recognized patterns through training.

If we aim to simulate recognition primed decision making in artificial agents, it may make sense to have galleries of low-level ready-made plans that can be enacted reactively to particular patterns in the game state, in particular for games where decisions must be made quickly or where many decisions are made at the same time. We need agents that can act reflexively to situations perceived as familiar.

## 2.5 Somatic Markers of Decision Making

A final body of work in the field of decision science relevant to simulating human decision makers in games is the neuro-psychological work of Damasio that resulted in the somatic marker hypothesis (Damasio, 2008). Building on previous work in emotion and decision making, Damasio constructed experimental psychological paradigms incorporating psycho-physiological measurements. The somatic marker hypothesis views decision making as a partially conscious cognitive process supported by corporeally distributed heuristics experienced by the subject as “feelings” (Damasio, 2000). The work has demonstrated an integration between two separate, but functionally interacting neurological systems in the human brain: A fast emotional system that labels expected outcomes quickly, in parallel and mostly unconsciously, and a slower, but more detailed, conscious system that integrates and sometimes inhibits the output of the fast system. Damasio’s work compared patients with structural brain damage to the connections between the two systems to people with no brain damage. Important to this discussion is the finding that normal individuals exhibit traceable physiological signals when deciding under uncertainty and these signals match the experience of emotionally deciding based on feeling what the preferred decision is rather than solely analytically knowing (Bechara et al., 1997). The ideas of somatic markers of decisions and decision making styles map well onto the models of decision theory, adaptive decision making and recognition primed decision making. Since we know that decision making elicits measurable physiological responses in normal humans too (Ahn and Picard, 2006; Ahn, 2010), it is not far-fetched to imagine that physiological input devices would be able to provide indicators of what kind of decision making is taking place in human players, potentially informing future player models <sup>2</sup>.

---

<sup>2</sup>The papers included in the appendices of this thesis deal with recognizing emotional responses to games in human game players, but do not link these measurements to in-game decision making.



## 2.6 Chapter Summary

This chapter provided an introduction to three psychological perspectives on decision making: decision theory, adaptive and bounded rationality, and recognition primed decision making. From each perspective a number of implications for agent-based modeling of decision making were identified. Additionally, a dominant perspective on the link between emotion, physiology, and decision making in humans was given.

In the following chapter, we give a short introduction to the fields of player modeling and procedural content generation and relate them to the perspectives of decision science in order to build a theoretical foundation for a method for computationally modeling and simulating decision making in the form of agents.

## Chapter 3

# Player Modeling and Procedural Content Generation

In the previous chapter, we reviewed three dominant perspectives in psychological decision science that inform the procedural persona concept. As models and simulations of player decision making, procedural personas fall naturally within the field of player modeling. When applied to the generation and evaluation of content, they become tools for controlling procedural content generation, as computational critics (Osborn et al., 2013), linking player modeling and procedural content generation. In this chapter we provide a short overview of related work within player modeling and procedural content generation in order to situate the procedural persona concept.

### 3.1 Player Modeling

Player modeling is defined by Yannakakis et al. (2013, p. 1) as

...the study of computational models of players in games. This includes the detection, modeling, prediction and expression of human player characteristics which are manifested through cognitive, affective and behavioral patterns.

In their holistic overview of player modeling, Yannakakis et al. define four kinds of data sources that player models can be built from: game-play input, objective information, game context, and player profile information.

Game-play input describes the individual, atomic actions that the player uses to control the game or aggregations thereof; in the case of digital games what is sent through the

input devices of the platform running the game. From a decision making perspective game-play input is the behavioral implementation of player decisions.

Objective information describes the state of the player herself and can be obtained from many different modalities that provide indications of the cognitive and affective state of the player such as e.g. facial expressions, posture, heart rate, electrodermal activity, etc. As mentioned earlier, this cognitive and affective information may be directly relevant to understanding player decision making, as described by the somatic marker hypothesis, and may assist in measuring player's attention, engagement, risk perception, and importance attributed to decisions.

Game context describes the parametrized state of the game at given points in time. From the perspective of modeling decision making in games, this information naturally becomes crucial in order to understand what is being decided upon. As noted in the previous chapter, players' analytic effort and the heuristics they apply depend heavily on their perception of the decision making problem at hand which of course to a large degree is determined by the game state.

Player profile information describes external models of players derived from other contexts such as measures of personality, intelligence or any other source of information that describes individual differences between players along one or more dimensions. From the perspective of decision making in games it is quite likely that players of different personality types also exhibit different decision making styles as suggested by e.g. Canossa et al. (2015).

Based on the inputs, the purpose of player modeling is to define a computational model of the player that accepts some or all of these kinds of inputs and provide outputs that accurately describe individual players or groups of players.

Yannakakis et al. define four categories of player model outputs: scalar values, class selections, ordinal data representing preferences, or no output, when the results of a player model are used to classify a player as a specific type of player or cluster member.

The main intent of the procedural persona method is to combine a utility function with an agent control architecture, simulating a combination of analytic and heuristic decision processes to output data that can be used to simulate players.

Each combination of a utility function and a control architecture represents a particular player profile or more specifically a decision making style. In terms of applying a persona model, when the persona is used to simulate a player, game context information (in the form of game states) is combined with this player profile through the controller architecture to generate the output. The output of the model may be either scalar (e.g.

choosing coordinate locations for an attack or choosing a movement speed or target), ordinal (e.g. producing a ranked list of choices), or class values (e.g. choosing between different targets or abilities) depending on the game which the procedural persona is implemented for and how the game is controlled.

If personas are used to classify human players, mapping them to predefined personas, the models may function as classifiers of human players. As such, the personas may produce output in any of the four categories defined by Yannakakis et al., depending on the game in question and how the personas are applied. For any particular game, the personas may be limited to some of the four categories.

Before a persona model can be applied in the way described above, the utility function must be defined. In the papers included in this thesis the utility functions of personas are defined in two different ways:

The first way of defining the utility function is having game designers directly specifying the utility matrix based on their understanding of the game rules and their personal experience as game designers, testers, and players. This approach is applied in the majority of the papers included in the thesis and aims to represent a situation where a game designer wants to generate synthetic play-traces, but does not have any human play-test subjects available. In the terms of Yannakakis et al. this means that the game designer is providing the player profile directly by defining the utility matrix.

The second way of defining the utility function in procedural personas explored in this thesis is learning it from human play-traces, or in the terminology of Yannakakis et al., a combination of game-play input and game context information. This approach is explored in Chapter 9, where agents are trained directly on human play traces and compared to personas defined by game designers.

These two ways of defining utility functions reflect a central dichotomy in player modeling in general: Player models tend to be dominated by either the influence of a priori information (theory) or the influence of observed information (data). The first kind is termed *model-based* approaches while the second kind is termed *model-free* approaches.

The procedural persona method, as envisioned in this thesis, primarily belongs to the model-based category, but Chapter 13 will suggest how procedural personas may be implemented in ways that draw on methods from both categories.

In the next chapter, we explore how to enable game designers to describe the utility functions of procedural personas by defining decision spaces within games.

Before proceeding, however, we will briefly review the topic of procedural content generation in games, which is an application of procedural personas explored in Chapter 10.

## 3.2 Procedural Content Generation in Games

Procedural content generation in games is an approach to providing large amounts of content in games while reducing either the amount of human effort necessary to produce the content or the amount of storage required to distribute the content. As a method, procedural content generation has been used almost since the beginning of digital games with the dungeon-crawling game *Rogue* (Toy et al., 1980) and the space trading game *Elite* (Braben and Bell, 1984) being well known early examples. Recently, procedural content generation has seen a strong interest as a method in commercial games and as a research topic within game artificial intelligence.

Togelius et al. (2011, p. 1) define procedural content generation as:

...creating game content automatically, through algorithmic means ... the term game content refers to all aspects of the game that affect gameplay other than non-player character (NPC) behaviour and the game engine itself. This set includes such aspects as terrain, maps, levels, stories, dialogue, quests, characters, rulesets, dynamics and weapons.

Developing a taxonomy of methods for procedural content generation, Togelius et al. identify three different methods of procedural content generation that each represent a different approach to controlling the quality of the generated content: constructive, generate and test, and search-based procedural content generation.

Constructive procedural content generation creates content through a linear process and does not evaluate or adjust the output. The method always assumes that whatever is produced from it will fall within the expected and acceptable range of outputs; the method itself guarantees that the generated content is sufficient, so to speak. Synthetic *testing* via procedural personas has little relevance for procedural content generation systems built around this method, since play-testing is moot for content that is guaranteed to be acceptable. However, procedural personas might be relevant for such a process as a step in the constructive process, for instance decorating a level with the consequences of simulated actions for the human player to experience. Synthetic play-traces might also be relevant if the designer wished to visualize potential strategies for solving levels generated by constructive processes. Procedural personas are not applied to games featuring constructive procedural content generation processes in the work presented here, but doing so is a potential avenue for future work.

Generate-and-test methods for procedural content generation take a slightly different approach than constructive methods. Rather than guaranteeing that the initial generation

of a piece of content, e.g. a level, always ends with an acceptable results, generate-and-test approaches evaluates the content against a number of criteria. If the content satisfies these criteria, it is accepted; otherwise it is discarded and a new attempt is made at generating something acceptable. Here, procedural personas may be useful in evaluating the generated content, either as the only evaluating mechanism or, more realistically, in conjunction with other evaluative mechanism. For instance, a requirement could be that a procedural persona with a particular decision making style should always be able to complete levels generated by a constructive system, should be able to complete them with some predefined probability, or should be able to attain some level of utility as defined by the persona's utility function. Personas can function as evaluators of whether a level is playable at all, or they may be able to assign a level a particular score, based on how e.g. much utility they derive from playing it. In this sense they may function as what Osborn et al. (2013) term computational critics - stand-ins for human evaluators of content. The immediate down-side to generate-and-test methods for procedural content generation is that they may be wasteful if it is hard or unlikely to generate content that satisfies the criteria. In that case it may be advantageous to use procedural content generation methods that do not follow this simple loop of producing and discarding content, but instead perform a directed search within the multi-dimensional space that defines the possible solutions to the content generation problem at hand.

The third approach, search-based procedural content generation, does exactly this, searching through potential content configurations in a directed manner, along the way evaluating which changes are most likely to generate better content. For search-based procedural content generation procedural personas can take on the same role as computational critics that they may take on in generate-and-test solutions, but in this case their use becomes much more efficient. They can be integrated directly into the content evaluations that guide the search through content space. Using procedural personas in this manner is demonstrated in Chapter 10, where procedural personas with differing utility functions are used to control search-based procedural generation of levels for the test-bed game *MiniDungeons 1*, through the application of evolutionary computation.

### 3.3 Chapter Summary

Until this point, we have discussed the utility functions and decisions of procedural personas in an abstract sense, only relating them to decision theory and reviewing how models of utility functions embedded in agents may be useful for player modeling and procedural content generation in games. In the following chapter, we become specific about how specific utility functions of procedural personas may be defined for specific

games with an outset in the play persona theory that procedural personas are built on top of. Additionally, we define a number of metrics that we will use to measure how well personas represent human player decision making and to measure differences between individual personas.

## Chapter 4

# Decision Making in Games

In order to model and simulate human decision making in games we must consider how to operationalize player decision making, defining measurable indicators of decisions. We should conduct this operationalization while taking into account the decision science related in the previous chapters and the derived requirements to decision making simulating agents. In this chapter, we address this problem by suggesting how elements of and occurrences in games can be used to infer decision making from behavior.

First, we review the concept of play personas which forms the final part of the theoretical underpinnings of the procedural persona concept and enables the us to propose an operationalization of decision making. Secondly, we review the concept of affordances and its relation to play personas. Thirdly, we link play personas and affordances to decision making and examine the procedural persona concept in detail. We define three metrics that we use to compare individual decision makers in games. We will later use these metrics to compare procedural personas to one another and evaluate how close procedural personas come to representing human players' decision making styles. Finally, we describe the agent control methods used to implement procedural personas in this thesis.

### 4.1 Play Personas and Procedural Personas

The concept of personas was first adapted to the domain of (digital) games under the headline of *play personas* by Canossa and Drachen (2009, p. 3) who define play personas as



“clusters of preferential interaction (*what*) and navigation (*where*) attitudes, temporally expressed (*when*), that coalesce around different kinds of inscribed affordances in the artefacts provided by game designers.”

Their work focuses on how assumptions about such player preferences can be used as metaphors for imagined player behavior during the design process, and how patterns in observed player behavior can be used to form lenses on the game’s design during play-testing.

Applying the perspective of decision making can further narrow the play-persona concept. Rather than considering any arbitrary reasons for player preferences valid, focusing on decision making provides a perspective to operationalize the backgrounds for preferences into combinations of affordances (Gibson, 1977) and utilities. In this perspective, any preference in a game, expressed through behavioral interaction with the game’s affordances, is either the result of decision making or the result of noise which falls outside of the explanatory power of the theoretical framework. This should allow us to explain less of the behavior in any given game, but allow us to provide stronger explanations of the behavior that we *can* explain.

The play persona concept is based on the notion of games being artifacts with affordances inscribed into them by game designers. Since procedural personas are built on play persona theory, this warrants a closer look at affordances in games: For any spatio-temporal configuration of a given game, a limited number of plausible affordances can be determined using information about the game mechanics and reward structures by analyzing and interpreting the games. Based on these affordances, different hypothetical combinations of utilities can be used to create metaphors for typical player behavior. Each combination then represents a utility function describing the motivations of a persona. To the extent that these metaphors match what actual human players decided, they can be considered lenses on the players’ decision making styles with utilities explaining how player preferences are distributed between the available affordances. In this thesis, we attempt to extend the play-persona concept into game playing procedural personas, by building generative models of player decision making from either designer metaphors, actual play data, or combinations of the two. As such, we need to identify a way of connecting affordances and utility to decision making.

Our solution is to operationalize decisions as actions relating to the affordances presented in a particular game. In this context, we define affordances as selected game mechanics accessible to the player. By focusing on the affordances of the game we abstract away the particular inputs of the player and only consider behavior in relation to the structured context that the game provides; the inscribed affordances. In this operationalization *a*

*decision has occurred when a player interacts with a selected affordance.* Put differently, we use expert game design knowledge about what constitutes meaningful interactions in the game to a priori define what constitutes decisions in relation to the game rules. This reduces the game's state space into a smaller *decision space* in which we can conduct our player modeling and simulation.

We must then reduce player motivations and actions onto this same decision space in order to connect the affordances of the game with player preferences, understood through utility functions. This means that we can only account for player motivations to the extent that they are directed at the affordances that we included in the decision space; only these affordances can provide utility to the player. If a player derives utility from the game by sources that are not included in our selected list of affordances, we cannot explain this in our framework, but must consider it noise or an idiosyncratic bias. This also means that this decision making perspective will never be capable of *explaining* all player behavior, though it may be able to *reproduce* it, by incorporating idiosyncratic player biases. We effectively split the sources of player behavior into two categories: Behavior derived from decision making related to the affordances of the game and behavior derived from other sources. The first category we can operationalize and analyze by way of our a priori defined affordances as sources of utility, while we will refrain from explaining the other part and simply operationalize this as the aspects of player behavior that we cannot explain, but only describe and include it as bias towards certain kinds of behavior or noise that influences the player's movement through decision space.

The link between the decision space, identified by way of the affordances, and player motivations can then be forged by encoding the relative value of interacting with each particular affordance in a *utility matrix* for the individual player. The weights of this utility matrix will describe the player's preferences with respect to the affordances included as sources of utility. Learning a player's decision making style, within the a priori defined decision space of the game, becomes a problem of learning these weights as accurately as possible. The advantage of the approach is that it allows us make specific predictions about what decisions players with specific utility matrix configurations will make in specific situations. The disadvantage is that we cannot account for reasons for behavior that we have not included a priori in our list of affordances that make up the dimensions of the utility matrix.

In order to construct this decision space we must identify the sources of affordance in the game. In the following section, we outline a strategy for accomplishing this.

## 4.2 Identifying Decisions in Games

The task of identifying relevant game mechanics for defining decision affordances in games is not necessarily a simple problem. At the most detailed level any game will arguably have a set of decision affordances unique to that particular game, but games share configurations of decision affordances within genres or other groupings. Ideally, an approach for modeling decision making in games should provide a generalizable method for this. This thesis does not address this question as it only treats three particular games and identifies decision affordances in each game in an ad hoc manner. However, work by Cowley et al. (2009) has demonstrated ways of applying game design patterns (Björk and Holopainen, 2004) to categorize behavioral patterns in games that could be construed as decision making affordances at the individual action level. Here, we define decision affordances individually for the three games in which we study decision making, and instead go into detail with regards to the levels of abstraction on which decision making takes place.

When considering players' interactions with a game's affordances, one approach could be to record and examine every instance of input to the game that the player delivers. We have already established that some decision making is not directly rational and cannot be expected to happen in direct pursuit of game goals. This may make some input seem random or misdirected in relation to the formal context of the game rules. Additionally, research by Canossa and Cheong (2011) indicates that it may be naive to even consider all inputs to a game as intended, willful, or goal-directed. Canossa and Cheong classify actions, and thereby inputs, as being the consequence of either intentional, trained or chaotic behavior, which we may translate roughly into planned, reactive or random behavior, respectively.

The three kinds of actions can be interpreted from the three decision theoretic frameworks we reviewed in the previous chapter, decision theory, bounded rationality, and recognition primed decision making.

Random behavior cannot be related to the game rules and must be considered either noise or a bias in player's decision making. Any model we choose to use for the human player's decision making should be capable of treating these facets of the in-game actions by modeling behaviors that are inexplicable in relation to the affordances as noise or biases.

Reactive behavior can be expected to occur when players respond with rote learned routines to in-game events, such as automatically jumping in front of an enemy in a platform game or firing at an enemy that appears in a first person shooter game. Relating reactive behavior to decision making, it can be understood as decision making consisting

only of heuristics: The player perceives a known pattern in the game and responds with the most appropriate general or specialized heuristic in her repertoire. This kind of behavioral response is best explained by recognition primed decision theory, which describes how the human mind maps perceivable patterns to predetermined immediately executable behavioral scripts that leave little to no room for conscious modification or planning (Klein, 1993). As such, reactive behavior can be understood as decisions that are consistently enacted in response to particular patterns in the game state. A player's expertise and motivation for the game in general or for a particular aspect of a game should be expected to influence to which extent decisions are made reactively, using very few cognitive resources, or are made in a planned, rational manner.

Planned behavior, can be explained as the result of intentional, rational decision making as described by decision theory. We should expect planned behavior to be describable and intelligible in relation to personal utility functions that players try to optimize their decision and behavior in accordance with. When a player is motivated toward a game and is awarded an unlimited amount of time and cognitive resources for making a decision, we should expect the decision making process to be describable in accordance with decision theory. We should expect the player though optimize rationally, though still having her decisions shaped by biases and still relying, to some extent, on heuristics that may or may not be appropriate depending on the player's level of training and expertise. The degree to which player relies on heuristics can be understood as a function of the player's motivation and resources in accordance with the bounded rationality perspective.

Few digital games contain only decision making challenges players can solve using only rational or reactive decision making. Rather, in many games players must rely on a combination of manual skill, memory capacity, and reasoning abilities simultaneously while pursuing several, often conflicting, goals on different time scales and often in real time. This, arguably, is part of what makes digital games an appealing medium full of interesting decisions.

This means that an analytical framework for understanding how a game affords decisions through mechanics should help us make sense of decisions and actions of multiple kinds occurring simultaneously. It also means that a meaningful in-game behavior, such as opening a door, engaging with a particular enemy, or choosing a particular inventory item, is unlikely to indicate just one kind of decision or one kind of action, but rather is the result of several kinds of decisions and actions. Assuming that we cannot tease apart these constituents through interactions with affordances, but must look at them in their integrated form, we suggest that we instead look at them as different levels of abstraction and grouping in relation to the game rules.

In the following section we identify three possible levels of abstraction for considering decision making in relation to affordances in games. We propose these three levels as perspectives for operationalizing player behavior into measurable indicators of decisions.

### 4.3 Three Pragmatic Levels of Decision Making in Games

Our first step of identifying what constitutes meaningful decisions is to locate and select decision making affordances in games. Once located, we examine interaction with these decision indicating affordances in three different ways: 1) individually, 2) in terms of the order in which they appear, and 3) in terms of their aggregate occurrence frequencies across delineated windows of game-play events e.g. levels. We call these three levels the *action level*, the *tactical level*, and the *strategic level*. In the papers included in this thesis, we eventually pursue all three of these levels, building generative agents that try to simulate human decision making styles at these three levels, respectively.

Action level decisions are considered atomically and are not related to other action level decisions that went before or come after, only the immediate game state in which they occurred. They are operationalized as the simplest meaningful effects of a player interaction with a mechanic. Typical examples would be moving a player character or utilizing a particular attack or weapon in a specific situation.

Tactical level decisions are operationalized as the interaction with higher level challenges or objectives in the game: Fighting an individual foe, reaching a particular location, or or acquiring a particular object. They typically encompass a number of action level decisions and typically require planning to organize series of action level decisions in an order leading to the defined interaction. Tactical level decisions describe ordered relations between affordances. Given the latest affordance that the player chose to interact with what will the next affordance be? The individual action level decisions leading to the tactical outcomes are not considered directly, so two tactical decisions are considered equivalent even if different series of action level decisions were made to realize them.

Strategic level decisions are the most abstract operationalization and describe the frequency distribution of the action and/or tactical decisions over the course of a predetermined section of a game. How often was each affordance interacted with in the particular level. This measure of decision making does not consider the order of the decisions, but simply observes how often they occur across the entire window of game-play events. E.g. how many monsters were fought in a level, how many attacks of which kinds were used, or how many points were acquired. They are similar to the metrics typically used in

game analytics to characterize play sessions across e.g. levels or maps (El-Nasr et al., 2013).

In the following section, we complete our operationalization of decision making by defining three metrics of decision similarity.

## 4.4 Metrics of Decision Similarity

In this section, we present the three different simulation based metrics used to evaluate how alike personas are in their decision making and how well they represent human decision makers. Each metric was constructed to capture interactions with affordances at each of the three different levels of decision making operationalization: the action level, the tactical level, and the strategic level.

### 4.4.1 Action Agreement Ratio

The first metric used to evaluate agent to human likeness is the *action agreement ratio* (AAR). AAR considers each step of a human play-trace a distinct decision. To produce the AAR between an agent and a human player, all distinct game states of the human play-traces are reconstructed. For each game state, the agent being tested is inserted into the game state and queried for the next preferred action, essentially asking: “What would you do?”. If the action is the same as the actual next human action, the agent is awarded one point. Finally, the AAR is computed by dividing the points with the number of decisions in the human play-trace. As such, a perfect AAR score of 1.0 represents an agent that for every situation in the player’s play trace decided to take exactly the same action as the player did. Figure 4.1 contains an illustration of how the AAR is produced, and Algorithm 1 describes each step in pseudo-code.

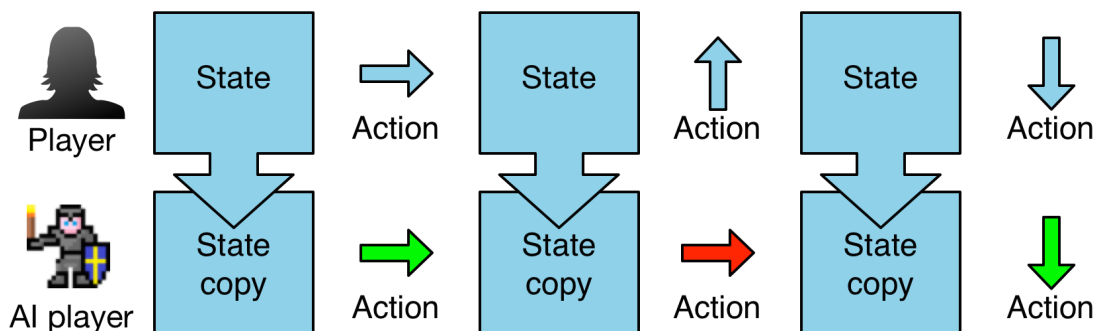


FIGURE 4.1: An illustration of how the Action Agreement Ratio is calculated.

---

**Algorithm 1:** Algorithm for calculating the AAR metric.

---

**input** : A play trace consisting of a vector  $G$  of original (player or agent) game states of length  $n$ .

**input** : A corresponding vector  $A$  of original actions, also of length  $n$ .

**input** : A template for a procedural persona  $P$ .

**output:** A scalar  $AAR$  representing the correlation between the  $D$  and a vector of generated agent decisions of equal length.

$i = 0$ ;

$hits = 0$ ;

**while**  $i < n$  **do**

```

    gameclone = clone( $G_i$ );
    procedural personas instance  $p = \text{new } P$ ;
    set game state of  $p$  to  $gameclone$ ;
    record next action  $a_p$  from  $p$ ;
    if  $a_p == A_i$  then
        |  $hits ++$ ;
    |  $i ++$ ;

```

$AAR = hits/n$ ;

return  $AAR$ ;

---

#### 4.4.2 Tactical Agreement Ratio

The second metric used for evaluating the likeness between agents and humans is the *tactical agreement ratio* (TAR). TAR does not necessarily consider individual actions decisions. It only considers reaching each distinct affordance in the level a significant decision, and may ignore individual actions in between. For each affordance reached in the human play-trace, the resulting game state is reconstructed and the agent being tested is inserted into the game state. The agent is then allowed as many actions as necessary to reach the next affordance, asking the question “What affordance would you go for next?” at the tactical level. If the next encountered affordance, in terms of both type and location, matches the actual next human one exactly, the agent is awarded a point. Finally, the TAR is computed by dividing the points with the number of affordances reached in the human play-trace. As such, a perfect TAR score of 1.0 represents an agent that visits every affordance visited by the player in the same order as the player originally did. Figure 4.2 contains an illustration of how the AAR is produced, and Algorithm 2 describes each step in pseudo-code.

#### 4.4.3 Strategic Agreement Ratio

The third metric used for evaluating the likeness between agents and humans is the *strategic agreement ratio* (SAR). Operating at the general and aggregate level, SAR considers the total amount of affordances engaged with for each level. For each affordance

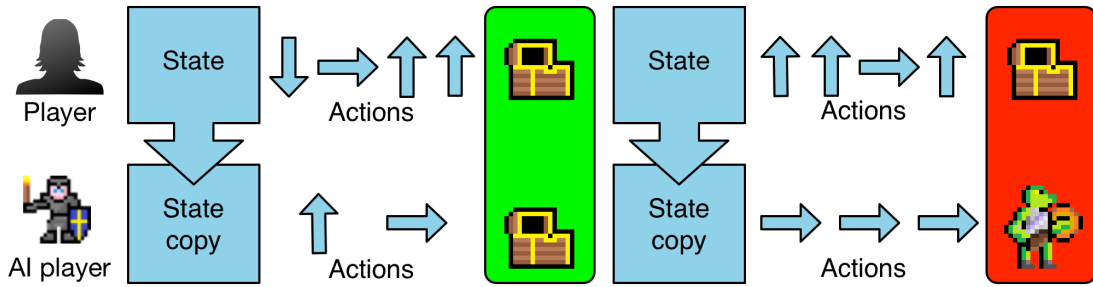


FIGURE 4.2: An illustration of how the Tactical Agreement Ratio is calculated.

---

**Algorithm 2:** Algorithm for calculating the TAR metric.

---

**input** : An original game state  $g$  (from a player or agent session).

**input** : An vector  $T$  of length  $m$  containing affordances encountered in the original game.

**input** : A template for a procedural persona  $P$ .

**output:** A scalar  $TAR$  representing the correlation between the affordance encounters in  $T$  and a vector of generated agent affordance encounters of equal length.

$j = 0;$

$hits = 0;$

**while**  $j < m$  **do**

$gameclone = clone(G_j);$

    procedural personas instance  $p = new P;$

    set game state of  $p$  to  $gameclone;$

**while**  $p$  has not encountered an affordance  $p_t$  in  $gameclone$  **do**

        get next action  $p_a$  from  $p;$

        apply  $p_a$  to  $gameclone;$

    record next affordance  $p_t$  encountered by  $p$  in  $gameclone;$

**if**  $p_t == T_j$  **then**

$hits ++;$

$j ++;$

$TAR = hits/m;$

return  $TAR;$

---

the absolute difference between the agent's measure and the player's interaction counts is calculated and normalized by the maximal possible number for the level or, in the case of the number of affordances with no natural upper limit, in relation to a constant. These statistics are then summed, divided by the number of statistics (in this case five). The score, which is an expression of how different the agent's statistics are from the player's, is subtracted from 1.0 to produce the SAR. As such, a perfect SAR score of 1.0 would indicate an agent that e.g. fought exactly the same number of monsters, collected exactly the same number of treasures, died in combat or exited the level just like the player, and did so in exactly the same number of actions. In other words, the SAR asks the question "How often would you go for each affordance in this level?". Figure 4.3 contains an illustration of how the AAR is produced, and Algorithm 3 describes each



step in pseudo-code.

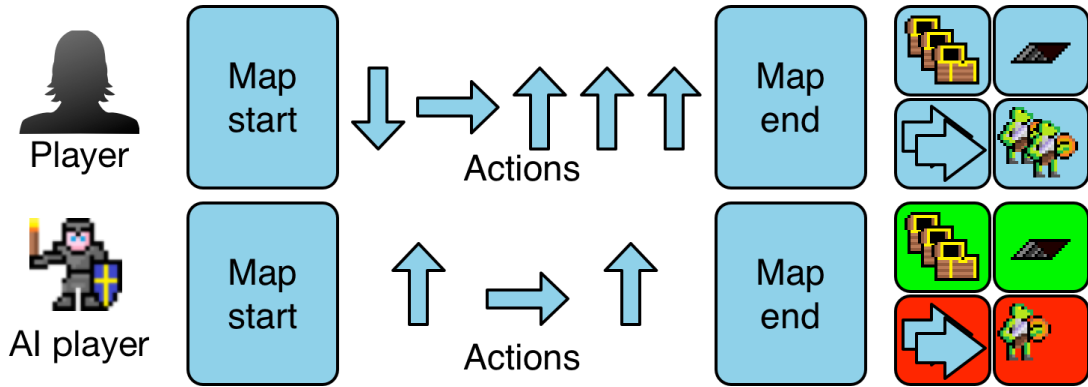


FIGURE 4.3: An illustration of how the Strategic Agreement Ratio is calculated.

---

**Algorithm 3:** Algorithm for calculating the SAR metric.

---

**input** : An original game state  $g$  (from a player or agent session).

**input** : A list of selected affordance types  $Q$  of length  $o$ .

**input** : A vector  $S$  of affordance interaction frequency counts, also of length  $o$ .

**input** : A template for a procedural persona  $P$ .

**output:** A scalar  $SAR$  representing the correlation between the affordance encounter frequencies in  $S$  and a corresponding vector generated by an agent.

$gameclone = clone(g)$ ;

procedural personas instance  $p = new P$ ;

set game state of  $p$  to  $gameclone$ ;

$gameover = false$ ;

$T = new$  vector of length  $o$ ;

**while**  $!gameover$  **do**

apply next action  $a_p$  of  $p$  to  $gameclone$ ;

**for**  $k = 0; k < o; k++$  **do**

**if**  $p$  interacted with affordance of type  $Q_k$  **then**

$T_k++ = 1$ ;

**if**  $dead(p) || won(p)$  **then**

$gameover = true$ ;

$SAR = 0$ ;

**for**  $l = 0; l++ ; l < o$  **do**

$SAR++ = T_l / S_L$ ;

$SAR = SAR / o$ ;

$SAR = 1 - SAR$ ;

return  $SAR$ ;

---

In the following section, we describe which agent control methods we used for implementing procedural personas which were evaluated using the three metrics described above.

## 4.5 Agent-Control Methods for Procedural Personas

To construct the procedural personas described in this chapter, we need to choose a specific agent implementation that forms the generative core of our decision maker model, our persona(s). In this and previous chapters, we identified a number of ideal agent qualities that would be useful for decision modeling which can be summarized in the following list of requirements:

- Agents should support a utility function, describing how they value different events, affordances, in the game.
- Agents should be able to represent both the analytic and heuristic steps in human decision making.
- Agents should be configurable in how much of their decision making is based on analysis or search and how much is based on heuristics.

Commercial and academic game artificial intelligence offer a large gallery of examples and methods to choose from for agent implementations: ranging from finite state machines and behavior trees to neural-networks trained by back-propagation and neuroevolution, to name a few examples (Rabin, 2004; Mikkulainen et al., 2006; Risi and Togelius, 2014; Yannakakis and Togelius, 2014). Some commercial implementations, when used to produce e.g. AI opponents with varying play styles as in the *Civilization* (Firaxis Games, 2010) games, have even been used to produce agents that may even be thought of as personas, exhibiting different decision making styles. The literature, notably Mark (2009) also describes the use of utility functions for generating variation in agent behavior.

In spite of this, agents in the literature that are modeled with different play styles typically play *with* the player, but not *as* different players. They are often - as e.g. in the case of the AI players in *Civilization* - given access to information or other advantages and are crafted for providing an enjoyable game experience rather than an accurate description of possible or typical player play styles. The procedural personas, in contrast, are crafted for playing *as* players that are given the same information and action options that human players would receive. Still, many of the agent control methods in the literature are useful for and adaptable to the procedural persona concept. In the papers of this thesis, four particular methods are appropriated:

**A\* Search in State Space** is used in the first paper in Chapter 6 to create a single agent that becomes a reference process representing a decision making style. The A\* search algorithm operates with a combination of search and heuristics and easily incorporates the notion of utility in its cost function and heuristic. The drawback of using A\* search in state space is that both the cost function and heuristic, typically, are defined and tweaked by the algorithm designer. Each affordance of the game must be considered in detail and added to the cost and heuristic functions specifically, making adaptation to games with many affordances time consuming.

**Q-Learning** is used in the second paper in Chapter 7 to learn how to satisfy different utility functions for particular levels. While the method, as implemented here, is shown to perform well, it is limited by being time-consuming to train and by not being able to generalize to new game content.

**Neuro-evolution** is employed in Chapters 8 to 10 and is the most extensively applied method in this thesis. Here, it is applied in a simple variant, where the topology of the controller network is kept static and only weights of individuals are evolved. While the method still requires off-line training it is shown to generalize gracefully to unseen content and utility functions may be converted directly into fitness functions. This means that personas based on neuroevolution can be used to play content as it is being created and changed, while producing consistent results.

**Monte-Carlo Tree Search** is used in the final paper of the thesis, Chapter 11. This method also support utility functions in a straight forward manner. Furthermore, it has the advantage of not requiring off-line training and being highly generalizable to new content. However, the method is shown to potentially require some manual adaptation, e.g. in the form of guided roll-outs, to perform optimally for particular games.

Each method comes with its own set of advantages and disadvantages. The implementations of the four agent control methods are described and evaluated in detail in the respective papers. In Chapter 13, other promising agent control methods that could be combined with the procedural persona concept in future work are described.

## 4.6 Chapter Summary

In this chapter we reviewed the theory of play personas, a framework for prescribing and describing player behavior in games, useful for game design and game analytics.

Extending the play persona concept, we defined the concept of procedural personas which encode play personas as game playing agents by modeling players as decision makers: observed in-game or imagined by the designer.

We defined three analytical levels for understanding decision making in games: the action level, the tactical level, and the strategic level. We argued that these levels have different meanings for different games as they depend on the game rules. Therefore, they must be defined for the individual game before procedural personas can be constructed and their definitions are highly dependent on the affordances of the game in question. We described three simulation based metrics for measuring the likeness between players and personas, one for each analytical level. Finally, we contextualized and described the AI agent control methods used to drive the behavior of procedural personas in the papers of this thesis.

In the following chapter we present the three game domains that were used to explore and develop the procedural persona concept in this thesis.

# Chapter 5

## Domains

In this chapter we introduce the three domains that are used for the decision modeling experiments in this thesis: *The Mario AI Benchmark Framework*, *MiniDungeons 1*, and *MiniDungeons 2*.

### 5.1 The Mario AI Benchmark

*The Mario AI Benchmark* (Karakovskiy and Togelius, 2012) is a well established testbed for working with procedural content generation, player modeling, agent development and testing, and human imitation in platform games. It is a replication of the game mechanics and content of the classic 2D platform game series *Super Mario Brothers* (Miyamoto and Tezuka, 1985). The testbed has the advantage of being immediately familiar to most players in terms of content and mechanics. The object of the game is to move the player character, Mario, towards the right of the screen while avoiding a number of obstacles and avoiding or killing a number of enemies. The game has simple keyboard inputs consisting of buttons mapped to six different actions: *Down*, *Jump*, *Left*, *Right*, *Speed*, and *Up*.

The game is played in a real-time (25 fps) simulation with an experientially continuous deterministic world with predictable physics. It allows for a large number of unique positions and action sequences, ensuring that any two sessions played by humans are unlikely to ever be completely identical.

The game rules that the player is afforded to comply with are simple: Players should win levels as quickly as possible, while avoiding or killing enemies in their way, and collecting as many items as possible.

At every given moment during the course of a play-through, only a subset of the level is visible to the player, providing no more information than can be assimilated at any given moment by a perceptually and cognitively normally functioning individual.

The game presents most *relevant* information about the game state at any given time and can, as such, be considered a game rich in information (though not perfect, as blocks may contain various hidden power-ups and enemies can be hidden from sight for varying amounts of time). Figure 5.1 shows an example screen-shot from the *Mario AI Benchmark Framework* along with a number of positions that the player character could hypothetically occupy and some potential paths between those positions.

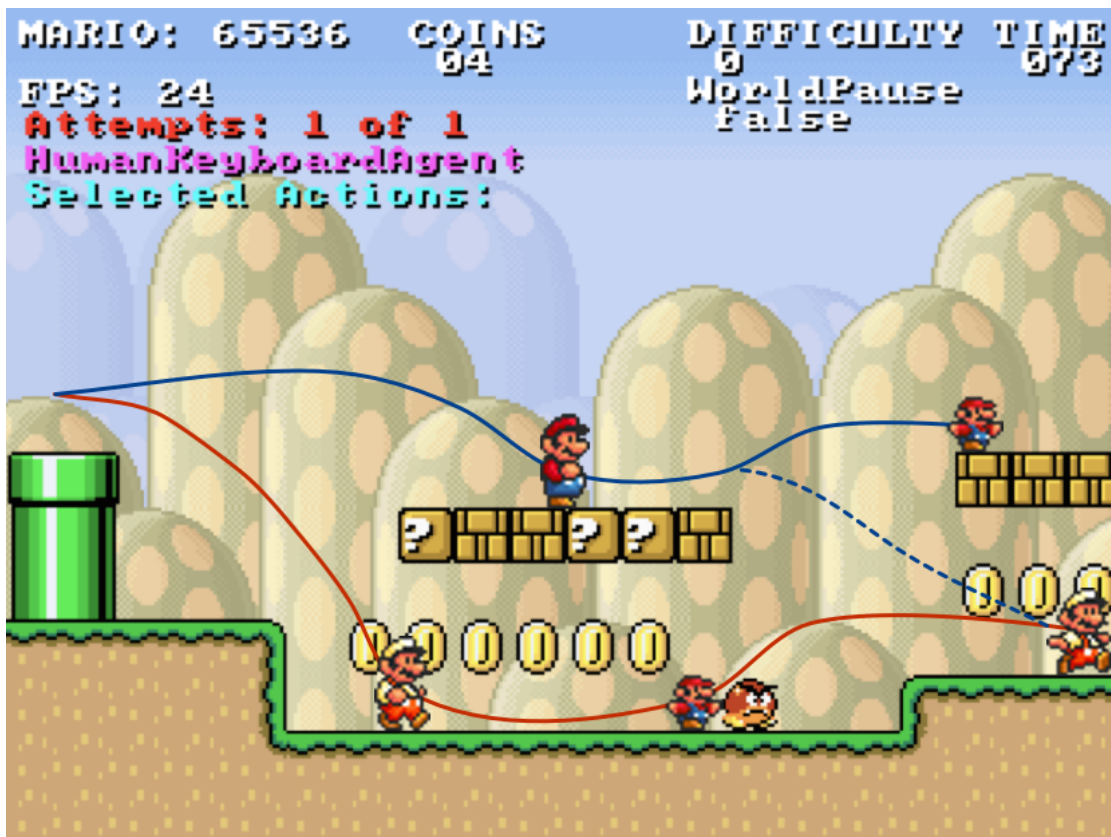


FIGURE 5.1: A screen-shot from the Mario AI Benchmark Framework, showing a few different potential paths from the left-hand side of the screen to the right-hand side.

The *Mario AI Benchmark* is included in the first paper of this thesis as an initial examination of the problem of learning about human decision making styles by comparing them to synthetic agent decision making styles. The game is investigated at the action level by comparing human play-traces to agent play-traces and using the individual players' deviations from the agent play trace to characterize the players. While the game formed a useful initial context for developing the notion of using agents as reference processes for understanding human players, the framework is not used in the subsequent papers which instead focus on the *MiniDungeons* games. This is primarily due to the

fact that the high frequency decision making in the *Mario AI Benchmark* makes it hard to construct a simple decision space for the game. Moreover, the fact that the game relies heavily on trained manual skill makes it well suited for studying recognition primed decision making, but perhaps less well suited for studying tactical or strategic decision making. As the first study finds, some players will be characterized by their (lack of) skill alone. While this is interesting in itself, it may be disturbing for the purpose of studying decision making styles in games.

Instead, the rest of the papers in this thesis use the *MiniDungeons* games as test-beds for studying decision making styles. These games were specifically developed to require little to no manual skill allowing us to focus on the expression of decision making styles through behavior, rather than the expression of manual skill. In their design, the games were inspired by classic dungeon crawling games featuring procedural content generation, such as the original *Rogue* (Toy et al., 1980), but also modern turn-based dungeon crawling games emphasizing decision making were a source of inspiration, especially *868-HACK* (Brough, 2013) and *Hoplite* (Cowley et al., 2013).

## 5.2 MiniDungeons 1

*MiniDungeons 1* is the first game included in this thesis developed specifically for the purpose of studying and simulating human decision making (Holmgård et al., 2014b) and tries to solve some of the challenges identified in the work with the *Mario AI Benchmark*. Implemented in *Java* (Gosling et al., 2011) using the *Processing* framework (Reas and Fry, 2001), the game is a simple turn-based rogue-like puzzle game, played in a web browser or as a stand-alone application, and is controlled using only the arrow keys of the keyboard. There is no time pressure in the game and the player is free to spend as much time deciding between each action as she wants. These design aspects largely eliminate the problems identified in the *Mario AI Benchmark*, removing the influence of manual skill on the decisions that players make in the game.

*MiniDungeons 1* levels are laid out on a grid of  $12 \times 12$  tiles: tiles can be walls (which obstruct movement), empty, or contain monsters, treasure, the level entrance or exit. In *MiniDungeons 1*, a hero (controlled by the player) starts at the level entrance and must proceed to the level exit: stepping on the exit tile concludes a level and loads the next one. A hero starts each level with 40 hit points (HP) and dies at 0 HP. The hero can collect treasure by stepping on treasure tiles: treasures have no in-game effect but a treasure counter is shown on the user interface. The hero can drink potions by stepping on potion tiles: potions heal 10 HP, up to the maximum of 40 HP. The hero can kill monsters by stepping on monster tiles: monsters do not move and only engage the hero if

the hero moves onto their tile. Combat is stochastic: a monster deals a random number between 5 HP and 14 HP of damage to the hero and then dies. This means that the player has full information about the level except for monsters' damage. As such, the main challenge in *MiniDungeons 1* becomes to find a plausible route to the main affordance, the exit. At the same time, the player must decide which auxiliary affordances to pursue: how many additional monsters to kill and treasures to collect. The player quickly learns the monster damage distribution (a uniform distribution in the range 5 to 14 HP, both inclusive) and the extent to which the player decides to pursue each affordance is then assumed to provide information about how much utility each affordance provides to the player. A screen-shot from *MiniDungeons 1* is shown in Figure 5.2. Differences in affordance interactions are then assumed to represent differences in utility functions and per extension differences in decision making styles. The relatively small size of the levels and the fact that they are discretized spaces produce a high decision density. Even a single action, such as moving to an adjoining empty tile, significantly changes the game state in terms of remaining steps to the exit, monsters, potions, and treasures. This means that any input that significantly changes the game state entails a specific decision. The small number of affordances in this test-bed limits the number of utility sources that must be considered when constructing an agent-persona. Finally, the small level size means that most levels can be completed relatively quickly.

In a number of the papers included in this thesis, we demonstrate how a game as seemingly simple as *MiniDungeons 1* contains enough interesting decisions to allow us to observe a variety of decision making styles and simulate these using a gallery of different personas. The *MiniDungeons 1* game is well suited to investigate how affordances, utility functions, and risk interact, and the game is fundamentally about making decisions under risk. However, we found that the game is not complex enough to investigate the concept of adaptive and bounded rationality. Since the player character is the only one that moves and changes in *MiniDungeons 1*, it becomes relatively easy for the player to anticipate future game states. The game presents no antagonists to the player, and does not give the impression of working against the player, but simply presents a number of gambles that the player may choose to engage with or not. In order to address this other aspect of decision making, we designed and implemented the game *MiniDungeons 2*, which we describe in the following section.

### 5.3 MiniDungeons 2

*MiniDungeons 2* is also a turn-based rogue-like game, implemented in *Mono* (Xamarin, 2004) using the *Unity* (Unity Technologies, 2005) game engine. It retains a number of the





FIGURE 5.2: A level from *MiniDungeons 1*. The only moving character is the player's. The main challenge is to decide what affordances to pursue in the level under the risk that each monster may deal from 5 to 14 HP worth of damage.

design features of *MiniDungeons 1*. It is playable on smart-phones and web browsers in order to reach as many potential players as possible and requires little to no manual skill to play. Every decision the player makes in *MiniDungeons 2* has a significant impact on the game state meaning that every decision counts, bringing the game's state space and the decision space in close alignment. Levels in *MiniDungeons 2* may be solved in many different ways, in an attempt to support variety in decision making styles. Crucially, a complete game tree for a single level of *MiniDungeons 2* is difficult to simulate mentally, enticing players to conduct some aspects of decision making through analytic thinking

and other aspects through heuristic thinking.

Levels in *MiniDungeons 2* have a slightly different format from those in *MiniDungeons 1*, owing to the game being optimized for smart-phones. The tile based levels are sized 10 by 20 and tiles are either *Walls* or *Passable Tiles*. Walls are completely impassable and serve as barriers. Passable Tiles may contain objects and/or game play characters, i.e. the player-controlled Hero or computer-controlled characters (NPCs). Objects include Treasures, Potions, Traps, Portals, one Entrance, and one Exit. The level ends when the Hero reaches the Exit or when the Hero dies. Game play characters may move between empty tiles by moving either North, East, South, or West. All game play characters have hit points (HP) and can deal damage.

Objects have different properties:

**Treasures** when reached are consumed by the Hero, increasing the player's treasure score. Treasures may also be consumed by Ogres.

**Potions** when reached are consumed by the Hero, increasing the Hero's HP by 1, to a maximum of 10 HP. Potions may also be consumed by Blobs.

**Traps** deal 1 damage to any character that moves into them.

**Portals** come in pairs: a character moving into a portal is immediately (on the same turn) teleported to the linked portal.

**The Entrance and the Exit** determine where the Hero starts and where the Hero must go to complete the level, respectively.

The player controls the *Hero* of the game and always moves first, upon which each object and character in the level in sequence responds deterministically. The Hero starts each level with 5 HP. The Hero possesses a single *Javelin* which can be thrown at any other character to which the Hero has an unbroken line of sight. The Hero deals 1 damage to other characters on collision or by throwing the Javelin at them. The Javelin remains on the tile to which it was thrown, and the Hero must go there to collect it. NPCs then respond to the player's decision:

**Goblins** move 1 step toward the Hero along the shortest path if they have an unbroken line of sight to the Hero. They have 1 HP and deal 1 damage on collision. Goblins avoid colliding with other Goblins or Goblin Wizards.

**Goblin Wizards** deal 1 ranged damage to the Hero, if they have an unbroken line of sight and are within 5 tiles of the Hero; otherwise, they move 1 step toward the Hero. Goblin Wizards have 1 HP and deal no damage on collision.

**Blobs** remain static unless they have an unbroken line of sight to either the Hero or a Potion. If a Blob sees either, it moves 1 tile toward the closest one, preferring Potions over the Hero. When Blobs collide with each other, they merge into a larger, more powerful Blob. The simplest Blob has 1 HP and deals 1 damage to a colliding non-Blob character; this upgrades to 2 HP and 2 damage, and 3 HP and 3 damage at maximum power. A more powerful Blob which receives damage loses one power level.

**Ogres** remain static unless they have an unbroken line of sight to either the Hero or a Treasure. If an Ogre sees either, it moves 1 tile towards the closest one, preferring Treasures over the Hero. If the Ogre reaches a Treasure it consumes the Treasure and becomes fancier to look at. Ogres have 2 HP and deal 2 damage to any other character they collide with, including other Ogres.

**Minitaurs** always move 1 step toward the player along the shortest path as determined by A\* path-finding, disregarding other characters and objects. Collision with a Minitaur deals 1 damage to the colliding character. Unlike other enemy characters a Minitaur does not have HP and does not die, but will be knocked out for 3 rounds if it receives damage.

The characters and objects in *MiniDungeons 2* have simple properties, and respond deterministically to the player's decisions. Still, the number of moving parts in the game means that is difficult to impossible to mentally simulate a full level, at least from the beginning when the hero is far from the exit, and all monsters are still alive. This means that the player must rely on heuristic problem solving to some extent, and extrapolating from theory, we can expect this extent to rely on the player's cognitive resources and motivation. While the complexity of making each decision is higher in *MiniDungeons 2* than in *MiniDungeons 1*, the game contains no stochasticity and in that sense no hidden information or risk, since all outcomes in principle could be calculated ahead of time.

In addition to the balance between analytic and heuristic decision making shaping the player's decisions, we should expect it to be shaped by her decision making style, again representable in a utility matrix.

## 5.4 Chapter Summary

In this chapter, we presented the three games that we use as test-bed domains for studying and modeling decision making.

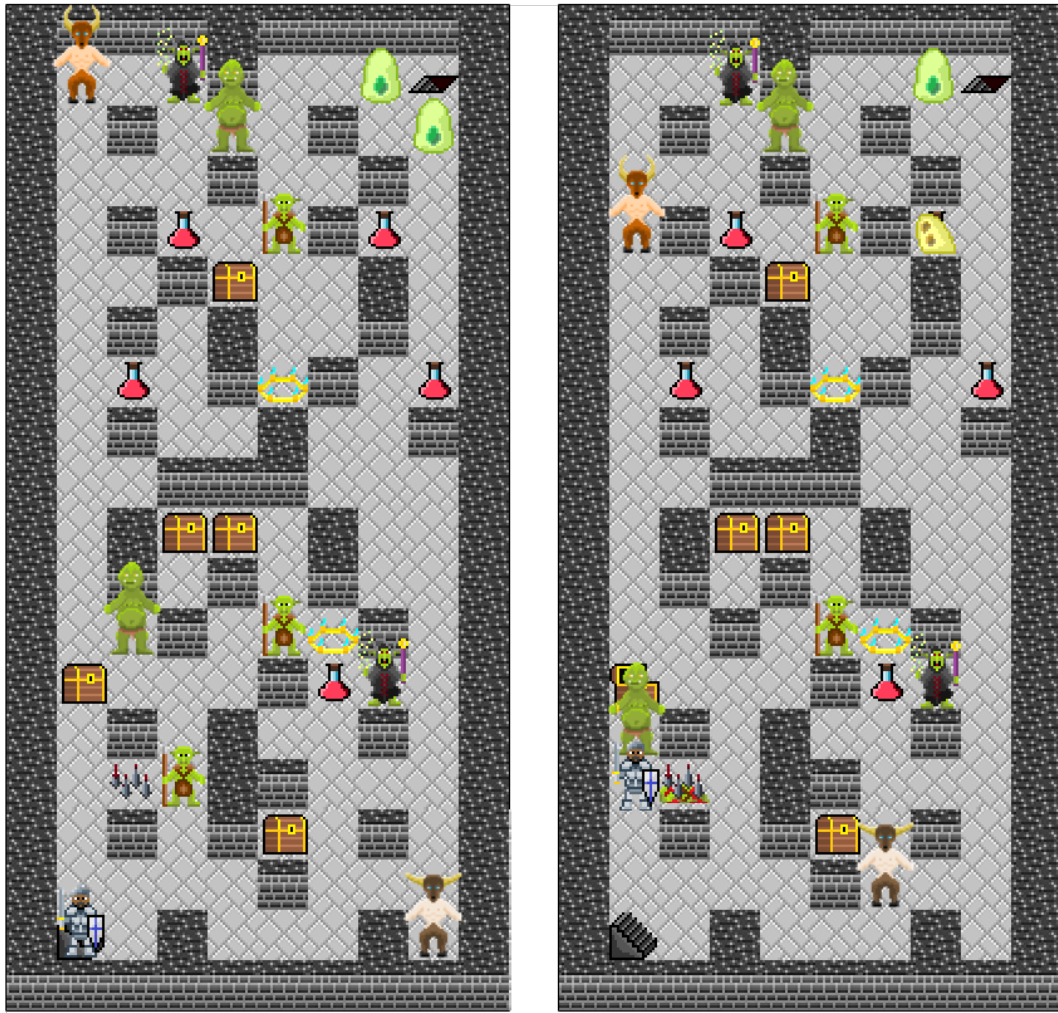


FIGURE 5.3: A typical MiniDungeons 2 level at start (left panel) and after three turns (right panel).

The three games included in this thesis together only cover a small part of the amount of games that it would be necessary to study in order to provide a comprehensive investigation of decision making in games.

In an effort to focus on the problem of decision making making in games, relatively simple 2D games with relatively limited action spaces, no explicit narratives, and no continuity or state permanence between levels were chosen. This naturally limits the scope of the conclusions that can be drawn from this thesis, and future work should focus on expanding the research agenda to games of other genres and greater complexity, a topic we revisit in Chapter 13.

The *Mario AI Benchmark* provides a game which is continuous in time and space from the perspective of the player and is played in real time. This means that the player is

constantly taking random, reactive, and planned actions in the game and it seems plausible that decision making in the game must rely on heuristic and recognition primed decision making to a large extent. The game forces the player to apply bounded rationality since the action progresses in real-time and the enemies in the game act even if the player does not.

In contrast, the *MiniDungeons* games take place in discretized time and space. This may remove the cognitive demand on the player as it sets her free to spend as much time as she wants making decisions. For this reason, we should expect a minimal amount of random and reactive actions in the *MiniDungeons* games, and we should expect players to be able to rely on analytical decision making to the extent that they are able and motivated to. The *MiniDungeons 1* game is not analytically complex in terms of the rules of the game, but the stochasticity of the game forces players to either learn heuristics describing the likely outcomes of monster fights or to analytically consider likely results by recording past outcomes. *MiniDungeons 2*, on the other hand, has no stochasticity, but has a complexity that prevents players from analytically extrapolating the full consequences of their actions, instead relying on heuristics to predict the long term consequences of an individual action.

As such, even though the games at the surface level are similar in their simplicity, they can be seen as quite distinct in terms of the kinds of decision making they afford their players.

The following six chapters present the papers that study decision making in the three domains presented here.

## Chapter 6

# Decision Making Styles as Deviation from Rational Action

### Reference:

Holmgård, Christoffer, Julian Togelius, and Georgios N. Yannakakis (2013). “Decision Making Styles as Deviation from Rational Action. A Super Mario Case Study”. In: *Proceedings of the Ninth Annual AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.

### 6.1 Abstract

In this paper we describe a method of modeling play styles as deviations from approximations of game theoretically rational actions. These deviations are interpreted as containing information about player skill and player decision making style. We hypothesize that this information is useful for differentiating between players and for understanding why human player behavior is attributed intentionality which we argue is a prerequisite for believability. To investigate these hypotheses we describe an experiment comparing 400 games in the *Mario AI Benchmark* testbed, played by humans, with equivalent games played by an approximately game theoretically rationally playing AI agent. The player actions’ deviations from the rational agent’s actions are subjected to feature extraction, and the resulting features are used to cluster play sessions into expressions of different play styles. We discuss how these styles differ, and how believable agent behavior might be approached by using these styles as an outset for a planning agent. Finally, we discuss the implications of making assumptions about rational game play and the problematic aspects of inferring player intentions from behavior.

## 6.2 Introduction

Recent work in Game AI research has seen an increasing interest in the generation of believable bot behavior: bots that not only challenge or interest humans, but play like humans. For instance, the well known Mario AI Championship (Karakovskiy and Togelius, 2012; Shaker et al., 2013b) recently added a Turing Track to the competition and the 2K Botprize (Hingston, 2013; Hingston, 2010) has long been a successful recurring event. Both competitions challenge researchers and developers to create bot players whose behaviors are as indistinguishable from those of human players as possible.

At the core of the pursuit lies evidence that the experience of playing against another sentient player brings engagement and entertainment to most games (Togelius et al., 2012). The phenomenon that believable human-like interaction from a computer system will generally result in human emotional and behavioral reciprocation has been documented for many kinds of human-computer-interaction (Reeves and Nash, 1996). Believability can be construed as a question of being able to attribute intentionality to a bot through inferring from observation that it is exhibiting goal directed behavior and by interpreting and understanding this behavior. Dennett named this process of ascribing agency to an object from assumptions about beliefs and desires the “Intentional Stance” (Dennett, 1987). In short, if we ascribe beliefs and desires to an artificial agent, we are likely to also ascribe to it intentions and from the intentionality comes the tendency toward treating the object as sentient.

Our hypothesis is that the way humans deviate from an optimal course of action in the game theoretical sense, whether due to lack of skill or the pursuit of goals not formalized in the game’s rule structure, is useful in making behavior seem intentional. We further hypothesize that decisions are a useful way of operationalizing the deviations. We attempt such an operationalization by contextually analyzing the observable actions that humans take within the rule systems of games as collected in play traces.

## 6.3 Related Work

Modeling play styles is nothing new in general. Various signals from the player have been employed to enable differentiation between and grouping of players, ranging from facial expressions during game play (Asteriadis et al., 2012) over spatial exploration and behavior overall (Drachen et al., 2009b; Asteriadis et al., 2012) to simply player performance in terms of score (Asteriadis et al., 2012).

Additionally, the psychological literature contains a cornucopia of models for describing how behavior is derived from or influenced by unobservable, latent traits. Below, we briefly visit and relate to previous work in generating human-like behavior from latent trait models and exemplify their application to play style modeling.

### **6.3.1 Latent Trait Models of Behavior**

One approach for creating believable bots is to start out with general models of human cognition and motivation and use these models as generative systems for bot behavior. Cognitive psychology concerned with personality and motivation sees human behavior as the expression of latent tendencies inherent in the individual that are stable over time.

Personality models are generally concerned with preference and appraisal tendencies. They are typically constructed by examining people's preferences and appraisals in response to general questions or specific situations, real or imagined. There is evidence that such models can be used to explain play styles to a certain extent. For instance, personality has been shown to significantly influence behavior in on-line virtual worlds (Yee et al., 2011b). Personality models rest on the idea of a cognitive-emotional system that responds to various situations in a predictable fashion because the system – barring any life-altering experiences such as physical or psychological trauma – is relatively stable over time. Personality models are accepted as having explanatory use and validity in cognitive psychology and have been used for generating believable bot behavior (Rosenthal and Congdon, 2012). A similar, but less investigated and established latent model, is humans having decision making styles as traits. The general idea is analogue to that of personality profiles, but suggests that humans exhibit stable decision making biases. There is evidence that humans possess such stable and predictable decision making styles (Scott and Bruce, 1995). The question then becomes how to recognize such decision making biases in game play.

### **6.3.2 Irrational Play**

The idea that play is not purely rational, but is driven by multiple motivations aside from the wish to perform optimally within the strict rules of the game, is well known. For instance, *agon* is only one of Callois' classic categories (Caillois, 2001) of play and the idea has also been expressed in concepts such as a game-play continuum (Malaby, 2007). In the same vein, research has attempted to capture the deviation from rational play by observing the interactions between players in multiplayer games (Smith, 2006).



### 6.3.2.1 The Compliant-Constructive Continuum.

It has been proposed that the discrepancy between the individual player's intentions, and the ones strictly afforded by the game rules, can be described as residing on a compliant-constructive play style continuum (Caspersen et al., 2006).

The completely compliant player only uses the rules of the game to respond to the affordances presented by the game, attempting to move closer to achieving the winning condition. An example would be a *Counter-Strike* (Hidden Path Entertainment and Valve Corporation, 2012) player who directs all her play efforts toward making her own team win, disregarding any aesthetic preference in e.g. distance to enemies or weapon choice in order to optimize her contribution to her team's coordinated efforts.

The completely constructive player uses the game rules to enable intrinsically motivated actions and situations that may or may not be related to the winning conditions, and hence affordances, of the game. An example would be a player using the game world to enact a protest against violence through inaction or graffiti (Antunes and Leymarie, 2010).

These two examples represent positions of play styles on opposite ends of the compliant-constructive continuum. Importantly, the *completely* compliant and rational play style can be thought of as apersonal and a matter of *solving* the game: From a given state in a deterministic game of perfect information, the compliant-rationally best action(s) will be the same for any player. By extension, the style of a particular player should then be expected to be found in the part of the play actions that are constructive, rather than compliant, and thus suboptimal from a game theoretical perspective.

Taking previous work on play styles, latent traits, and rational play into consideration, the novelty of the approach described here lies in the understanding of play styles as *series of decisions leading to actions systematically deviating from game theoretically rational actions*.

## 6.4 Belief-Desire-Intention Agents

Key to investigating the usefulness of the analytical approach outlined above is having a clear mechanistic model of the process leading from intention to decision to action that can be reverse-inferred with some degree of precision from observed actions.

The artificial intelligence literature provides a well-known framework for describing and generating intentions and plans from beliefs and desires: The Belief-Desire-Intention

(BDI) framework for agents (Rao and Georgeff, 1997). The framework is originally generative, but this work aims to use the model analytically on empirical observations.

The use of BDI agents has a rather long history in simulations, but they have less commonly been found in computer games, the most notable examples perhaps being the *Black & White* series of god-games (Lionhead Studios, 2001, 2005) (Norling and Sonenberg, 2004). The paradigm is fundamentally a reasoning and planning approach that uses three high level concepts, *beliefs*, *desires*, and *intentions*, that together constitute an agent model.

**Beliefs** describe the agent's current representation of its own state and the state of its surrounding environment.

**Desires** describe all the current goals that the agent would like to achieve at a given time.

**Intention(s)** describes the active goal(s) that the agent is pursuing at any given time.

In addition to this, a typical approach is for the agent to have a plan library, from which it selects courses of action to form intentions (Norling and Sonenberg, 2004).

Two main aspects of the BDI paradigm make it especially useful for modeling human decision making: The psychologistic nature of the model maps well to a human introspective understanding of thought processes. Norling and Sonenberg (Norling and Sonenberg, 2004) mention that the framework does not match the actual way that human reasoning occurs, but by small appropriations, the BDI paradigm can fit the typical model of human reasoning from cognitive psychology: If beliefs are understood as a common headline for perception and evaluation of the state of an agent's internals as well as its environment it need not be a completely conscious representation. We can take the beliefs as being the total sum of the agent's available information, including declarative, non-declarative, procedural, and emotional knowledge. By the same reasoning, desires can be understood as motivations widely, incorporating both conscious wishes as well as unconscious drives, instincts and behavioral tendencies all the way down to the conditioned level. Finally, intentions can be understood as the selection basis for the current plan under execution, regardless of whether this plan was motivated by conscious determination or by partially or wholly intuitive or procedural impulses. That means that the moment an intention is formed and followed, in the BDI framework, it can be understood as a decision.

### 6.4.1 Decision Making Styles

In this work we attempt to use the BDI framework as a backdrop for understanding play traces from the Mario AI Benchmark (MAIB) testbed (Karakovskiy and Togelius, 2012), assuming that the actions observed during game play are decisions that in turn are realizations of intentions. We further assume that any variation from the rationally optimal is grounded in intentions beyond the scope of the game rules, and a realization of the player’s play style as a personal trait<sup>1</sup>, realized dynamically in interaction with the context of the game rules. As such, the complete chain of inference leads backwards from actions to decisions to intentions to beliefs and desires and cannot be understood without a deep understanding of the game’s rule structure. We do not necessarily assume that beliefs or desires are wholly conscious, and as such decisions may be based partly or wholly on procedural knowledge and evaluative processes at the subliminal level. This is especially relevant to the MAIB framework, since it is a real-time game where perceptual and motor skills are emphasized.

The key to the project outlined here then becomes the question of how to approximate how a rational agent with the perceptual and motor capabilities of a human player would act while playing MAIB. If we assume that the player is a rational agent, and that we know exactly what information about the game state the player has access to and is able to perceive, the decision space of the game player narrows significantly, and we start becoming able to use combinations of normative and descriptive game theory to approximate what a perfectly playing rational agent would have done. In the following, we treat some of the general challenges to consider in collecting data on decision making from human players and proceed to examine an attempt at handling these challenges in a data set play traces from the MAIB.

## 6.5 Decision Making in the MAIB

The MAIB testbed is a well established framework for working with procedural content generation, player modeling, agent development and testing, and human imitation in platform games. It is a replication of the game mechanics and content of the classic 2D platform game Super Mario Brothers. The testbed has the advantage of being immediately familiar to most players in terms of content and mechanics. This may offset learning effects in experimental conditions and eases the administration of instructions

---

<sup>1</sup>Considering the source of the play style tendency as a trait is beyond the scope of this work, but as noted in the section on related work, decision making style, cognitive and personality models may be of use here.

during experimental runs. The game has simple keyboard inputs consisting of buttons mapped to six different actions: *Down*, *Jump*, *Left*, *Right*, *Speed*, and *Up*.

The MAIB testbed offers several features that are of interest to the study of decision making in games. Because the game is played in a real-time (25 fps) simulation with an experientially continuous deterministic world with predictable physics, it offers a well suited backdrop for studying ecological decision making under well-known conditions. The testbed allows for a practically infinite number of unique positions and action sequences, ensuring that any two sessions played by humans are unlikely to ever be identical. Though the game world offers an impressive expressive richness vis-à-vis its limited reality, the game rules superimposed on top of the game world that the player is afforded to comply with are simple: Players should win levels as quickly as possible, while avoiding or killing enemies in their way, and collecting as many items as possible.

At every given moment during the course of a play through, only a subset of the level is visible to the player, providing no more information than can be assimilated at any given moment by a perceptually and cognitively normally functioning individual. The game presents most *relevant* information about the game state at any given time and can, as such, be considered a game rich in information (though not perfect, as blocks may contain various hidden power-ups and enemies can be hidden from sight for varying amounts of time).

Together these features allow us to construct an agent that plays the game as a completely rational, compliant human player would: By searching through the possible game states from any given position and finding the temporal path that best fulfills the above outlined affordances. One well-performing method for constructing such an agent has been provided by Baumgarten in previous work (Champanard, 2009; Togelius et al., 2010; Baumgarten, 2013) in the form of a MAIB playing A\*-agent. This agent was used to approximate the actions of a perfectly compliant, rational player during data collection<sup>2</sup>.

## 6.6 Method

The following method was developed for discovering player decision making styles from actions performed in the MAIB testbed: Human subjects are asked to play randomly generated levels in the MAIB. All human actions are logged and from these a play trace is constructed, representing their path through each particular level. An A\*-agent solves

---

<sup>2</sup>As the A\*-algorithm itself is well known, we refer to Baumgarten's work (Champanard, 2009; Togelius et al., 2010; Baumgarten, 2013) for methods of appropriating the algorithm to the MAIB testbed. His description and code formed the basis for the implementation used for this study.

each level using a set of affordances determining what possible actions are given priority and to which extent, and a corresponding trace is constructed. In this particular case the only affordances are to complete the level as quickly and safely as possible. The difference between the two is determined by comparing the maximal Y values of the human and the agent traces for each tile in the level resulting in a deviation trace. Additionally, an action deviation matrix is computed between the normalized action frequencies of the player and the normalized action frequencies of the agent. Features are then extracted from the deviation trace, while the normalized input differences for each session are used directly. The set of observations is subjected to cluster analysis to discover play styles across individuals and their play sessions. The prevalence of clusters for each individual is correlated with measures of player skill as an indication of the relation between the human play style and the agent play style. Features are compared across clusters to determine how and to which degree clusters exhibit different play styles. Finally, the original traces of the discovered play style clusters are visualized on selected levels for interpretation.

### 6.6.1 Feature Extraction and Clustering

The following features are extracted from the deviation trace for every play through: The mean (*Mean*) of the deviation trace in order to represent the player's average deviation from the agent trace. The maximum deviation (*Max*) from the agent trace, in order to capture actions extremely different from the agent trace. The standard deviation of the deviation trace (*Sd*) in order to represent the variation of the deviation trace. The means of the first and second differences of the deviation trace (*Diff1*) and (*Diff2*), representing local variation in the deviation trace, i.e. the tendency of the player to move vertically in the level in a manner different from the agent's.

The action frequencies (*Down*, *Jump*, *Left*, *Right*, *Speed*, *Up*) of each player/the agent across the play through of the level are captured as control inputs from each frame of the game. The frequencies are normalized to a range of 0 to 1, relative to the number of frames from start to finish in the play session. The difference matrix is then calculated as the absolute value of the difference for each input type. In order to avoid any direct convolution of skill with play style, no measures of score or performance are used as features.

The total feature set is used as input for an agglomerative hierarchical clustering process, applying Ward's minimum variance method (Kaufman and Rousseeuw, 2005).

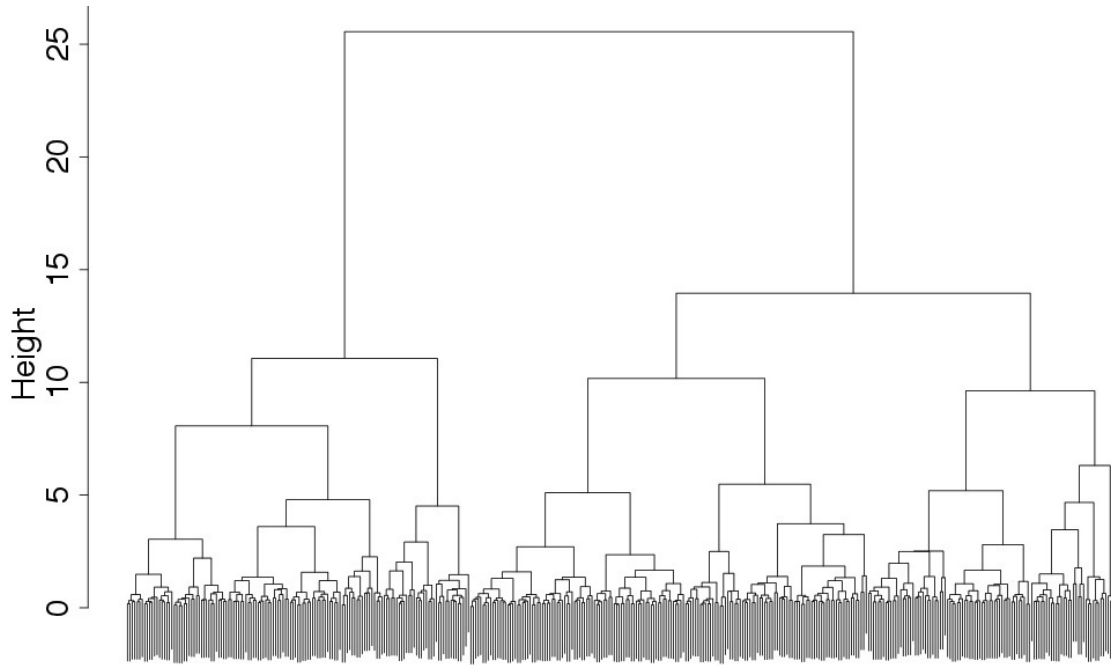


FIGURE 6.1: Dendrogram from the agglomerative hierarchical clustering of the extracted features from 400 observations. The tree was cut at the level of four clusters.

### 6.6.2 Data Collection

A data set was generated from 10 human players playing 40 different, short levels of the MAIB testbed, yielding a total of 400 play sessions. All player inputs were recorded and replayed to generate a trace of the players' routes through the levels. The levels were of varying difficulty, but all were exactly 100 tiles long. On average players completed levels on approximately 35% of the play throughs, though substantial individual differences were observed (range 0-70%, std.dev. 24.7). For each human play through a deviation trace and an action deviation matrix were calculated as described above.

## 6.7 Results

The clustering yielded 4 well defined clusters ( $C_1$ ,  $C_2$ ,  $C_3$ , and  $C_4$ ) depicted in Fig. 6.1. The selected cluster for each session was mapped back onto its player, yielding 10 observations with cluster membership frequencies for each player, depicted in Fig. 6.2. Additionally, each player's average score and average win rate across all sessions were added to the dataset as indications of player performance. A correlation analysis was conducted in order to investigate the relationship between deviation style predominance and in-game performance. The results are reported in Table 6.1 and indicate that the two play styles  $C_1$  and  $C_2$  are correlated positively with performance while play styles  $C_3$  and  $C_4$  are correlated negatively with performance. They also indicate that  $C_2$  is

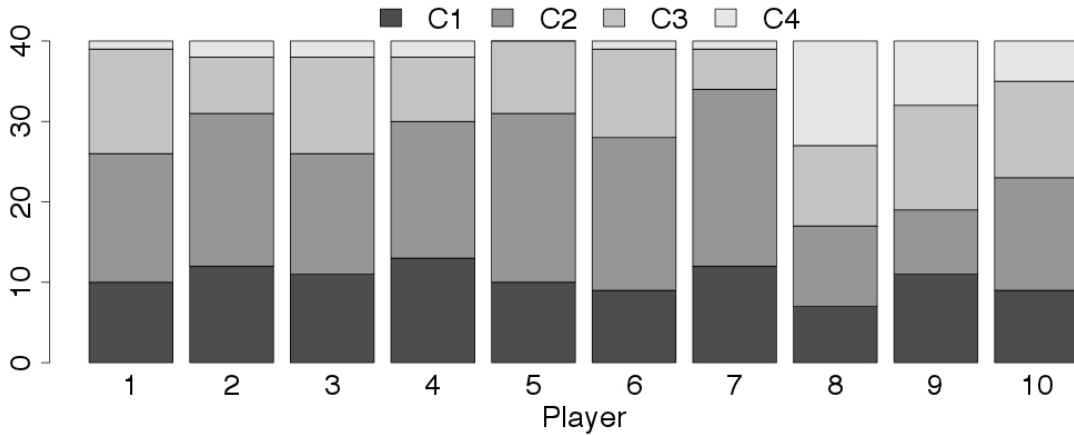


FIGURE 6.2: Players’ individual cluster expressions across all play throughs. Each column represents a player and each shading within each column represents how many of the individual player’s 40 play sessions were grouped into clusters  $C1$ ,  $C2$ ,  $C3$ , and  $C4$  respectively.

Feature	Wins	Score	C1	C2	C3	C4
Mean	-0.06	-0.06	-0.04	-0.05	0.07	0.02
Max	0.15	0.16	0.04	0.17	-0.03	<b>-0.21</b>
Diff1	-0.09	-0.10	-0.06	-0.11	0.06	0.09
Diff2	-0.01	-0.03	-0.01	-0.01	-0.09	0.06
Down	-0.06	-0.06	-0.07	-0.03	0.07	0.00
Jump	<b>-0.33</b>	<b>-0.33</b>	-0.09	<b>-0.35</b>	0.02	<b>0.44</b>
Left	0.08	0.09	0.06	0.14	-0.12	-0.05
Right	-0.13	-0.10	-0.02	-0.03	0.01	0.07
Speed	<b>-0.69</b>	<b>-0.68</b>	<b>-0.46</b>	<b>-0.67</b>	<b>0.72</b>	<b>0.39</b>
Up	-0.06	-0.08	-0.08	-0.05	0.09	-0.00
C1	<b>0.57</b>	<b>0.59</b>	-	<b>0.37</b>	<b>-0.46</b>	-0.16
C2	<b>0.86</b>	<b>0.85</b>	-	-	<b>-0.72</b>	<b>-0.85</b>
C3	<b>-0.66</b>	<b>-0.62</b>	-	-	-	<b>0.27</b>
C4	<b>-0.73</b>	<b>-0.75</b>	-	-	-	-

TABLE 6.1: Spearman correlation between cluster membership frequency, features, and performance measures. Values significant at the  $p < 0.05$  level are in **bold**. Significance values are subjected to Holm-Bonferroni correction for multiple tests.

characterized by a *Jump* frequency close to that of the agent, while  $C4$  is characterized by a *Jump* frequency different from the agent’s, and that  $C1$  and  $C2$  have *Speed* frequencies closer to the agent’s in contrast to  $C3$  and  $C4$ .

A mapping of play style clusters to levels was conducted, showing how often a given play style was expressed on each particular level. The frequencies of the results were used to identify levels that allow for the expression of all clusters identified across the dataset. The results are presented in Fig. 6.3 and indicate that most levels only enable the expression of some play styles.

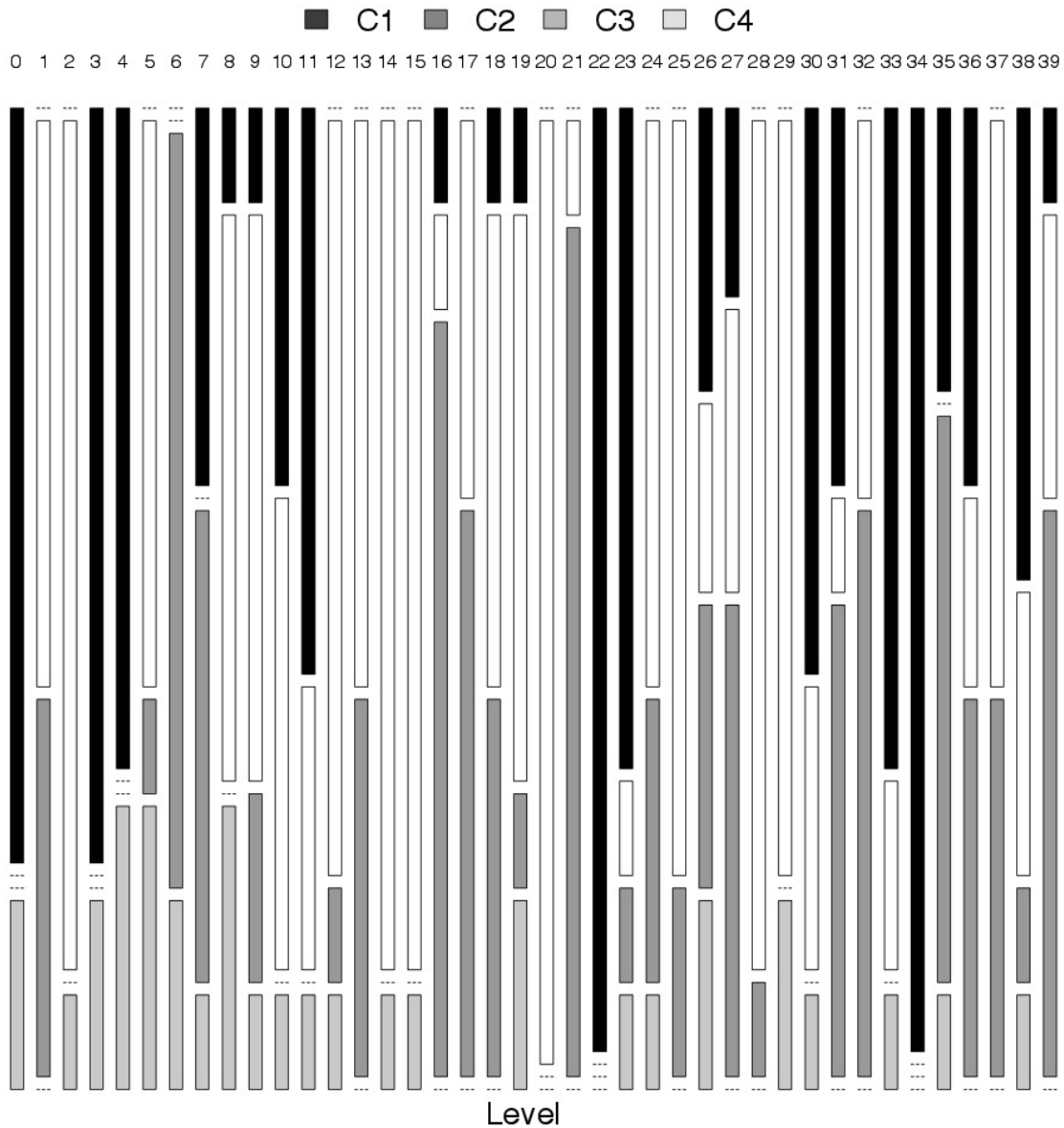


FIGURE 6.3: Expressed play style clusters mapped to levels. Each shading represents a cluster.

To further investigate the differences between the clusters across the features, the centrality of each cluster with respect to each feature was established using the Hodges-Lehmann Estimator of Location. To test for significant group differences, each feature was subjected to a Kruskal-Wallis analysis of variance (Lehmann and D'Abbrera, 1975). The results are reported in Table 6.2 and indicate that most features significantly differ across groups, with the exception of *Diff2*, *Down* and *Left*. Overall, *C4* resembles the agent the most as expressed by the mean trace difference. Examining the correlation between *C4* and performance measures, it seems plausible that *C4* resembles the agent because this play style dies early, but acts like the agent in the early stages of the level before dying. This interpretation is supported by the fact that the *Jump* and



<b>Feature</b>	<b>C1</b>	<b>C2</b>	<b>C3</b>	<b>C4</b>	<b>H</b>	<b>df</b>	<b>Sig.</b>
Mean	1.96	1.73	2.88	1.54	174.0	3	0.00
Max	6.00	7.50	7.50	4.50	309.3	3	0.00
Sd	1.70	1.91	2.22	1.43	152.4	3	0.00
Diff1	0.10	0.10	0.17	0.09	26.5	3	0.00
Diff2	-0.00	-0.04	-0.02	0.00	3.1	3	0.38
Down	0.00	0.00	0.00	0.00	5.8	3	0.12
Jump	0.08	0.08	0.09	0.23	15.8	3	0.00
Left	0.04	0.03	0.04	0.03	5.9	3	0.12
Right	0.44	0.34	0.38	0.42	25.7	3	0.00
Speed	0.60	0.53	0.78	0.91	16.2	3	0.00
Up	0.00	0.00	0.00	0.00	9.8	3	0.02

TABLE 6.2: Kruskal-Wallis H tests for differences between clusters. *C1-C4* contains the Hodges-Lehmann Estimator of Location (Lehmann and D’Abrera, 1975), as a measure of centrality, for each cluster for each feature to allow for comparison. Note that for some features, e.g. *Speed*, all clusters are different from one another, while for others, e.g. *Jump*, one cluster deviates.

*Speed* frequencies of *C4* are very different from the agent’s, while the *Jump* and *Speed* frequencies of the high-performing *C2* cluster are closer to those of the agent.

For exemplification, three levels enabling all play styles were selected and visualized in Fig. 6.4. These graphs indicate that *C2* traces resemble agent traces. So do *C4* traces, but only for a short period of time before losing, cutting the session short. The graphs show *C1* and *C3* diverging from the agent in terms of trace path, with varying performance. In terms of the compliant-constructive continuum relative to the A\*-agent, we suggest that *C2* and *C4* could be considered compliant and skilled/unskilled respectively, while *C1* and *C3* are acting more constructively and less stably. This interpretation, of course, only holds to the extent that one accepts the A\*-agent as a relevant proxy for a rational, compliant, skilled MAIB player.

## 6.8 Discussion

The results presented in this paper reveal a number of insights about the play styles of the participants. The clustering of the play styles and the correlations to performance measures indicate that the applied approach might indeed be able to differentiate between different kind of play styles, and by extension that the operationalization of decision making styles into deviation from rational behavior is applicable for a game of the scope of the MAIB. Further work should be undertaken, however, to investigate and control for the influence of the particular level as Fig. 6.3 suggests that such an influence is present.

Also, multiple theoretical assumptions still stand unresolved. It is unclear from the results presented here to which extent the assumptions of what constitutes rational behavior are indeed appropriate and how this approach would transfer to games of higher complexity than the MAIB. The A\*-agent is arguably a high-performing solution for the MAIB, but different agents might perform equally well. With more than one normative game theoretical solving approach to the game, it becomes difficult to prescribe one over the other as a perfect baseline. From a pragmatic perspective any high-performing solution might be good enough, as long as it serves as a baseline by which to differentiate players, but it is an assumption that warrants further research. One approach to this problem could be the construction of multiple baseline agents and characterizing players in terms of which baseline agent they resemble the most or the least — generating procedural personas.

A first step for future work should be to include longer and more varied levels, allowing for a greater expressive range in the play throughs. This would also force an elaboration of the assumptions of what rational behavior in the MAIB is, and would necessitate a hierarchy of affordances or a similar method for identifying the most plausible intentions of the player from multiple options. Should the agent prefer enemy kills over bonus items or vice versa? In the scope of the current study, this consideration is not necessary, as the only affordances are reaching the end of the level as quickly and safely as possible. While this limited scope benefits this first tentative exploration of the approach, more complex game situations will be needed to push the boundaries of decision style identification from deviations from rational play. In the same vein it is clear that any decision styles extracted from player behavior will be dependent on the game in question and that the extent to which decision making styles generalize between games remains unknown. If decision making styles in games are stable traits expressed across situations, this should be detectable across different games, but it remains to be seen if the contexts drown out any signal of the individual player's decision making style. A comparative study of multiple games, preferably using the same sample of players, would be necessary.

Also remaining is the answer to the question of how to use the clusters identified as decision making styles for synthesizing behavior in the BDI framework. The extension of this method to a multiple-agent approach would enable this, if each agent was characterized (even if not strictly implemented) in the BDI framework, exhibiting different desire hierarchies and different plans for achieving these desires. This could be achieved by using a battery of different agents or by constructing an agent with dynamic affordance-response preferences that could be weighted between or during play-throughs.

Finally, the difference trace used in this limited study is created with reference to an agent completing the whole level. A more precise approach might be for the agent to

determine its preferred action for each frame of the player’s play session, yielding a difference trace based on moment-by-moment differences instead of session-based differences.

## 6.9 Conclusion

We have presented a framework for characterizing player behavior in terms of deviations from rational actions. We believe this framework could be used as a foundation to further understanding of player behavior, which is often analyzed in a rather ad-hoc way using unsupervised learning. This framework was demonstrated using an analysis of play styles in the Mario AI Benchmark, with a high-performing A\*-based agent providing the ground against which human play traces were contrasted. This analysis yielded features that allowed players to cluster meaningfully with significant differences between them. These clusters were also found to correlate with playing performance. The current work provides ample opportunity for further investigation.

## 6.10 Acknowledgments

The authors would like to thank Juan Ortega and Noor Shaker, as well as all players, for their contributions in creating the dataset. We also thank Robin Baumgarten for making his code publicly available under the WTFPL license. Finally, we would like to thank our reviewers for feedback and suggestions for future work.

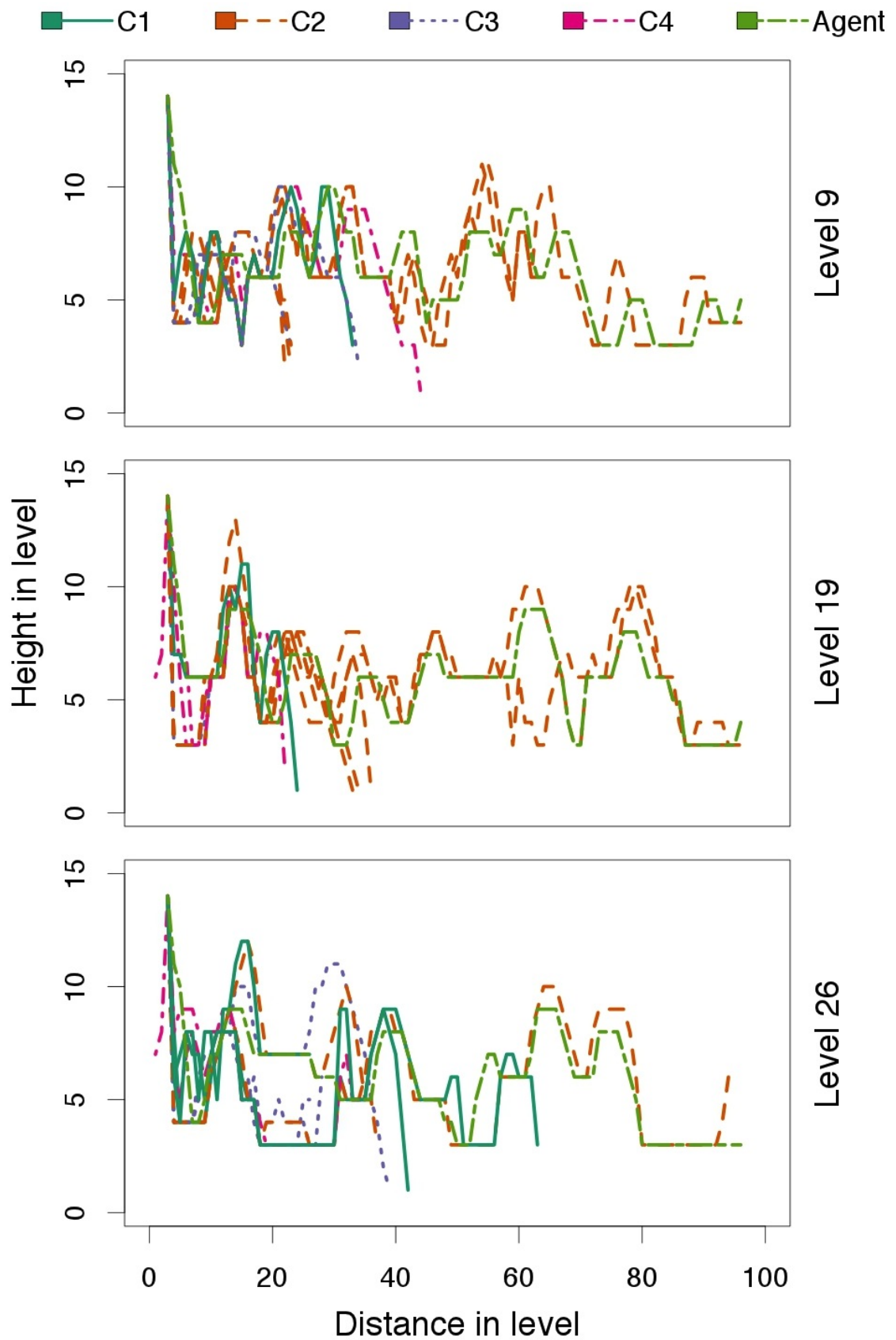


FIGURE 6.4: Traces from three levels. Each graph represents a trace of an individual player or the agent. Note that the high-performing *C2* traces bear resemblance to the agent traces.

## Chapter 7

# Generative Agents for Player Decision Modeling

### Reference:

Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis (2014). “Generative Agents for Player Decision Modeling in Games”. In: *Foundations of Digital Games*.

### 7.1 Abstract

This paper presents a method for modeling player decision making through the use of agents as AI-driven personas. The paper argues that artificial agents, as generative player models, have properties that allow them to be used as psychometrically valid, abstract simulations of a human player’s internal decision making processes. Such agents can then be used to interpret human decision making, as personas and playtesting tools in the game design process, as baselines for adapting agents to mimic classes of human players, or as believable, human-like opponents. This argument is explored in a crowdsourced decision making experiment, in which the decisions of human players are recorded in a small-scale dungeon themed puzzle game. Human decisions are compared to the decisions of a number of a priori defined “archetypical” agent-personas, and the humans are characterized by their likeness to or divergence from these. Essentially, at each step the action of the human is compared to what actions a number of reinforcement-learned agents would have taken in the same situation, where each agent is trained using a different reward scheme. Finally, extensions are outlined for adapting the agents to represent sub-classes found in the human decision making traces.

## 7.2 Introduction

This paper describes an approach to modeling, grouping and interpreting players based on their inferred utility function modeled through reinforcement learning in the form of a generative agent. It proposes how this could be extended to adapt to derived player groups through a process of clustering and inverse reinforcement learning. The example presented here uses trained Q-learning agents with manually configured reward parameters as a priori defined personas, but in principle any agent with a trainable, configurable reward function could be used to generate easily interpretable generative player models for player characterization and design support.

Player modeling in its various expressions facilitates at least one of four purposes: the description, prediction, interpretation, and in some cases reproduction of player behavior (Smith et al., 2011; Yannakakis et al., 2013). All four purposes are rarely addressed simultaneously in the same model for theoretical and practical reasons. An interest in understanding groupings of players might not necessarily entail an interest in accurately *predicting* or *reproducing* their behavior. Inversely, in order to create a player model that reproduces player behavior, it might not be necessary to account for *why* players exhibit a certain behavior in-game — only that a reproduction is convincing to a human spectator. Still, certain areas of investigation or specific applications might mandate the pursuit of player models that address all four purposes at once.

This paper expands on previous work (Holmgård et al., 2013a) and attempts to span all four purposes outlined above: it aims to describe, predict, and reproduce player decision making with the overarching goal of facilitating its interpretation. By modeling players as decision making agents and using these models to characterize players by their induced motivations, a high-level sketch of the players' decision making processes is drawn facilitating the interpretation of their preferences. The approach draws on the theoretical framework of prospect and decision theory, considering every action in a game a decision made under uncertainty (Tversky and Kahneman, 1974; Rubinstein, 1998).

This approach is based on three fundamental assumptions: The first assumption is that players exhibit a particular decision making tendency or style when playing a particular level or game, and that this tendency can be captured and expressed by approximating a *utility function* that shapes their decisions in-game. This utility function is *latent* in the sense that it cannot be observed directly. The second assumption is that in order to *validly* model this assumed utility function, the elements and procedures of the utility function, as a psychological construct, should be explicated in a manner that accounts for the key components and processes of the player's psychology, the outcomes of which

can be empirically observed. The third assumption is that a priori outlined models of player decision making styles can be used as archetypes or personas by comparing their generative output with the empirical observations of human decision and that a well-motivated artificial process that generatively mimics a player constitutes a valid abstract model of the player's internal process.

In the following section, we will first discuss related work from psychology and artificial intelligence, and the epistemological assumptions in that work. In Section 7.4 we explain the general structure of our experiments as well as the artificial intelligence methods involved. Sections 7.5 and 7.6 describe our method for data collection from human players, and the results of our attempts to classify human playtraces according to agreement with generative agents. We conclude by discussing the potential and limitations of the current work and overall methodology.

## 7.3 Related Work

The presented method of player decision style modeling draws in parallel on the literatures on decision theory, psychometric validity, and player modeling. This section outlines the insights drawn from each field and motivates the synthesis of the three.

### 7.3.1 Decision Theory

Psychology, behavioral economics and game theory share a common history under the umbrella of decision theory which tries to describe decision making through formal models. One of the central ideas of decision theory is that any decision a human makes under uncertainty, due to incomplete information or a stochastic outcome, is guided by a *utility function* that determines the decision makers willingness to take risks for an expected reward.

*Utility* to the decision maker is considered idiosyncratic, and decision theory makes no general claims about this, but typically defines a particular conception of utility a priori. Whatever is desirable to the decision maker is a potential source of utility to various degrees. It prescribes that, given its conception of utility, an agent acts rationally when it optimizes, within its computational constraints, its actions to achieve it (Rubinstein, 1998). The utility function describes the decision maker's risk/reward policy for this optimization.

Here, the purpose is to develop a method for decision modeling that is relevant for a wide range of computer games, including ones that support (even if they might not suggest)

unstructured play in the game world — or have competing or even conflicting goals. For that reason it is most relevant to interpret any player input to the game as a decision that is expressive of a utility function that is shaped both by the interaction between the game’s overt rules, its expressive space in total, and the player’s motivations and capabilities. Any action the game *affords* the player (Gibson, 1977) becomes a potential source of utility. If any action in the game can be taken as an output of the player’s utility function, this in turn allows for inducing the player’s concept(s) of utility by approximating and then interpreting the utility function.

Since the utility function weighs the values of the constituents of future game states, relative to the risk involved with potentially attaining them, it is necessary to define these constituents before attempting to model the utility function — constructing a selection of affordances that *could* provide utility in the game. This comes with the risk of identifying only a subset of the actual affordances or perhaps picking the wrong ones altogether.

Once an acceptably large set of possible affordances are defined, an approximation of the player’s utility function could technically be accomplished by any generative computational method capable of simulating the human decision maker, rule-based or search-based. However, since the interest here is not only reproducing the utility function, but also interpreting the computational generation of a given utility value as an abstract representation of the player’s same process, it is necessary to apply methods that allow for the inspection and interpretation of the weightings of the affordances behind the utility function. Once successfully constructed, such a model is then interpreted as an abstract simulation of the player’s decision making process. The following section briefly argues why this methodological approach can be considered appropriate in terms of psychometric validity.

### 7.3.2 Validity in Latent Trait Modeling

To construct player models that aim to discriminate between players or predict their actions, by modeling a process that is completely internal to the psychology of the player and therefore unobservable, an argumentation for the validity of the proposed model of the player’s psychology is necessary. The work presented here attempts to induce the player’s sources of utility, treating the utility function as a latent trait or state within the player, motivating her behavior. Recent research in psychometrics argues that a particular test or model for measuring a latent attribute is valid if “a) the attribute exists and b) variations in the attribute causally produce variations in the measurement outcome.” (Borsboom et al., 2004). Although at first glance this seems intuitive, the



necessity of a causal relation between the attribute and the measurement outcomes puts an explanatory onus on the theoretical framework and assumptions of the model. A psychological concept that cannot produce theoretical reasons for assuming the modeled processes in the psychology of the player runs the risk of regressing to operationalism where the process in the player is defined as what is measured through the empirical methods (Pedersen and Yannakakis, 2012), potentially mistaking outcome correspondence for process correspondence. To avoid this risk, a model that claims to represent unobservable processes in the psychology of a player needs a clear mechanistic chain of inference from the context to the player action to facilitate description, prediction, interpretation, and reproduction. Otherwise, it cannot claim to model the internal process of the player, but only produces a potentially unrelated, even if effective, mapping between the input and output states (Borsboom, 2005).

This is specifically what this work attempts to address by developing a model of player decision making that takes into account high-level characteristics of the human decision process, while remaining reasonably intelligible, by making strong claims about what aspects of a decision problem are evaluated by the player and what importance the player attributes to each aspect in the form of a persona.

### 7.3.3 Player Modeling

Yannakakis et al. (2013) present a high-level overview of player modeling approaches and argue that player models always, at least in an abstract sense, incorporate the whole player either overtly or tacitly. The paper usefully separates model-based and model-free player modeling approaches, while pointing to the fertile, hybrid middle ground between the two. From this perspective, the approach taken here is model-based in the sense that it makes strong assumptions about the psychology of the player and represents it in the form of agent-personas, but the actual agent training is model-free in the sense that a Q-learning agent is used. As such, the method presented here is a hybrid one.

Smith et al. (2011), present a useful, inclusive taxonomy of player models, identifying opportunities for filling gaps in the already known gallery of approaches to player modeling. They present four facets of player models that can be used to describe their *kind*: the scope, purpose, domain and source of the player model. The method that is presented here would, under their taxonomy, be categorized as a Class Induced Generative Action model. Smith et al. specifically note that “Class models are more difficult to motivate in an academic context, requiring either justification of a theory of stereotypes or aggregation of sufficient individual data to build up class descriptors. Thus, we expect class models to be used more in practice than they are reported.”. This precisely touches

upon the considerations of validity outlined in the preceding section, and helps explain why the class based category of academic player models has no examples in Smith et al.’s survey. The quote also describes the potential applicability of class based models: Stereotypical players, or personas, are widely used in game design and development for guiding content creation (Tychsen and Canossa, 2008; Canossa and Drachen, 2009), taking the place of play testers when actual play testing is infeasible or undesirable. Typically, a game designer uses the persona as a starting point for imagining what the persona would do in a particular part of the game, or actually plays the game while informally simulating the persona’s play style. This implies that the game designer has a mental model of the decision making process of the persona, typically based on previous experience and the interpretation of qualitative and/or quantitative data from play testing, metrics, etc. The purpose of our modeling method is obviously not to supplant this part of the game design process, but to provide the game designer with an external representation of not only *how* different personas would play the game, but at the abstract level also *why*. Such a model could form a point of comparison and contrast to the game designer’s internal mental model or become part of a mixed-initiative content authoring tool, suggesting content suitable for one or more personas, configured by the designer, adapted to human data, or built as a hybrid of the two.

## 7.4 Testbed

For the purpose of exploring the argument presented above, a simple testbed game was created along with a set of archetypical generative agents.

### 7.4.1 Game Environment

The game environment, MiniDungeons (see Fig. 7.1), aims to evoke the fundamental mechanics of a rogue-like dungeon exploration game. It puts the player in a two-dimensional dungeon on a grid of 12 by 12 tiles, viewed from a top-down perspective. Tiles are either passable or impassable to the player. Passable tiles may be occupied by monsters, rewards, potions, the dungeon entrance or the dungeon exit. All tiles and their current state is visible to the player, so the game applies no notion of fog-of-war or limited visibility. The player has a hitpoint counter and a treasure counter, and the player loses the level if her hitpoints (HP) drop to zero. The player starts each level at the dungeon entrance with 40 HP, and every turn can move to any adjacent, passable tile. When moving onto a monster tile, combat is resolved instantly, the monster is removed and the player loses a number of HP. Combat is stochastic: enemies may deal between 5 and 14 points of damage, determined each time the level starts. Moving onto a treasure tile



FIGURE 7.1: The game environment on one of the levels used in the experimental protocol. The hero, shown in gray armor, moves around the level collecting treasures (brown closed chests), potions (red bottles) and killing enemies (green goblins). The hero starts at the entrance (stairway leading up, left of the screen) and the level ends when the exit is reached (stairway going down, right of the screen). The hero's hitpoints are shown at the bottom, along with the number of treasures collected and the most recent event.

removes the treasure and increases the treasure counter by one, while moving onto a potion tile removes the potion and increases the player's HP by 10 (up to a maximum of 40). If the player moves onto the dungeon exit, the level is completed.

The number of tile types and allowed player actions is very limited, and monsters do not move. Hidden information is only a factor in the game for combat actions, as

enemies deal a variable amount of damage, but the damage range is quickly induced after a few rounds of combat. For all purposes, the complete game rules are quickly learned by human players and are simple enough to potentially allow a number of agent construction approaches.

The relatively small size of the level and the fact that it is a discretized space, results in a high decision density. Even a single action, such as moving to an adjoining empty tile, significantly changes the game state in terms of remaining steps to the exit, monsters, potions, and treasures. This means that any input that significantly changes the game state entails a specific decision. The bounded number of affordances in this testbed limit the number of utility sources that must be considered when constructing an agent-persona. Finally, the small level size means that most playthroughs can be completed relatively quickly.

### 7.4.2 Generative Agents

To produce an agent representative of archetypical players, any technique capable of incorporating the concept of a utility function would technically be a possibility. Any reinforcement learning technique satisfies these requirements, including any form of dynamic programming, Monte Carlo methods, and temporal-difference learning (Sutton and Barto, 1998). Among them, one-step Q-learning was selected for its simplicity as well as its ability to handle the stochastic nature of combat implemented in the testbed game. Additionally, the small gameworld and limited number of hero moves in each level position permit the use of a lookup table for storing state-action pairs. In an attempt to maintain the Markov property of each state, states in the lookup table consist of the entire gameworld (including passable and impassable tiles, the hero's location and the location of undefeated monsters and uncollected treasures and potions) as well as an abstraction of the hero's hitpoints. The latter is encoded as an integer with 4 possible values, with 0 for 1-5 HP (can certainly not defeat any monster), 1 for 6-14 HP (is likely to die from a monster), 2 for 15-30 HP (can defeat at least one monster) and 3 for 31-40 HP (will not benefit to the full extent from a potion). The addition of these hitpoint ranges to the state description implicitly includes a model of the environment since the enumerators were selected based on the damage range of monsters and the HP healed by potions; although one of the advantages of temporal-difference learning is its ability to operate without a model of the environment, the addition of hitpoint enumerators aimed to speed up convergence of the Q-learning process.

In Q-learning (Watkins and Dayan, 1992), the agent in a particular state  $s$  performs an action  $a$  (move up, down, left or right) and observes the subsequent state  $s'$ . The  $Q(s, a)$

TABLE 7.1: Description of agents.

Agent (Abbrev.)	Playing Style
Baseline Player (B)	Reach exit.
Runner (R)	Minimize moves.
Survivalist (S)	Minimize risk.
Monster Killer (M)	Kill all monsters in level.
Treasure Collector (T)	Collect all treasure in level.

TABLE 7.2: Rewards  $r$  for specific game events.

Event	Agent				
	B	R	S	M	T
Killed monster				1	
Was killed			-1		
Reached exit	0.5	0.5	0.5	0.5	0.5
Collected treasure					1
Moved		-0.01			

value is then increased by  $\alpha[r + \gamma \max_a Q(s', a) - Q(s, a)]$ , where  $r$  is the reward in state  $s'$ ,  $\alpha$  is the learning rate and  $\gamma$  is the discount factor of future rewards. For training the agents in the presented experiment on a specific game level,  $2.5 \cdot 10^5$  games were played with  $\alpha = 0.5$  and  $\gamma = 0.9$ . During training, the action with the highest Q value was selected with a likelihood of  $1 - \epsilon$  ( $\epsilon$ -greedy); in the experiments detailed in this paper,  $\epsilon$  starts at 1 and starts decreasing linearly after 2500 games from  $\epsilon = 1$  to  $\epsilon = 0.1$  at the end of the training session. When not selecting the highest Q value or in unvisited states, exploration favors the least often taken action in that state.

The reward function of the Q-learning agent is simply the model of the player’s utility function. In order to produce multiple different personas for comparisons with players, a number of distinct agents were developed which had different playing styles (see Table 7.1). All possible outcomes of an action are assigned rewards and each agent (except Baseline) receives a single additional reward; this is expected to create distinct behaviors each emphasizing a particular affordance as a source of utility. While more elaborate strategies with multiple rewards could be included, this paper focuses on “archetypical” agents which are straightforward to understand or modify by designers.

## 7.5 Data Collection

In order to collect human decisions in the form of playtraces in the game environment, a crowdsourcing experiment was conducted. The experiment placed the game on a public webpage which was advertised via e-mail and social media.

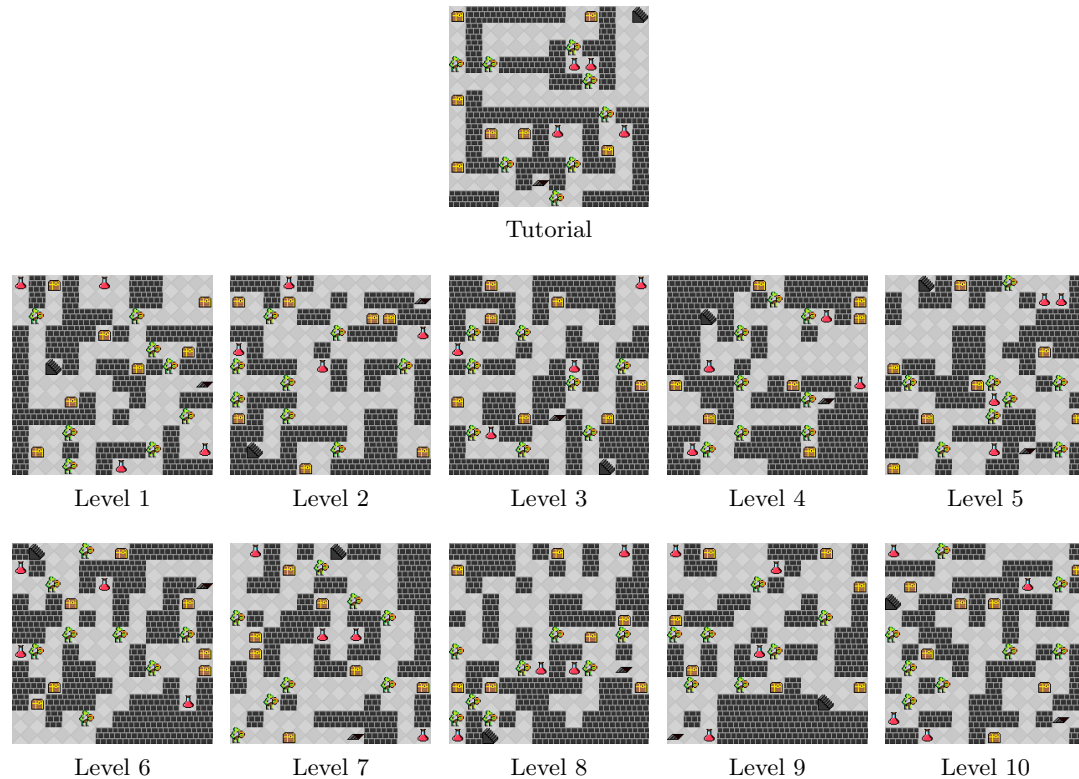


FIGURE 7.2: The levels used for the data collections experiment. The tutorial level is hand-crafted, and could be played multiple times. The “real” levels (1-10) were played only once (no retries if the hero died) and were created in a mixed-initiative fashion.

The starting screen informed the participants that they would be taking part in an experiment concerning computer games, but not its goals of modeling decision making styles. Upon starting, participants had the option of voluntarily providing their name and e-mail address and were informed that participants who chose to do so would enter a lottery and a competition. One participant would be drawn at random and additionally the participant who “did best” would receive a prize as well. In order to ensure variation in the players’ concepts of utility, the notion of what constituted *best* was not explained and left to the player’s imagination. This design choice was expected to motivate players to exhibit different play styles, i.e. allocating different priorities to reaching the exit of the level, avoiding damage, killing monsters, or collecting treasures and potions. By the same logic, the decision to participate in the competition and lottery was left to the player, since we assumed that this would be of utility to some players and irrelevant to others.

Following a brief introduction on the mechanics and visuals of the game, participants began play on a “tutorial level”, which they were allowed to replay as many times as they wished, followed by 10 “real levels” (see Fig. 7.2), each of which they could play once (i.e. without replays if the hero died). Between levels, players were presented with a summary screen of their previous level, with information on the hero’s final HP,

monsters killed, treasures collected, potions drunk, actions taken and percentage of level explored. As with the choice of leaving the notion of best performance unclear in the starting screen, showing as many diverse statistics as possible was expected to elicit different play styles among participants. All player actions on every level were logged and stored in an online database.

Apart from the hand-crafted tutorial level, the levels used in the protocol were created via a mixed-initiative design process. Dungeons were generated via constrained genetic algorithms according to the process described in Liapis et al. (2013d), followed by manual adjustments in order to increase the range of interesting, risky actions and the rewards they offer. Most levels have multiple paths to the exit, each path needing different degrees of combat or no combat at all. All levels also have side passages and diversions, with treasures and potions often guarded by monsters, but at times also unguarded, either at the end of a long side passage or along a path to the exit. Finally, monsters are usually placed in corridors allowing no way through except via combat; some levels (such as level 8) also include unavoidable monsters on the path from entrance to exit.

### 7.5.1 Human Playtraces

38 players successfully completed all 10 levels of the experiment. Some of the most consistent behavioral patterns across players was that of treasure and potion collecting, since both were collected quite consistently by most users.

While treasures were never explicitly deemed important and serve no in-game purpose, the name itself and its significance in many role-playing games plausibly made several players strive towards collecting all of them; the fact that, apart from hit points, treasures collected was the only other statistic visible on the user interface may have also contributed to this. Although not all players targeted treasures, 32 of the 38 players finished the levels with more than 60 total treasures (out of 70). Potions, on the other hand, were often collected by necessity in order to survive combat with monsters which were, for the most part, guarding treasures. As such, it is not surprising that most players collected potions, although there was not as obvious or consistent a drive to collect potions as there was for collecting treasure; out of 38 players, 22 finished the levels with more than 30 total potions (out of 40), and 11 with more than 35.

In terms of actions taken and tiles explored, little variation between players existed, although the (few) outliers are of interest. Two players finished all 10 levels having visited 349 and 395 tiles in total, respectively, which compared to the average 594.4 explored tiles across players indicates that they were trying to complete each level quickly, possibly due to lack of interest or in order to see the next level.

In terms of monsters killed, player behavior was less consistent: since every level contained 8 monsters, even with the help of potions the likelihood of defeating all of them was slim due to the stochastic nature of combat. Due to the fact that each level had different needs for killing monsters (such as unavoidable monsters for reaching the exit), there were few consistent patterns either between players or between levels. The data indicates, however, that players did not explicitly target killing monsters as their goal, possibly because they had no chance of replaying the level if they died. Of the 38 players, only 13 finished the levels with more than 60 total monster kills (out of 80) and only 5 with more than 70. Even players who collected all treasures in all levels did not succeed in killing all the monsters in every level, and no player reached 80 out of 80 monster kills.

An interesting visual aid for qualitatively assessing the behavior of different players is the level’s “heatmap”, i.e. the tiles visited by the player during her playthrough. Fig. 7.3 shows some indicative heatmaps of different players on the same level, which illustrate the different player behaviors. Certain players acted as “completionists”, and explored most of the level, collected all the treasure, drank all the potions and killed all the monsters (Fig. 7.3a). Other players rushed to the exit, killing only the minimal number of monsters and ignoring treasures and potions even if they were not guarded by monsters (Fig. 7.3e). Many players collected the unguarded potions and treasures, and a few guarded ones if the risk was limited (Fig. 7.3c) while others did not accurately assess the risk involved and died; Fig. 7.3b and Fig. 7.3f are particularly good examples of the latter, since the players could have collected the unguarded potions before attacking the monster which killed them.

### 7.5.2 Artificial Playtraces

The five generative agents of Table 7.1 were trained for each level of the user study. Each agent was trained via  $2.5 \cdot 10^5$  playthroughs, using the parameters described in Section 7.4.2. Once training was completed, exploration and learning were disabled ( $\alpha = \epsilon = 0$ ) and 20 test playthroughs of the level were performed to assess the agent’s performance — unless otherwise noted, statistics in this section will refer to the average of those 20 playthroughs.

The behavior of the generative agents was largely dependent on the level in which they were trained. Table 7.3 includes some indicative game statistics of the agents’ overall playthrough of levels 1 to 10, which provide some insight on the agents’ behavior. In several levels the Baseline (B), the Runner (R) and the Survivalist (S) agents had very similar behaviors as they took the shortest path to the exit (see Fig. 7.4a, where their



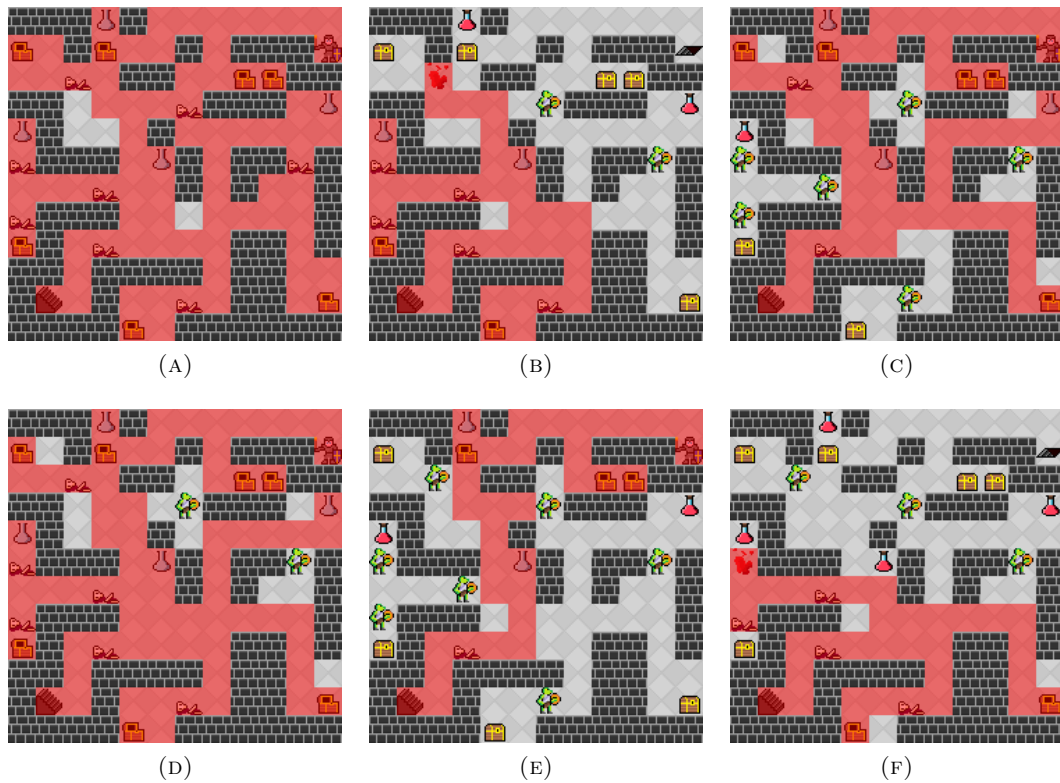


FIGURE 7.3: Heatmaps of selected players in Level 2. Some acted as “completionists”; others rushed to the exit. Many players only collected guarded items if the risk was limited while others took excessive risks and died. The heatmaps indicate that a single level allows for different decision making styles in spite of the apparent simplicity of the testbed.

heatmap is identical); this was due to the fact that the shortest path to the exit usually did not contain enough monsters to kill the player (which would be detrimental to the Survivalist agent). Despite such similarities, agent S did not die in any of the 200 runs (20 runs of each of the 10 levels), while agent B died 13 times, agent R died 23 times, agent M died 63 times and agent T died 169 times. The high death rate of Monster Killer (M) and Treasure Collector (T) agents is due to the fact that, since they were not penalized for dying, the agents took unnecessary risks to kill monsters and collect treasures, respectively. While they were not as thorough in clearing the entire level as human players, agent M finished all 10 levels with 53.8 total monsters killed (out of 80) while agent T finished all 10 levels with 48.9 treasures (out of 70), far more than other agents. Of the remaining statistics it is worth noting that the Runner agent finished all levels with the lowest number of tiles explored, although agents B and S have only somewhat higher values. Finally, the Monster Killer agent collected the largest number of potions in order to survive more combat encounters and achieve more monster kills. The Survivalist agent was also expected to collect a fair number of potions, in order to increase the chance of surviving, but the fact that most levels did not have enough

TABLE 7.3: Game statistics of each artificial agent for the entire playthrough of 10 levels. With the exception of Times Died, values are averaged across 20 test runs; Times Died includes all 200 playthroughs tested.

Statistic	Agent				
	B	R	S	M	T
Monsters	22.7	22.6	21.4	53.8	48.2
Treasures	9.4	7.8	11.0	9.4	48.9
Potions	2.1	2.0	3.1	16.1	3.7
Tiles Explored	236	230	244	302	328
Times Died	13	22	0	63	169

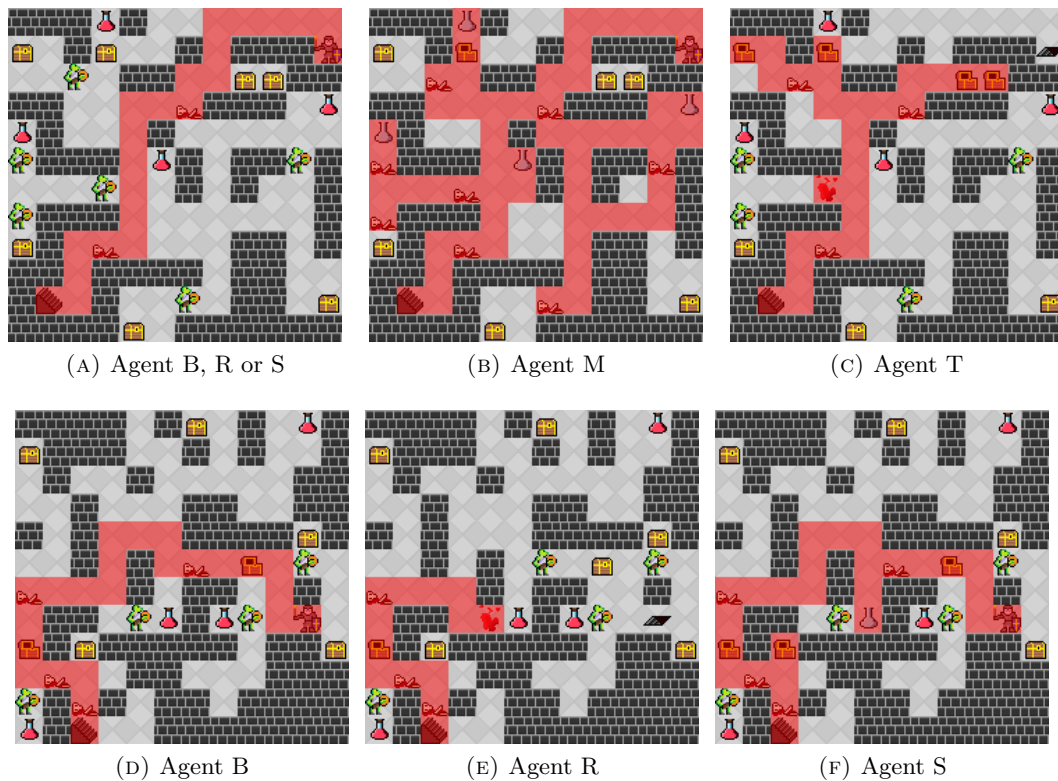


FIGURE 7.4: Some indicative heatmaps of trained agents on Level 2 (Fig. 7.4a–7.4c) and Level 8 (Fig. 7.4d–7.4f). The different playstyles of the agent-personas are showcased, although in Level 2 agents B, R and S all share the heatmap of Fig. 7.4a.

unavoidable monsters between the dungeon entrance and the exit made such a strategy redundant except in special cases (see Fig. 7.4f).

## 7.6 Results

In order to compare player decisions to agent-persona decisions, a simple metric was defined: for each player’s playtrace, we replay the whole game and at each point in time, we input the state description to all of our artificial agents, and compare the

player’s decision to the decision of the different agents. Essentially, we ask: “What would Q do?”. This metric expresses the degree of agreement on next best action between the individual player and the agent-persona. It is directly grounded in the theoretical considerations of decision making outlined above, and assumes that for every given state of the game, an agent-persona that is adequately representative of a player in that particular state will select the same action as the player. More precisely, the metric was calculated as the number of agent-persona/human player agreements  $N_a$  for each decision made in the human decision trace, normalized with respect to the number of decisions in the player’s decision trace  $N$ , i.e.  $N_a/N$ . One advantage of this metric is that it gives a numeric representation of the degree to which an agent adequately represents a player across a level. The utilities of each agent could subsequently be tweaked through iterations of training using a simple hill-climbing approach to maximize the agreement ratio with regard to an individual player or to clusters of players. Another advantage is that the agreement ratios would be easily intelligible to game designers using the agent-personas in a content creation process. In order to test the agent-personas as well as the comparison metrics, a Random Controller was constructed which chose randomly from all legal moves from each game state. This addition investigates to which degree the agent-personas decided and represented players differently from a random agent.

For each level in the user study, each playtrace was examined to determine which agent-persona had the highest agreement, and hence represented the best fit for the playtrace. Table 7.4 indicates the number of times each agent was the best fit for each level. As is evident from the table, most playtraces matched the Treasure Collector (T) persona, while subgroups of players matched other personas. This finding corroborates the observation in Section 7.5.1 of players’ tendency to collect treasures, evidenced by the large proportion of players that collected most (and some all) treasures across levels. This behavior may have stemmed from the treasure counter on the user interface as well as the encouragement of being the “best” in the game. Unfortunately, no post-play qualitative data were collected, which could have helped illuminate individual motivations of players. Although the Treasure Collector persona does seem to dominate the dataset in terms of agreements, all other agents except for the Survivalist have a strong minority representation as best fits. The general relevance of the method is supported by the fact that only a single playtrace was characterized best by the Random Controller (Z).

In order to assess the performance of the best fitting agent-personas for each playtrace, the agreements are visualized in the plot depicted in Fig. 7.5. The plot shows how the agent on average agreed with players on 60%–70% of their decisions. A Mann-Whitney U test unsurprisingly indicated that collectively, the best-fitting agent for each playtrace agreed significantly more with players than the Random Controller ( $W = 155633.5, p < 0.001$ ).

TABLE 7.4: Frequencies of agent-persona best fit across levels.

Level	Agent						Total
	B	R	S	M	T	Z	
1	1			10	27		38
2	2	4		5	27		38
3		5	2		31		38
4	1	2	1		34		38
5		3			35		38
6		3		4	31		38
7	6	3		7	22		38
8	1		4		33		38
9	3	2		1	31	1	38
10	1	7		2	28		38
Total	15	29	7	29	299	1	380

TABLE 7.5: Statistics of the individual agent-personas. All agent-personas attain high maximal values. This indicates that all agents, except for the random controller, are relevant approximations of some players.

Agent	Mean	SD	Max	Min	N
Baseline Player (B)	0.52	0.10	0.94	0.25	15
Runner (R)	0.54	0.09	0.94	0.37	29
Survivalist (S)	0.53	0.11	0.94	0.25	7
Monster Killer (M)	0.54	0.10	0.80	0.23	29
Treasure Collector (T)	0.63	0.11	0.90	0.35	299
Random Controller (Z)	0.43	0.02	0.49	0.37	1

Table 7.5 summarizes the performance of each agent across all levels. The results show that all agents attain a high level of agreement with some playtraces and very low levels of agreement with others. This indicates a variety in the expressed utility functions of the agent-personas, but the fact that the Treasures Collector agent dominates the data set in terms of best fit suggests that this agent possibly could be split into multiple agents to better represent the playtraces for which it is the best fit.

## 7.7 Discussion

The method developed and demonstrated in this paper seems to have a number of attractive characteristics, allowing for the construction of decision making personas and determining to which degree different players agree to them. However, this method also suffers from a number of limitations warranting further work. These limitations concern the data collection method, the agent as an abstract model of human decision makers, and the scalability of the computational approach.

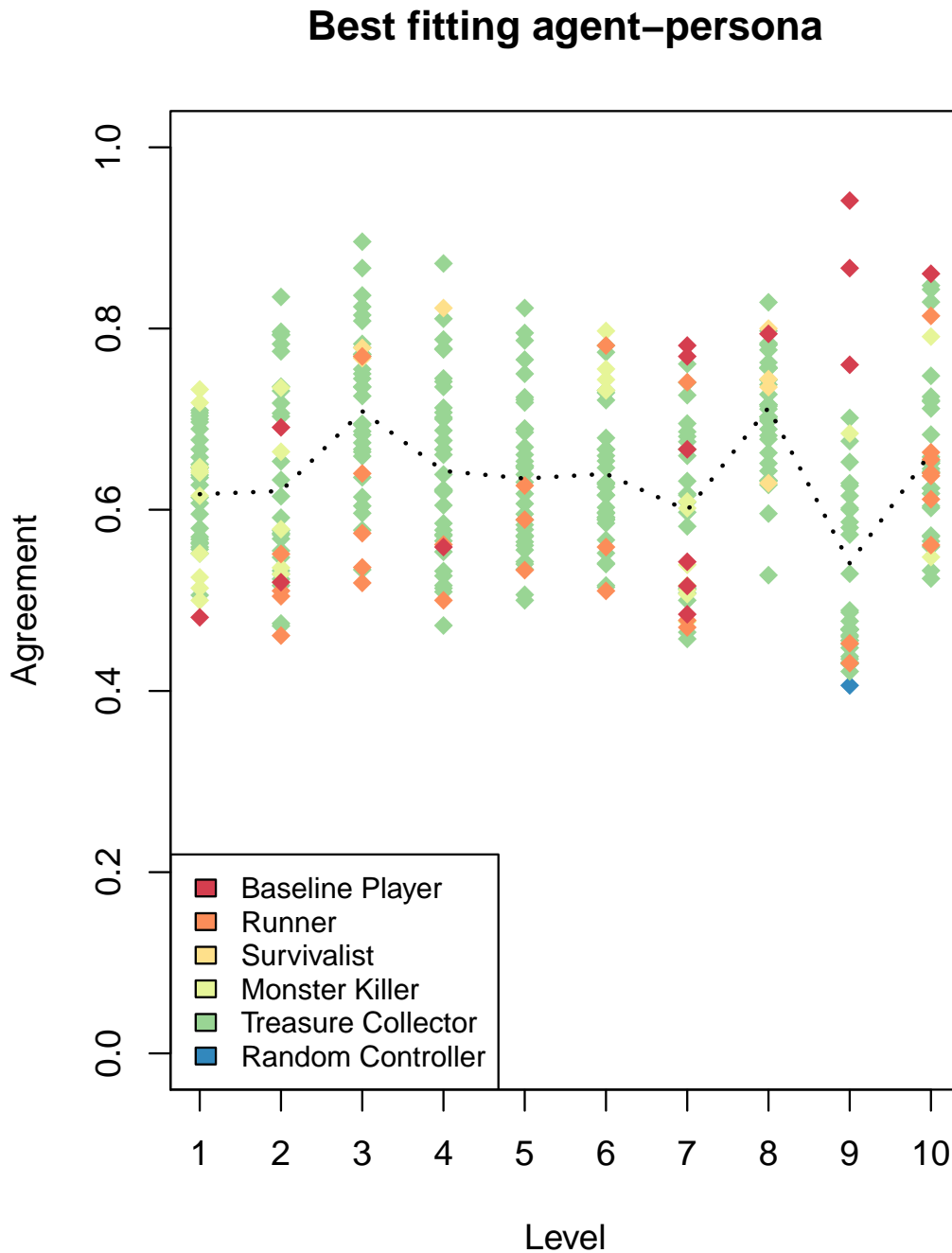


FIGURE 7.5: The best agreeing agent for each level. The dotted line indicates the mean agreement of the best fitting agent-personas across all players on each level.

The applied data collection method sought to enable players to engage with the game in accordance with their individual motivations and hence utilities. As part of this goal, players were given the option of participating in a competition and lottery. The collected data exhibits a predominance of behavior matching the Treasure Collector persona which could be a consequence of players trying to win the competition.

The utility function of the agent-persona is constant over the course of each level, and only one agent-persona is used to characterize a full decision making trace. This means that if the player changes her conception of utility while playing the level, she will quite possibly match several personas during the playthrough. A response to this limitation could be to subdivide decision traces, e.g. via a sliding window, to find the best agent-persona match for each point along the decision making trace. As an extension, this approach could be used to cover all playtraces for an individual human player to investigate which personas are matched across all the player's traces. Relatedly, the utility function of the agent will necessarily be a high-level abstraction of the player's. While this is intentional, other factors influencing the agent's evaluations and learning, such as exploration chance, learning rate and  $\gamma$  value (discount of future rewards) are kept constant in the experiment presented here. Each of these could have, at some level, relevant psychological counterparts such as openness to new experience, ability at learning rules and content, and tendency and motivation to plan ahead; the extend to which these parameters map to human psychology should be explored, and agents with different configurations of these parameters should be tested.

The testbed game used for the development and demonstration of the method has a limited number of affordances that are considered potential components of the player's utility function. Hence, the construction of various agent-personas based on various configurations of these is a manageable task, which can be done manually. For more complex games, the number of affordances may be significantly higher, making it difficult and time consuming to construct agents that cover the space of possible utility configurations to a degree that a good agent-persona match could be found for every player. This affects the scalability of the method, albeit the degree remains unknown at this point. One possible solution could be to use the method for modeling players at a conceptual level and designing content at a sketch level, rather than at a detailed level, though this will naturally depend on the game in question. While the Q-learning agents were demonstrated to work well, the training of the agents is computationally demanding and hence time consuming. The time needed to train the Q-learning agents on an average desktop computer would likely exceed the time a content designer would be willing to wait for agent-based feedback. A better approach would be to use a generic trained agent, whose policy was not tied to a particular level. Possible approaches could include using agents based on Q-learning with neural networks, Monte Carlo Tree Search or evolutionary rule-based systems.

Future work will focus on addressing these limitations, in an attempt to find a faster performing, more accurately representative, and scalable approach to modeling human decision making in the form of generative agents. We will also attempt to adapt the a

priori constructed agents to fit either individual players or, more realistically, generalized representations of players. Such generalized representations could be obtained by clustering players based on their difference from the various agent-personas, and training the closest agent-persona to match the center of the cluster (Holmgård et al., 2013a).

## 7.8 Conclusion

This paper presented a theory-based method of using generative agents as models of human decision making in computer games and explored it in a simple scenario. A theoretical argument for considering agents eligible for representing variations in human decision making processes as agent-personas was presented. To test this argument, a crowdsourced human decision making experiment was conducted using a testbed game. A number of Q-learning agents were developed as agent-personas, and the decision making of human players was compared to the decision making of the agents. The comparison demonstrated that the agents were useful as personas for characterizing and discriminating between the human players. Although the suggested method has a number of limitations in its current form, key findings demonstrate that a high-level abstraction of human decision making, in the form of agents, is possible and can provide useful insights on possible and plausible interactions with game levels, whether hand crafted or procedurally generated. We believe that the method could be of use to player modeling as well as game design and development.

## 7.9 Acknowledgements

We thank the participants of the user study. The research is supported, in part, by the FP7 ICT project C2Learn (project no: 318480).

## Chapter 8

# Evolving Personas for Player Decision Modeling

### Reference:

Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis (2014). “Evolving Personas for Player Decision Modeling”. In: *IEEE Conference on Computational Intelligence and Games*.

### 8.1 Abstract

This paper explores how evolved game playing agents can be used to represent a priori defined archetypical ways of playing a test-bed game, as procedural personas. The end goal of such procedural personas is substituting players when authoring game content manually, procedurally, or both (in a mixed-initiative setting). Building on previous work, we compare the performance of newly evolved agents to agents trained via Q-learning as well as a number of baseline agents. Comparisons are performed on the grounds of game playing ability, generalizability, and conformity among agents. Finally, all agents’ decision making styles are matched to the decision making styles of human players in order to investigate whether the different methods can yield agents who mimic or differ from human decision making in similar ways. The experiments performed in this paper conclude that agents developed from a priori defined objectives can express human decision making styles and that they are more generalizable and versatile than Q-learning and hand-crafted agents.



## 8.2 Introduction

Decision making is a central aspect of almost any interesting agonistic (Caillois, 2001) game, as noted by Sid Meier who famously stated that “a game is a series of interesting choices” (Rollings and Morris, 2004). Typically, play sessions in agonistic games can be described as chains of decisions by one or more players in a proactively and reactively changing environment. Players make decisions while the environment either observes, responds, or proceeds agnostically, or enacts some combination of the three. Capturing, describing, modeling, and reproducing chains of decisions is of interest to game researchers, developers, and players for several reasons.

One reason can be to characterize typical chains of decisions as being representative of certain *decision making styles* in playing particular games. They represent certain ways of navigating the *decision space* of the game at a chosen level of abstraction. The appropriate level of abstraction naturally differs among games and one game may have several levels of abstraction where decision making can be characterized. For instance, playing a game of Mario entails decisions at the aesthetic level (do I prefer fireball or raccoon tail?) and the tactical level (do I attempt to engage or evade enemies?) as well as the atomic level (do I press up or down?). Decision style characterization should hypothetically be possible at all three levels (Holmgård et al., 2013a), and the relevant level of analysis must be determined by the purpose of the decision style characterization.

In this paper we address the problem of modeling human decision making in games in the following two ways: Firstly, we attempt to represent archetypical decision making styles in a test-bed game via game playing agents which we call *procedural personas*. Secondly, we map agent decision making styles to human ones in order to measure to which extent our personas are capable of expressing typical human ways of making decisions within our test-bed game. The chief motivation for the work presented is to use agents that express decision making styles to test manually or procedurally/co-creatively (Yannakakis et al., 2014) generated game content such as levels (Liapis et al., 2013b). This could support the game design process by providing low-cost, low-fidelity mock playtesting of content during development. Such a method could e.g. be of use and interest to level designers during the design of a new level, allowing for quick impressions of what decisions archetypical players might make in the level. By continuously comparing procedural persona behavior with human behavior, and adjusting persona decision making styles accordingly, the personas might be continuously refined during iterative development and playtesting cycles to better represent identified subgroups of human players, as outlined in our previous work (Holmgård et al., 2013a). The envisioned process is illustrated in Fig. 8.1.

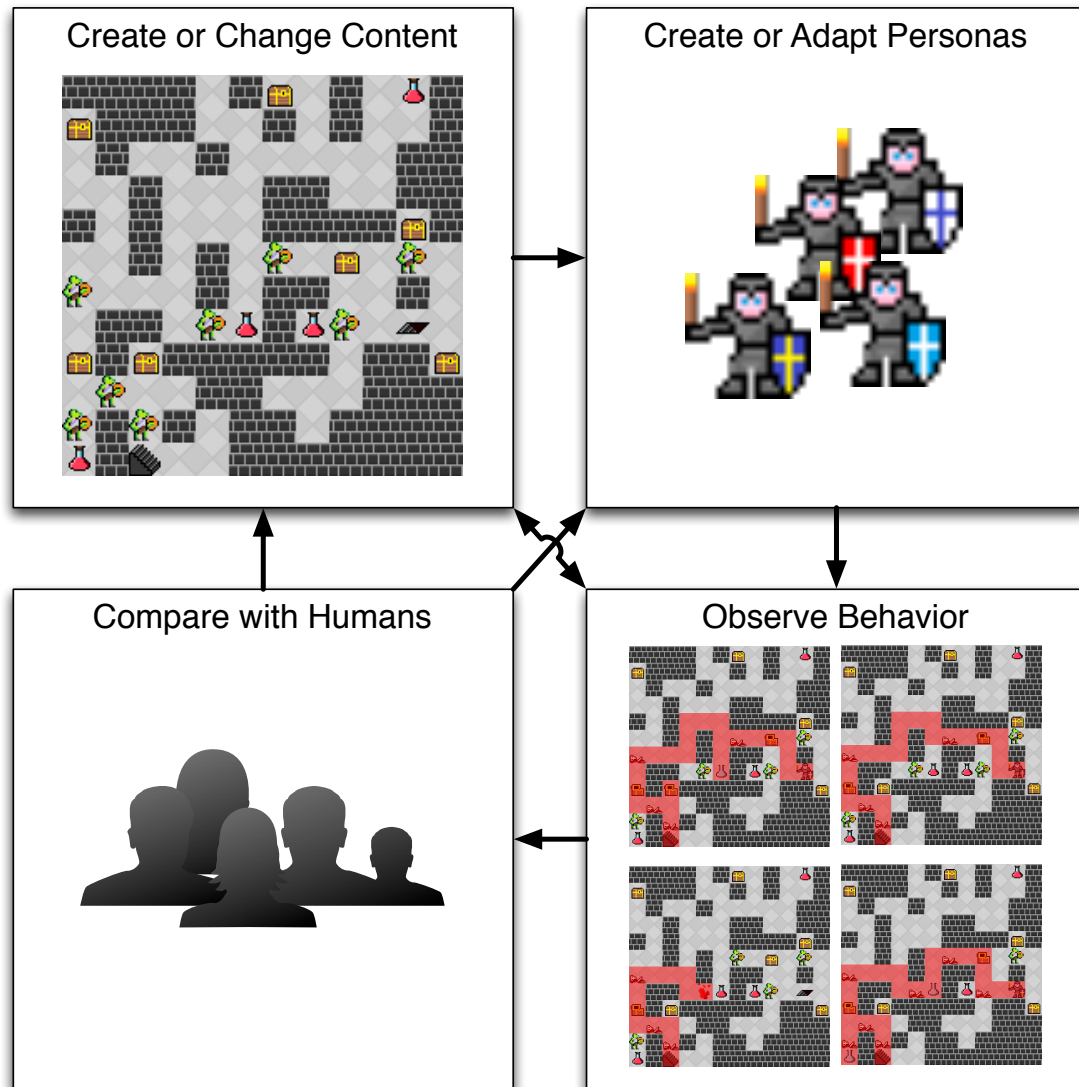


FIGURE 8.1: The intended designer use of procedural personas.

In order to model decision making styles, this paper contributes to and expands on previous work (Holmgård et al., 2014b) by developing personas based on linear perceptrons and comparing their performance against previously developed personas based on Q-learning as well as several baseline agents. Our previous approach was limited in terms of performance, generalizability, and scalability; the approach presented here is an attempt to address these problems. To describe this process, we first present related work and outline how our approach is based on psychological decision theory. Secondly, we briefly present the reinforcement learning experiments which we are building on. Thirdly, we describe our method of and fitness functions for evolving linear perceptrons to represent archetypical decision making styles for procedural personas. Fourthly, we present our experiments and results in using these evolved personas to express decision making styles and capture typical player styles in decision making and comparing them

to the reinforcement learning agents. We conclude by arguing that evolutionary methods are better suited to replicating observed player behavior than previously applied td-learning techniques, and we point out limitations and under-explored aspects of the method.

## 8.3 Related Work

This paper builds upon a theoretical framework of human decision-making; additionally, it is related to player modeling as well as the simulation-based evaluation of procedurally generated content. A brief survey of these domains follows.

### 8.3.1 Decision Theory

As described in Holmgård et al. (2014b), decision theory (Kahneman and Tversky, 1979) deals with human decision making under risk and uncertainty. One of its fundamental assumptions is that human decision making can be described as being shaped by utility. Briefly stated, utility captures how much an expected outcome of a decision is worth to the decision maker versus the expected cost and risk of attaining that outcome. Humans typically attempt to optimize the utility gained from a decision which is then considered rational action. Research in decision theory has shown that the nature of this optimization process is shaped by the utility expected from the decision, meaning that the assignment of cognitive resources and the balance between heuristic and analytic reasoning is based on the perceived importance of the decision (Gigerenzer and Gaissmaier, 2011). The decision making process happens under *bounded rationality* (Rubinstein, 1998). Generally, utility is considered idiosyncratic and decision theory does not try to explain why a given outcome has utility to the decision maker (though other directions in psychology such as personality psychology (Yee, 2006) or motivational psychology (Canossa, 2012) might be helpful in explaining this). Instead, it looks at the decision maker's tendency to take risks to attain particular outcomes. This means that for real world decision making problems, the possible sources of utility are practically infinite though often context can be used to identify *probable* sources of utility.

Games, and certain computer games in particular, can be considered special, limited cases of decision making problems, when the game's decision space is delineated by its rules and mechanics. As a game becomes more complex this decision space of course expands and complexifies rapidly. However, knowing the rules and mechanics of a game provides a well-defined context for making assumptions about possible sources of utility in the game. A game's stated goals and the possibilities inherent in the game's mechanics

constitute affordances (Gibson, 1977) which are likely to be of utility to the player, since they are typically the very reason for playing the game in the first place. By analyzing the mechanics of a game we should be able to detect likely sources of utility, though due to the idiosyncratic nature of utilities we can never be certain to have covered all cases. This is relevant to our purpose of developing decision making procedural personas, as the analysis of the affordances in a game can provide us with a list of possible goals to direct the behaviors of procedural personas. Once a hierarchy of goals, representing sources of utility, has been established for a persona, we can effectively use this as a representation or metaphor for a decision making style.

In the following section, we describe how this approach can be used to enable a form of player modeling which seems to be relatively underexplored in the academic literature, though perhaps more common for ad hoc industry purposes (Smith et al., 2011).

### 8.3.2 Player Modeling

Since this work aims to represent archetypical decision making styles in our test-bed game, each resulting persona can be considered an individual player model. Smith et al. (2011) provide a useful inclusive taxonomy of player modeling methodologies. The work categorizes player modeling techniques via four different facets: the *scope*, *purpose*, *domain*, and *source* of the player model. **Scope** determines the generalizability of the player model. For this work, the scope of each model is limited to the game in question, since the decisions and utilities are contingent on the particular game, in this case MiniDungeons. **Purpose** refers to the intended use of the model. Our method is generative in the sense that the final intent is to express decision making styles in games, either styles defined a priori by designers or styles adapted to match human styles observed across groups or from individual playthroughs. **Domain** refers to what the model generates, in this case player decisions expressed through in-game actions at the same level that human players would. **Source** refers to the motivation or substrate from which the player model is derived. The models generated from our approach are hybrids in the taxonomy. They are initially *interpreted* in the sense that the a priori personas are developed by the game’s designers based on expert knowledge about typical decision making styles of human players, but aim to grow empirically *induced* in the sense that they are partly evaluated on how well they express the decision making styles of actual human players and ultimately should evolve to adapt to these. As such, the method attempts to achieve player modeling by evolving from game designer interpreted personas to player data induced personas, bridging the designer’s expert knowledge and empirical play data. Approaching the problem from an alternative framework by Yannakakis et al. (2013) our method combines a way to move iteratively from a *model-based* (designer

centric) player model to a *model-free* (data centric) player model, creating a *hybrid* player model.

### 8.3.3 Procedural Content Generation

As stated above, the main goal of the procedural persona method is to provide low-cost, low-fidelity mock playtesting in a manner that is useful in supporting game content creation. This may be useful to a human designer manually creating a piece of content in an editor, but human designers are, to some extent, capable of informally mentally simulating different decision making styles their content might enable. We would argue that the procedural persona method could potentially be of greater use to search based procedural content generation processes (Togelius et al., 2011) that are either wholly procedural or based on mixed-initiative co-creative processes where a human designer and an AI-driven support tool collaboratively produce content (Liapis et al., 2013a). Human designers might use procedural personas as input to a co-creative process, controlling the AI's search for novel content by asking it to generate content that fits certain decision making styles.

## 8.4 Previous Work

In this section we briefly present the previous work this paper builds on, and the testbed game on which the experiments were performed.

### 8.4.1 MiniDungeons

The test-bed game used, *MiniDungeons*, implements the fundamental mechanics of a roguelike dungeon exploration game. The turn-based game puts the player in a top-down viewed tile-based dungeon (of 12 by 12 tiles) containing monsters, potions, and treasures, as displayed in Fig. 8.3. Impassable tiles constitute the walls of the dungeon, while passable tiles may contain enemies or items for the player. All of the level is visible to the player who can move freely between passable tiles. When the player moves to a tile occupied by a monster or item, immediately the monster is fought or the item is collected and applied. The player has a 40 *hit point* (HP) health counter and dies if this drops to zero. Monsters randomly deal between 5 and 14 HP of damage while potions heal 10 HP up to the maximum value of 40 HP. Treasures have no game mechanical effect other than adding to a counter of collected treasures which is displayed to the player. The game contains 10 levels (see Fig. 8.2) and a tutorial level. Excepting the

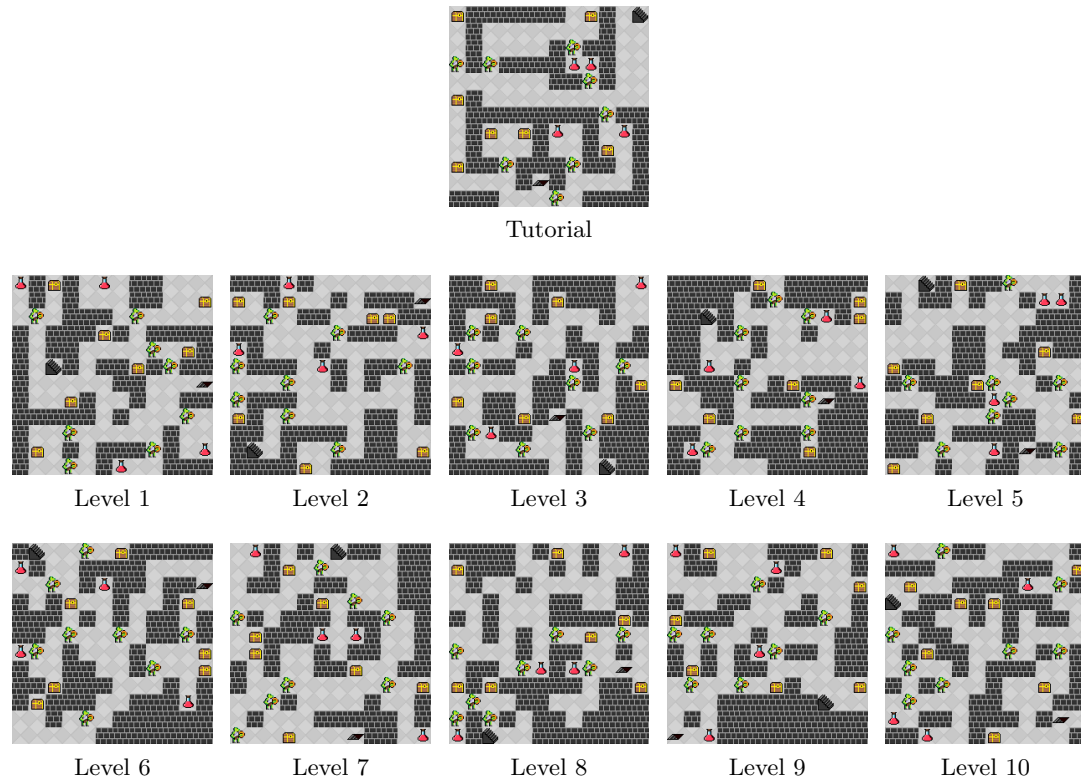


FIGURE 8.2: The levels included in MiniDungeons. The tutorial level is hand-crafted, and could be played multiple times. The “real” levels (1-10) were played only once (no retries if the hero died) and were created in a mixed-initiative fashion.

tutorial level, all levels are generated using the multi-genre mixed-initiative co-creation tool Sentient Sketchbook (Liapis et al., 2013c). For further details on the test-bed game and discussion of its properties, we refer to our previous work (Holmgård et al., 2014b).

### 8.4.2 Previous Experiments with Q-learning

Our previous work demonstrated a proof-of-concept for the idea of training procedural personas to express decision making styles using temporal difference-based (td-based) reinforcement learning, specifically Q-learning.

A data set of 380 human play traces from MiniDungeons was collected and used as a reference for determining to which extent the defined personas expressed actual human decision making styles. The resulting personas matched the human players’ decisions with an average precision of 78%, a result that would not have been feasible with any one persona alone.

The specification and logic of the personas was straightforward and intuitive, as the reinforcing rewards given to the agents during training worked as a direct metaphor for the personas’ respective utilities. For instance a “Monster Killer” type persona was



FIGURE 8.3: The MiniDungeons interface while a game is played.

given a large reward for every monster killed in a given level and a smaller reward for reaching the exit. Personas trained on this reward configuration consistently expressed decisions that we would argue are subjectively interpretable as a “Monster Killer” style.

Unfortunately, the technique of using td-based reinforcement learning suffers from a number of issues that precludes it from being useful for the practical purpose of on-line interactive content creation. Firstly, the method is computationally expensive as it requires the Q-learning agent to run through a significant number of simulations to learn the appropriate Q-table, amounting to hours worth of training time on a modern

desktop computer to train a single persona for a single level. Secondly, given the applied technique of Q-learning, where agents were trained on an only slightly abstracted version of the state space, the personas do not generalize across levels. This necessitates retraining of the personas whenever the level content is changed, further exacerbating the problem of practical applicability. Thirdly, the Q-learning agents would not provide useful starting points for adapting the personas to observed human behavior as it would most likely be more efficient to train new agents using inverse reinforcement learning to represent groups of human players' or individual human players' decision making styles. This would again make it impractical to implement the desired iterative hybridization of the designer specified interpreted model and the observation based induced model. Though optimizations, such as applying active learning to the Q-learning process, might possibly mediate these drawbacks, the sum of the concerns listed above motivates us to attempt to replicate the results of the td-based reinforcement learning technique with faster, evolution-based methods.

## 8.5 Methods

In the following section we describe how we achieve this by evolving linear perceptrons selecting which of the currently available affordances to act on. While these agents are structured very differently, they enact decision making styles equivalent to those of the Q-learning agents trained on observation of the game's state space.

### 8.5.1 Evolved Controllers

In order to control the personas in MiniDungeons and express the desired decision making styles, 7 linear perceptrons are combined into an evolvable controller. The perceptrons take 8 inputs in addition to the bias, and through weighted sums produce the 7 outputs. The 8 inputs consist of the hero's current hit points (1 to 40) and 7 distance measures derived from A\* path finding (with Manhattan Distance heuristic) in the maze: the distance to the nearest monster, the distance to the nearest treasure, the distance to the nearest treasure while avoiding monsters, the distance to the nearest potion, the distance to the nearest potion while avoiding monsters, the distance to the level exit, and the distance to the exit while avoiding monsters. The inputs are chosen under the assumption that human players will typically survey the whole play area and pick from the available paths to the various affordances in the level. The distance to each affordance type, accepting and avoiding the risk of fighting a monster, are then considered an acceptable abstraction of the game state into a number of utility providing options that the player can choose from. In the same vein, each of the linear perceptrons are



mapped to represent one of the available strategies, with or without risk taking: pursuing the nearest monster, pursuing the nearest treasure, pursuing the nearest treasure while avoiding monsters, pursuing the nearest potion, pursuing the nearest potion while avoiding monsters, pursuing the exit, or pursuing the exit while avoiding monsters. The controller re-evaluates the state of the game for each step. The network is fed forward, and the linear perceptron with the highest activation value is identified and from the corresponding affordance, the next step in the path is selected. In the case that an affordance is unavailable, e.g. if all paths to the nearest treasure, avoiding monsters, is blocked, the next ranked affordance, based on activation, is selected, and so on, until the controller ultimately picks the risky path to the exit as a final fall-back affordance.

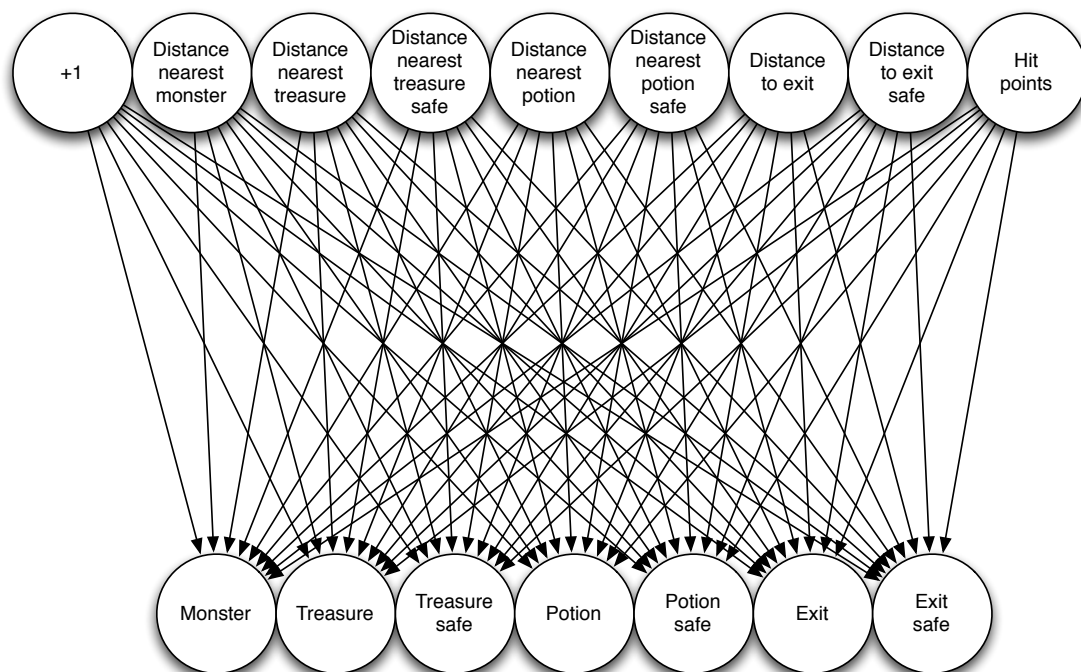


FIGURE 8.4: The controller network which was evolved to generate the five personas.

### 8.5.2 Evolutionary algorithm

We use a  $(\mu + \alpha)$  evolution strategy without self-adaptation. This is a truncation-based evolutionary algorithm which for each generation retains the 50% best performing individuals, discards the lowest performing half, and produces single-parent offspring from the remaining individuals to maintain the population size. Finally all population individuals are mutated, except for members of an elite group, consisting of the top performing 2% of the population, which remains unchanged. Mutation is accomplished by changing each connection weight in the network with a random number drawn from a Gaussian distribution centered around zero with a standard deviation of 0.3. All experiments are done using a population size of 100 individuals, trained for 100 generations. One set

of personas are evolved for one level specifically for 100 generations. Another, generalized, set of personas are evolved by playing on 9 of the 10 levels, keeping the 10th level unseen. As such, each generation of these generalized agents is exposed to 9 times the level content compared to the level specific personas.

### 8.5.3 Fitness Functions for Evolving Personas

As noted above, the procedural personas are evolved to represent archetypical decision making styles, motivated by utilities. In an effort to represent this in the personas, the fitness functions used to evolve the linear perceptrons are constructed as compounds of the relative importance of each potential source of utility in the game and the persona's ability to achieve these utilities. Five different potential sources of positive or negative utility were identified in the MiniDungeons game, based on an analysis of the game's mechanics: making a *move*, fighting a *monster*, collecting a *treasure*, *dying*, and reaching the *exit* of the level. Collecting an HP restoring potion could have been considered a source of utility in itself, but was not included, as it was considered subsumed under the other sources of utility. This resulted in five distinct personas being defined: The *Exit* persona simply tries to reach the exit of the level, representing a player who mainly cares about progressing through the levels of the game. The *Runner* persona attempts to reach the exit in the fewest steps possible, representing a step-optimizing "speed runner". The *Survivalist* avoids damage to the largest extent possible, while moving toward the exit, representing a conservative player who does not like to lose. The *Monster Killer* represents an aggressive player by seeking out and fighting every monster in the level, while attempting to reach the exit without dying. The *Treasure Collector* represents a completionist player who cares about collecting every treasure, before progressing to the next level. The specific utilities of the personas were set to mirror the rewards given to our Q-learning agents in our previous work (Holmgård et al., 2014b) as closely as possible. The individual values are presented in Table 8.1. Collecting treasure and killing monsters are associated with positive utility, while moving or dying are associated with negative utility for the relevant personas. All personas derive a slight amount of negative utility from each move made, in order to ensure progression through the level. The value subtracted is doubled for the Runner persona.

Agents' fitness scores are calculated by dividing the amount of utility obtained by the individual during the playthrough by the maximally attainable utility for the level in question. The only exception to this rule is the number of moves made, which is not normalized. The sources of utility depend on the persona and are shown in Table 8.1 along with their utility weights. For personas evolved on a single level, the fitness is computed for each playthrough of the level. For personas evolved across multiple levels,

the fitness for each level played is first computed after which the mean across all 9 played levels is computed and used as the fitness score for the individual in that generation.

#### 8.5.4 Decision-level playtrace comparison

To determine the degree to which the decision making styles of the evolved personas match the previous reinforcement learning trained personas, as well as actual human decision making styles, a simple measure of agreement is used. For each human play trace, we replay the whole game and at each decision point (in this case every action), we input the state description to all of our artificial agents, and compare the player's decision to the decisions of the different agents. Essentially, we query each agent "What would you do, given this situation?". The resulting metric is a simple count of agreements, normalized to a ratio from 0 to 1 by the number of decisions in the human play trace.

#### 8.5.5 Baseline Agents

In order to allow for a fair comparison of the performance of the reinforcement learning trained agents and the evolved agents, a number of baseline agents are constructed. The simplest one of these is a random controller that every step picks a legal (i.e. leading to a passable tile) decision in the level. Additionally, five more advanced baseline agents are constructed using a finite state machine on top of the A\* algorithm for pathfinding. These agents act single-mindedly by always making a decision for following the shortest path toward one primary objective until this objective is exhausted, after which they follow the shortest path to the exit. Their objectives are to pursue either monsters, treasure, or the exit. The treasure and exit objectives are implemented in two modes: one where the baseline agent tries to avoid monsters along the path if possible and one where monsters are ignored and fought if they are present along the shortest path. The baseline agents are not considered personas as such, since their behaviors are too simple, but are included in an attempt to provide baselines for simple random moves, and for single-mindedly following paths to classes of affordances in the game. The complete list of personas tested in our experiments is provided in Table 8.1 along with the parameters for their training or evolution.

## 8.6 Results

To verify that the developed personas actually exhibit decision making styles in accordance with their intended persona identity, Table 8.3 presents the performance of each

TABLE 8.1: All personas tested in the experiments.

Persona	Code	Utility weight of tiles or events				
		Moved	Monster	Treasure	Death	Exit
Q-learning						
Exit	$q_E$					0.5
Runner	$q_R$	-0.01				0.5
Survivalist	$q_S$				-1	0.5
Monster Killer	$q_M$		1			0.5
Treasure Coll.	$q_T$			1		0.5
Evolution						
Exit	$e_E$	-0.01				0.5
Runner	$e_R$	-0.02				0.5
Survivalist	$e_S$	-0.01			-1	0.5
Monster Killer	$e_M$	-0.01	1			0.5
Treasure Coll.	$e_T$	-0.01		1		0.5

TABLE 8.2: Baseline agents used in experiments.

Baseline Agent	Code	Method	Primary Objective
Monster Killer	$a_M$	A* ignoring monsters	Nearest monster
Runner	$a_R$	A* ignoring monsters	Exit
Treasure Collector	$a_T$	A* ignoring monsters	Nearest treasure
Runner Safe	$a_{R,s}$	A* avoiding monsters	Exit
Treasure Collector Safe	$a_{T,s}$	A* avoiding monsters	Nearest treasure
Random Controller	$Z$	Random legal move	None

individual agent across all levels. This verification is necessarily a process of subjective interpretation. From Table 8.3 we can identify that both Q-learning based and evolved Exit personas engage in little combat. Additionally they exhibit a relatively low degree of exploration. The Monster Killer personas engage with the most monsters across methods, exhibiting the desired decision making style. Interestingly, the evolved Monster Killer personas tend to collect more potions than the Q-learning trained Monster Killer and also succeed in killing more monsters. The Runner persona exhibits a relatively low exploration value. In the case of the Q-learning agent, the exploration is lower than most other personas, but significantly higher than the survivalist persona, while in the case of the evolved personas it is practically tied with the survivalist. The special case of the Q-learning survivalist can be attributed to the fact that in some cases that persona opts to stop progressing when faced with monsters blocking its path, as is also evident from the low number of monsters killed. The evolved survivalist on the other hand proceeds to fight monsters when this is the only available course of action to reach the exit. Given the restricted nature of the play environment the Runner and Survivalist generally overlap in their performance statistics, as the safest and fastest paths to the exit typically deviate with only a few steps. The Treasure Collector personas unsurprisingly consistently exhibit the greatest collection of treasures as well as the greatest

TABLE 8.3: Average statistics of each persona’s play traces across all levels in MiniDungeons. M stands for monsters killed, T for treasures collected, P for potions drunk, Ex for tiles explored, and D for times died. With the exception of D, values are averaged across 20 test runs; D includes all playthroughs tested.

Persona or Agent	M	T	P	Ex	D
$q_E$ Exit	22.8	9.4	2.1	237.9	8
$q_M$ Monster Killer	54.4	9.0	15.9	300.3	66
$q_R$ Runner	22.4	7.7	2.0	231.8	19
$q_S$ Survivalist	4.0	5.0	1.0	135.0	0
$q_T$ Treasure Collector	48.6	49.4	3.9	335.7	167
$e_E$ Exit	22.1	5.5	1.6	217.0	10
$e_M$ Monster Killer	69.5	10.3	30.9	413.9	89
$e_R$ Runner	23.1	6.5	1.4	230.2	15
$e_S$ Survivalist	21.6	7.2	1.2	227.2	2
$e_T$ Treasure Collector	50.1	57.5	6.5	413.9	151
$\Sigma e_E$ Exit	24.2	4.5	1.4	217.7	30
$\Sigma e_M$ Monster Killer	75.5	10.4	37.0	454.8	102
$\Sigma e_R$ Runner	24.1	4.7	1.7	216.8	27
$\Sigma e_S$ Survivalist	24.1	5.0	1.9	221.0	23
$\Sigma e_T$ Treasure Collector	58.9	60.4	23.1	493.4	93
$a_M$ Monster Killer	49.2	3.8	1.2	200.6	200
$a_R$ Runner	24.3	4.8	1.7	218.9	25
$a_{R,s}$ Runner Safe	19.0	4.5	1.4	225.4	31
$a_T$ Treasure Collector	48.2	49.1	1.6	329.9	190
$a_{T,s}$ Treasure Collector Safe	44.7	57.5	3.0	421.9	122
$Z$ Random Controller	58.5	38.4	19.9	521.2	123

exploration of the game levels. The Q-learning trained Treasure Collector and the level specifically evolved Treasure Collector seem relatively comparable. Notably, the generalized, evolved Treasure Collector performs better than both of them and picks up more potions on its way through the level. The A\* based baseline agents generally perform worse than their persona counterparts, but following comparative strategies, while the random controller has the largest exploration ratio of all personas and agents, since the controller just roams the map until it randomly reaches the exit, runs out of allocated testing actions, or dies. A set of indicative heatmaps from level 7 is included in Fig. 8.5, showing the varied behaviors of the personas and agents.

### 8.6.1 Agreements between Q-learning personas and evolved personas

As mentioned above, the evolved personas’ fitness functions are designed in an attempt to make them emulate the decision making styles expressed by the Q-learning trained personas. To test whether this is accomplished, the decision-level playtrace comparison method described in Sec. 8.5.4 is applied between similar personas, across all 380 collected playtraces, using the human decisions as the baseline. This comparison is not



FIGURE 8.5: Heatmaps of all personas and baseline agents on level 7 of MiniDungeons.

made between Q-learning personas and generalized, evolved personas, since these would have to be tested on levels unseen to them, but seen to the Q-learning personas. For each step in each human play trace, the game is advanced to the game state from which the human play trace was collected. The game state is then input to the comparable pair of personas, and both are queried for their next action. If they report the same action, even if in disagreement with the human player, this is counted as an inter-persona agreement. If they report different actions, irregardless of the human choice, this is counted as an inter-persona disagreement. The results are presented in Table 8.4. As is evident, all

TABLE 8.4: Agreements between individual personas based on human players' play traces.

Q-learning Persona	Evolved Persona	Agreement
$q_E$ Exit	$e_E$ Exit	0.71
$q_M$ Monster Killer	$e_M$ Monster Killer	0.69
$q_R$ Runner	$e_R$ Runner	0.72
$q_S$ Survivalist	$e_S$ Survivalist	0.63
$q_T$ Treasure Collector	$e_T$ Treasure Collector	0.70

TABLE 8.5: Agreements between personas/baseline agents and human players. Evolved personas which were generalized by holding out the test level are marked with a  $\Sigma$ .

Q-learning Persona	Agreement	Evolved Persona	Agreement
$q_E$ Exit	0.52	$e_E$ Exit	0.55
$q_R$ Runner	0.53	$e_R$ Runner	0.56
$q_S$ Survivalist	0.49	$e_S$ Survivalist	0.57
$q_M$ Monster Killer	0.54	$e_M$ Monster Killer	0.59
$q_T$ Treasure Collector	0.62	$e_T$ Treasure Collector	0.71
Baseline Agent	Agreement	Generalized persona	Agreement
$a_M$ Monster Killer	0.53	$\Sigma e_E$ Exit	0.55
$a_R$ Runner	0.55	$\Sigma e_R$ Runner	0.55
$a_{R,s}$ Runner Safe	0.55	$\Sigma e_S$ Survivalist	0.56
$a_T$ Treasure Collector	0.71	$\Sigma e_M$ Monster Killer	0.60
$a_{T,s}$ Treasure Collector Safe	0.70	$\Sigma e_T$ Treasure Collector	0.74
$Z$ Random Controller	0.43		

types of personas seem to exhibit agreements at levels ranging from approximately 60% to 70%. Though better than random performance, this indicates that the weightings of utilities cannot be naively transferred from one method to the other.

### 8.6.2 Agreements between personas and human players

To ascertain to which extent the individual personas express actual human player decision making styles all personas (as well as baseline agents) are compared to the human play data, again using the decision-level playtrace comparison method. In this configuration, agent reports are only counted as persona-human agreements if they report exactly the same action that is present in the human play trace, in response to the game state. The results for each agent, averaged across all human play traces on all levels, are presented in Table 8.5.

From Table 8.5 it is clear that the average action agreement ratio across all players seems limited, perhaps with the exception of the Treasure Collector personas and baseline agents. This is, however, to be expected, as each persona is should express a distinct

TABLE 8.6: Frequencies of personas and baseline agents being the best match for individual human play traces.

Persona or Agent	1	2	3	4	5	6	7	8	9	10	Total
$q_E$ Exit	0	0	0	0	0	0	1	0	3	1	5
$q_R$ Runner	0	0	2	0	0	1	1	0	0	0	4
$q_S$ Survivalist	0	0	0	1	0	0	0	0	0	0	1
$q_T$ Treasure Collector	1	2	1	1	4	1	2	1	0	2	15
$e_E$ Exit	0	0	0	0	0	0	1	0	0	1	2
$e_R$ Runner	0	0	0	0	0	0	2	0	0	0	2
$e_S$ Survivalist	0	0	0	0	0	0	0	1	0	0	1
$e_M$ Monster Killer	8	0	0	1	0	0	1	0	3	0	13
$e_T$ Treasure Collector	0	15	19	13	10	12	1	11	7	3	91
$\Sigma e_M$ Monster Killer	0	0	0	1	1	0	6	1	0	0	9
$\Sigma e_T$ Treasure Collector	28	7	13	20	19	22	21	23	18	23	194
$a_{E,s}$ Exit Safe	0	1	0	0	0	0	0	0	0	0	1
$a_T$ Treasure Collector	1	7	1	0	4	2	0	0	1	1	17
$a_{T,s}$ Treasure Collector Safe	0	6	2	1	0	0	2	1	6	7	25
$Z$ Random Controller	0	0	0	0	0	0	0	0	0	0	0
Total	38	38	38	38	38	38	38	38	38	38	380

decision style which should only be displayed by some human players. Therefore, averaging the agreements across all players on all levels obscures the human decision making style expressiveness of each persona. To mitigate this problem, we instead map each human play trace to the persona which agrees the most with it, or put differently, the *best matching persona*. Table 8.6 shows the frequency with which each persona or agent is the best match for the human players. The results indicate that treasure seeking personas dominate the covering of the human play traces, followed by monster killing, and finally the exit seeking strategies expressed; Survivalist and Runner personas only cover a few human play traces. The baseline agents also cover some human playtraces. This could be attributed to the limited decision space of the game and the fact that treasure collecting is a common culturally reinforced affordance for roguelike games; without further information about the players, however, this remains speculation. Two notable characteristics of the results, however, is that the evolved personas generally cover more of the human play trace sample than the other methods, and that the generalized, evolved agents tend to attain better coverage of the human play traces than the level specific ones. This could speculatively be attributed to the fact that the generalized evolved agents have learned from a broader range of examples.

While the frequency table provides insight into how much of the human sample each persona/agent covers, it is also relevant to investigate the quality of each persona/agent's coverage. Table 8.7 displays the mean agreement for each persona or agent. The mean is calculated over all human playtraces for which the persona or agent was the best match, for every level. From the results it is clear that the best matching personas agree with



TABLE 8.7: Mean agreement ratios for personas and baseline agents when they were selected as best matches.

Persona or Agent	1	2	3	4	5	6	7	8	9	10
$q_E$							0.78	0.86	0.86	
$q_R$			0.70			0.78	0.74			
$q_S$				0.60						
$q_T$	0.65	0.71	0.77	0.78	0.79	0.62	0.66	0.74		0.85
$e_E$							0.88			0.85
$e_R$							0.72			
$e_S$								0.82		
$e_M$	0.72			0.79			0.69		0.74	
$e_T$		0.79	0.80	0.73	0.76	0.81	0.66	0.79	0.74	0.86
$\Sigma e_M$				0.70	0.72		0.75	0.80		
$\Sigma e_T$	0.75	0.71	0.83	0.77	0.75	0.78	0.72	0.83	0.76	0.82
$a_{E,s}$		0.71								
$a_T$	0.69	0.76	0.86		0.73	0.77			0.72	0.87
$a_{T,s}$		0.75	0.79	0.78			0.66	0.79	0.81	0.83

TABLE 8.8: Mean agreement ratio across all traces from all levels where personas or agents were selected as best matches.

Persona or Agent	n	Mean	SD	Min	Max
$q_E$ Exit	5	0.84	0.07	0.76	0.94
$q_R$ Runner	4	0.73	0.06	0.64	0.78
$q_S$ Survivalist	1	0.60		0.60	0.60
$q_T$ Treasure Collector	15	0.74	0.07	0.62	0.85
$e_E$ Exit	2	0.87	0.02	0.85	0.88
$e_R$ Runner	2	0.72	0.16	0.61	0.83
$e_S$ Survivalist	1	0.82		0.82	0.82
$e_M$ Monster Killer	13	0.73	0.05	0.65	0.84
$e_T$ Treasure Collector	91	0.78	0.07	0.56	0.95
$\Sigma e_M$ Monster Killer	9	0.75	0.06	0.64	0.82
$\Sigma e_T$ Treasure Collector	194	0.78	0.07	0.57	0.95
$a_{E,s}$ Exit Safe	1	0.71		0.71	0.71
$a_T$ Treasure Collector	17	0.76	0.06	0.67	0.87
$a_{T,s}$ Treasure Collector Safe	25	0.79	0.07	0.63	0.90

human players on between 60% and 88% of decisions, with a great deal of variation across levels. In order to better capture each persona's agreement with humans, Table 8.8 lists the mean agreement values for each persona across all play traces on all levels. Generally, the personas display mean performances ranging from 0.7 to 0.8, which is comparable to the baseline agents in the relatively few cases that they attain a best match, but still leaves room for improvement at matching human decision making styles.

Finally, Table 8.9 compares the mean agreements for best matches between Q-learning and evolved personas designed to express the same decision making style. For most personas, not enough human play traces were matched for robust comparisons. The

TABLE 8.9: Wilcoxon Rank Sum Test for differences between persona pairs, with personas created either via Q-learning or evolution.

Persona	n	Persona	n	W	p
$q_E$ Exit	5	$e_E$ Exit	2	6.0	0.86
$q_R$ Runner	4	$e_R$ Runner	2	4.0	1.00
$q_S$ Survivalist	1	$e_S$ Survivalist	1	1.0	1.00
$q_M$ Monster Killer	0	$q_M$ Monster Killer	13		
$q_T$ Treasure Collector	15	$e_T$ Treasure Collector	91	880.5	0.07
$q_E$ Exit	5	$\Sigma e_E$ Exit	0		
$q_R$ Runner	4	$\Sigma e_R$ Runner	0		
$q_S$ Survivalist	1	$\Sigma e_S$ Survivalist	0		
$q_M$ Monster Killer	0	$\Sigma e_M$ Monster Killer	9		
$q_T$ Treasure Collector	15	$\Sigma e_T$ Treasure Collector	194	1801.0	0.13
$e_E$ Exit	2	$\Sigma e_E$ Exit	0		
$e_R$ Runner	2	$\Sigma e_R$ Runner	0		
$e_S$ Survivalist	1	$\Sigma e_S$ Survivalist	0		
$e_M$ Monster Killer	13	$\Sigma e_M$ Monster Killer	9	67.0	0.60
$e_T$ Treasure Collector	91	$\Sigma e_T$ Treasure Collector	194	8553.5	0.67

Treasure Collectors, however, show a borderline significant difference between  $q_T$  and  $e_T$  using Wilcoxon’s Rank Sum test ( $\alpha$ -level 0.05). Comparisons are not made between baseline agents and personas, since the baseline agents are not based on utilities.

## 8.7 Discussion

It could be argued that the testbed game we employ is too simple and not representative of actual games. However, we argue that the game is similar in complexity to many of a new wave of roguelike games that have recently become popular on hand-held devices - the likes of *868-hack* (Brough, 2013), *Hoplite* (Cowley et al., 2013) and *Out There* (Mi-Clos Studio, 2014). Those games cannot be considered “toy problems” more than any other successful game. This is not to say that MiniDungeons is as entertaining or “deep” (in game design terms) as those games. Future work should include iterating over the MiniDungeons design to provide more satisfying gameplay, which will help us collect more and better player data.

## 8.8 Conclusion

We have addressed the problem of creating procedural personas, which are generalized generative player models that represent the behavior of a class of players with particular playing styles or decision making styles. Based on an analysis of the affordances in a

simple roguelike game, we identified five different reward structures, which were used in the training of personas. A persona representation was devised based on an evolvable perceptron that selects which immediate goal to pursue based on knowledge of internal state and distances to various level features. This evolutionary persona representation was compared with a previously devised method based on Q-learning, and it was found that the evolutionary solution is better both at agreeing with human players and optimizing the rewards, while also being generalizable to unseen levels. These models are well-suited for e.g. simulation-based testing in procedural content generation.

## 8.9 Acknowledgements

We thank the players of the game. The research is supported by the FP7 ICT project C2Learn (project no: 318480).

## Chapter 9

# Personas versus Clones for Player Decision Modeling

### Reference:

Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N Yannakakis (2014). “Personas versus Clones for Player Decision Modeling”. In: *13th International Conference on Entertainment Computing (ICEC 2014)*. Springer, pp. 159–166.

### 9.1 Abstract

The current paper investigates multiple approaches to modeling human decision making styles for procedural play-testing. Building on decision and persona theory we evolve game playing agents representing human decision making styles. Three kinds of agents are evolved from the same representation: procedural personas, evolved from game designer expert knowledge, clones, evolved from observations of human play and aimed at general behavioral replication, and specialized agents, also evolved from observation, but aimed at determining the maximal behavioral replication ability of the representation. These three methods are then compared on their ability to represent individual human decision makers. Comparisons are conducted using three different proposed metrics that address the problem of matching decisions at the action, tactical, and strategic levels. Results indicate that a small gallery of personas evolved from designer intuitions can capture human decision making styles equally well as clones evolved from human play-traces for the testbed game MiniDungeons.

## 9.2 Introduction

This paper investigates how to create models of human decision making styles in games using generative, game-playing agents for procedural play-testing. It proposes an evolution based framework for representing player decision making in games and a simulation based method for evaluating human likeness of game playing agents at three different levels. The framework is applied in two different ways: evolving in a top-down manner from designer-driven intuitions and evolving in a bottom-up, data-driven manner from play-traces. The evaluation method is then used on both applications of the framework to evaluate their performances. Finally, a possibility for combining the two applications of the framework, allowing for hybrid top-down/bottom-up decision modeling through generative agents is suggested.

Generative, game-playing agents that represent and replicate human decision making may be useful in games for many purposes e.g. as believable stand-ins for human players or as benchmark rivals for players to surpass. The work presented here focuses on using game playing agents representing human decision making styles as stand-in players, supporting the traditional process of human play-testing.

Play-testing is typically an integral part of game development (Fullerton et al., 2004). The complexity and cost of the play-testing depends on the kind of game under development, the stage in the games development process, and the objectives of the play-testing. At one extreme play-testing may be conducted by the game designer herself by simply imagining how players might interact with the game, a feature or a piece content. At the other extreme play-testing may be conducted under highly instrumented laboratory conditions or at a massive scale in the wild by telemetrically collecting data from players after the launch of the game (El-Nasr et al., 2013).

In this paper, we suggest there may be an opportunity for methods using generative agents to support designers in situations where new content is being developed, but access to human play-testers is limited or impossible. For example, when a level designer is implementing a new level for a game or making changes to an existing one, these changes might not be large enough to mandate a full play-test with human players. Still, it might be useful for the level designer to observe how different kinds of players would interact with the level.

In situations like these, generative game playing agents based on models of human decision making might provide designers with surrogate play-traces to inform their design process and explore what parts of the game space players are likely to interact with and how, effectively delivering *procedural play-testing*.

When agents sufficiently simulate a particular archetypal human decision making style we call agents *procedural personas*. Integrated with content creation tools, we envision that procedural personas will allow for mixed-initiative game design tools that yield immediate feedback during the design process, even if this feedback is not a completely accurate representation of how human players might play the game. Additionally, play-traces from procedural personas can be used as input for procedural content generation systems shaping the output in response to the generative player models (Liapis et al., 2015).

In other words, agents that play like humans can help understand content by playing it as it is being created.

### 9.2.1 Research Questions

An important question then arises with respect to which sources of information about player decision making styles are useful for constructing believable procedural personas that simulate human decision making with sufficient accuracy. Do we need some amount of low level behavioral data from actual players or can we derive the same information from the expert knowledge of a game designer?

A second question is how general we can make the resulting models. Can we ensure that they perform consistently on unseen content that either no play-traces were sampled from or that the game designer was not explicitly considering?

The work presented here addresses these questions by comparing two particular methods for realizing procedural personas, one drawing on designer expert knowledge and one using empirically gathered play test data, in order to evaluate which method produces the best models for generating synthetic play-test data.

### 9.2.2 Prior Work

In previous work we have designed a simple turn-based, tile-based dungeon crawling game, *MiniDungeons*, which features monsters, treasures and potions in mazes (Holmgård et al., 2014b). 38 players played 10 levels of this game and we recorded their every action. Next, we analyzed the design of the game to extract a number of possible affordances which we translated into partially conflicting objectives that a player might seek to fulfill (e.g. kill all monsters, avoid danger or get to the exit quickly). Using these *affordances* we trained agents to play the game rationally for each objective. Both Q-learning (Holmgård et al., 2014b) and evolutionary algorithms (Holmgård et al., 2014a) were used to train high-performing agents; the evolved agents have the benefit that they

generalize to levels they were not trained on in contrast to Q-learning agents which were unable to perform on levels they previously had not seen.

### 9.2.3 Metrics and Methods for Comparing Agents to Human Players

The agents' behaviors were compared to play-traces of the human players through a metric we call the *action agreement ratio* (AAR) which compares agents and humans at the action level — comparing every action of the player and the agent and asking if the agent would pursue the same *next action* as the player. But is this really the right level of analysis for comparing players to agents? It could be argued that the microscopic level of comparing actions gives a biased view of how well an agent's behavior reproduces player behavior, and that it is more interesting to look at behavior not on the level of atomic decisions, but rather at the level of tactical or strategic decisions. Further, are we right to assume that players exhibit boundedly rational behavior given some set of objectives? It might be that with the same agent representation, we could train agents that reproduce player behavior better by using the actual play-traces as training data instead of focusing on player objectives. The current paper tries to answer these two questions.

Expanding on previous work (Holmgård et al., 2014d), we propose two new play-trace comparison methods, *tactical agreement ratio* (TAR) and *strategic agreement ratio* (SAR) that, instead of asking whether an agent would perform the same singular action as the player in a given state, ask whether it would choose to pursue the same *next affordance* or the same *overall outcome*, respectively.

We also train a second class of agents to behave as similarly as possible to human players on *unseen* levels using play-traces as objectives, again evaluated on the three levels of comparison: the action level, the tactical level, and the strategic level. We call such agents *clones*.

Finally, we train a third class of agents to behave as similarly as possible to human players on *previously seen* levels in order to explore the maximal performance of our chosen representation. We call such agents *specialized agents* as they are likely to be the closest fit of the representation to an individual play-trace, but are trained for just one particular level.

### 9.2.4 Modeling Bounded Rationality

Grounded in contemporary decision science, this paper has two central assumptions about human players' decision making: The first is that players' decisions are guided

by their expected *utility* for a given decision; i.e. the amount of experienced value they expect to derive from the consequences of a decision.

The second is that human players exhibit *bounded rationality* i.e. players allocate limited amounts of cognitive resources to decisions in games either due to innate limitations or because they only apply part of their cognitive capacity to the decision due to conscious or subliminal reasons. Decision making style in games thus depends not only on preferences in outcomes, but also the resources the player is willing or able to allocate to the decision making task. Our approach to simulating a decision maker in the form of a generative agent is to represent these two characteristics, the player's rational utility function and the player's cognitive bounds, in the implementation of the agent.

In the following we outline the relations between persona theory, decision theory, player modeling, and the resulting concept of procedural personas. We briefly describe our testbed game, MiniDungeons, and the methods we used to create game playing personas and clones, before we present the results from comparing the resulting agents to the human players.

## 9.3 Related Work

In this section, we review decision theory, the concept of personas as applied to (digital) games, player modeling, and the relations between the three areas in this study.

### 9.3.1 Decision Theory and Utility

The personas used for expressing designer notions of archetypal player behavior in MiniDungeons are structured around the central concepts of decision theory. Decision theory states that whenever a human makes a *rational decision* in a given situation, the decision is a result of an attempt to optimize the expected *utility* (Kahneman and Tversky, 1979). Utility describes any positive outcome for the decision maker and is fundamentally assumed to be idiosyncratic. This means that in principle no definite assumptions can be made about what can provide utility to the decision maker. The problem is further complicated by the fact that the effort a decision maker directs toward attaining maximum utility from a decision can be contingent on the expected utility itself. For problems that are expected to provide low utility even in the best case, humans are prone to rely more heavily on heuristics and biases for the decision making process, further bounding the rational analysis applied to the problem (Simon, 1955; Rubinstein, 1998; Kahneman, 2003; Gigerenzer and Gaissmaier, 2011).



In practice, however, for structured, well-defined problems, such as many games, insights from e.g. psychology or contextual information about the decision maker or the decision problem may provide us with opportunities for assuming which decisions are important and which outcomes may be of utility to the decision maker. As decision spaces, most games are special cases since the available decisions and their consequences are highly structured by the game’s mechanics and evaluation mechanisms. Games, through their design, often provide specific affordances (Gibson, 1977; Elias et al., 2012) to the player, and suggest utility for various outcomes. This perspective forms the basis for our understanding of player behavior in our testbed game, as we assume that players are interacting with the game in accordance with the rules, understanding and responding to the affordances of our game. That, in turn, motivates our use of utility for attaining game rule based affordances as the defining characteristics of the personas we develop. Similar theoretical perspectives have been described by other authors, notably Mark (2009). When attempting to characterize player decision making styles in games using utilities, it is important to consider the level of decision making relevant for the game, as described in Canossa and Cheong (2011). Here, we model players at the individual *action level*, at the more *tactical level* of game affordances, and at the *strategic level* of aggregate outcomes.

In the following section, we suggest how the concept of play-personas can be used to arrive at a selection of utility configurations for a particular game.

### 9.3.2 Personas

The concept of personas was first adapted to the domain of (digital) games under the headline of *play-personas* by Canossa and Drachen (2009) who define play-personas as “clusters of preferential interaction (*what*) and navigation (*where*) attitudes, temporally expressed (*when*), that coalesce around different kinds of inscribed affordances in the artefacts provided by game designers”. Their work focuses on how assumptions about such player preferences can be used as metaphors for imagined player behavior during the design process or patterns in observed player behavior can be used to form lenses on the game’s design during play-testing.

Applying the perspective of decision theory further narrows the play-persona concept. Rather than considering any arbitrary reasons for player preferences valid, decision theory provides a perspective to operationalize the backgrounds for preferences into combinations of affordances and utilities. For any spatio-temporal configuration of a given game, a limited number of plausible affordances can be determined using information

about the game mechanics and reward structures. Based on these affordances, different hypothetical combinations of utilities can be used to create metaphors for typical player behavior. To the extent that these metaphors match what actual human players decided, they can be considered lenses on the players' decision making styles with utilities explaining how player preferences are distributed between the available affordances. Our long term research agenda is to operationalize the play-persona concept into actual game playing procedural personas, by building generative models of player behavior from designer metaphors, actual play data, or combinations of the two.

In the following section, we argue for the use of game playing agents to provide such representations of possible configurations of utilities, drawn from play-personas.

### 9.3.3 Player Modeling

Generative models of player behavior can be learned using a number of different methods. A key dichotomy in any player modeling approach lies in the influence of theory (vs. data) for the construction of the player model (Yannakakis et al., 2013). On one end, *model-based* approaches rely on a theoretical framework (in our case persona theory or expert domain knowledge) and on the other hand, computational models are built in a *model-free*, data-driven fashion. In this paper, *personas* represent the model-based approach while what we term *clones* represent the data-driven approach. Within the model-free approach, a fundamental distinction is between *direct* and *indirect* player imitation, where the former uses supervised learning methods to train agents directly on play-traces, and the latter uses some form of reinforcement learning to train agents to behave in a way that agrees with high-level features extracted from the play-traces (Togelius et al., 2007). In several investigations, direct and indirect comparisons have been compared for imitating player behavior in racing games (Togelius et al., 2007; Van Hoorn et al., 2009) and platform games (Ortega et al., 2013). Model-free player modeling can be done by imitating the player directly, using supervised learning methods on the play-traces, or indirectly using some form of reinforcement learning to train agents to behave in a way that agrees with high-level features extracted from the play-traces (Togelius et al., 2007). Evolutionary computation can be used to optimize an agent to behave similarly to a play-trace or optimize it to exhibit the same macro-properties as said play-trace (Togelius et al., 2007; Van Hoorn et al., 2009; Ortega et al., 2013). Direct imitation is prone to a form of over-fitting where the agent only learns to cope with situations which exist in the play-traces, and might behave erratically when faced with new situations. Indirect imitation to a large extent solves this problem by learning a more robust, general strategy, which could be termed a decision making style. Here, we investigate this problem by comparing clones that are directly trained on play-traces,

but tested on unseen maps, to the personas, while we use specialized agents directly trained and tested on seen maps as a best case, but context dependent, performance of the chosen agent representation.

In the following section we describe the MiniDungeons testbed game in further detail.

## 9.4 MiniDungeons

The testbed game, *MiniDungeons*, implements the fundamental mechanics of a dungeon exploration game where the player navigates an avatar through a dungeon containing enemies, power-ups, and rewards. The turn-based game puts the player in a top-down viewed tile-based 12 by 12 dungeon containing monsters, potions, and treasures. Impassable tiles constitute the walls of the dungeon, while passable tiles contain enemies or items for the player. All of the level is visible to the player who can move freely between passable tiles. When the player moves to a tile occupied by a monster or item, immediately the monster is fought or the item is collected and applied. The player has a 40 *hit point* (HP) health counter and dies if this drops to zero. Monsters randomly deal between 5 and 14 HP of damage while potions heal 10 HP up to the maximum value of 40 HP. Treasures have no game mechanical effect other than adding to a counter of collected treasure. The game contains one tutorial level and 10 “real” levels. For further details on the test-bed game and discussion of its properties, we refer to Holmgård et al. (2014b). The necessary data for developing and evaluating the agents was collected from 38 anonymous users who played MiniDungeons on-line; this resulted in 380 individual play-traces on the 10 MiniDungeons levels provided. The data was subsequently used to evolve clones and specialized agents as described below. Figure 7.3 shows Level 2 from the game, along with human play-traces from the level, exemplifying the diversity of human decision making styles expressed in even a simple game like this. In Section 9.6.1 we give a brief introduction to our method of representation, but first we introduce three metrics that we propose for evaluating to which degree personas, clones, and specialized agents successfully enact human decision making styles in MiniDungeons.

## 9.5 Agreement Ratios for Evaluating Player Models

In this section, we present the three different metrics used to evaluate the performance of the agents. Each metric was constructed to capture a different level of game play, ranging from the specific and atomic to the general and aggregated.

FIGURE 9.1: Heat-maps of six selected human play-traces in Level 2 of MiniDungeons, showing a diversity of player decision making styles. Note that in two heat-maps, top center and bottom right, the player died before completing the level.



### 9.5.1 Action Agreement Ratio

The first metric used to evaluate agent to human likeness is the *action agreement ratio* (AAR). AAR considers each step of a human play-trace a distinct decision. To produce the AAR between an agent and a human player, all distinct game states of the human play-traces are reconstructed. For each game state, the agent being tested is inserted into the game state and queried for the next preferred action, essentially asking: “What would you do?”. If the action is the same as the actual next human action, the agent is awarded one point. Finally, the AAR is computed by dividing the points with the number of decisions in the human play-trace. As such, a perfect AAR score of 1.0 represents an agent that for every situation in the player’s play trace decided to take exactly the same action as the player did.

### 9.5.2 Tactical Agreement Ratio

The second metric used for evaluating the likeness between agents and humans is the *tactical agreement ratio* (TAR). TAR only considers reaching each distinct affordance in the level a significant decision, ignoring the individual actions in between. For MiniDungeons the affordances considered relevant are: fighting a monster, drinking a potion,

collecting a treasure, or exiting a level. For each affordance reached in the human play-trace, the resulting game state is reconstructed and the agent being tested is inserted into the game state. The agent is then allowed as many actions as necessary to reach the next affordance, asking the question “What affordance would you go for next?” at the tactical level. If the next encountered affordance, in terms of both type and location, matches the actual next human one exactly, the agent is awarded a point. Finally, the TAR is computed by dividing the points with the number of affordances reached in the human play-trace. As such, a perfect TAR score of 1.0 represents an agent that visits every affordance in the level in the same order as the player originally did.

### 9.5.3 Strategic Agreement Ratio

The third metric used for evaluating the likeness between agents and humans is the *strategic agreement ratio* (SAR). Operating at the general and aggregate level, SAR considers the total amount of affordances engaged with for each level. The affordances considered by SAR are: the number of monsters fought, the number of treasures collected, the number of actions taken, and whether the agent reached the exit and was alive at the end of the level. For each affordance the absolute difference between the agent’s measure and the player’s measure (e.g. monsters killed) is calculated and normalized by the maximal possible number for the level or, in the case of the number of moves, in relation to the number of moves in the player’s play trace. These statistics are then summed, divided by the number of statistics (in this case five). The score, which is an expression of how different the agent’s statistics are from the player’s, is subtracted from 1.0 to produce the SAR. As such, a perfect SAR score of 1.0 would indicate an agent that fought exactly the same number of monsters, collected exactly the same number of treasures, died in combat or exited the level just like the player, and did so in exactly the same number of actions. In other words, the SAR asks the question “How often would you go for each affordance in this level?”

In the following section we describe the testbed game, MiniDungeons, in which these comparison metrics were used, and the controllers that were evolved and evaluated using the metrics as fitness functions.

## 9.6 Generative Agents

This section describes the general controller framework used to evolve both personas, clones, and specialized agents, and the evolutionary algorithm used to learn behaviors from utilities in the case of personas and from the AAR, TAR, and SAR metrics in the

cases of clones and specialized agents. All agents were evolved using the same algorithm; what differentiates personas, clones, and specialized agents is the fitness function and whether they are evolved across multiple levels (personas and clones) or on just one level (specialized agents).

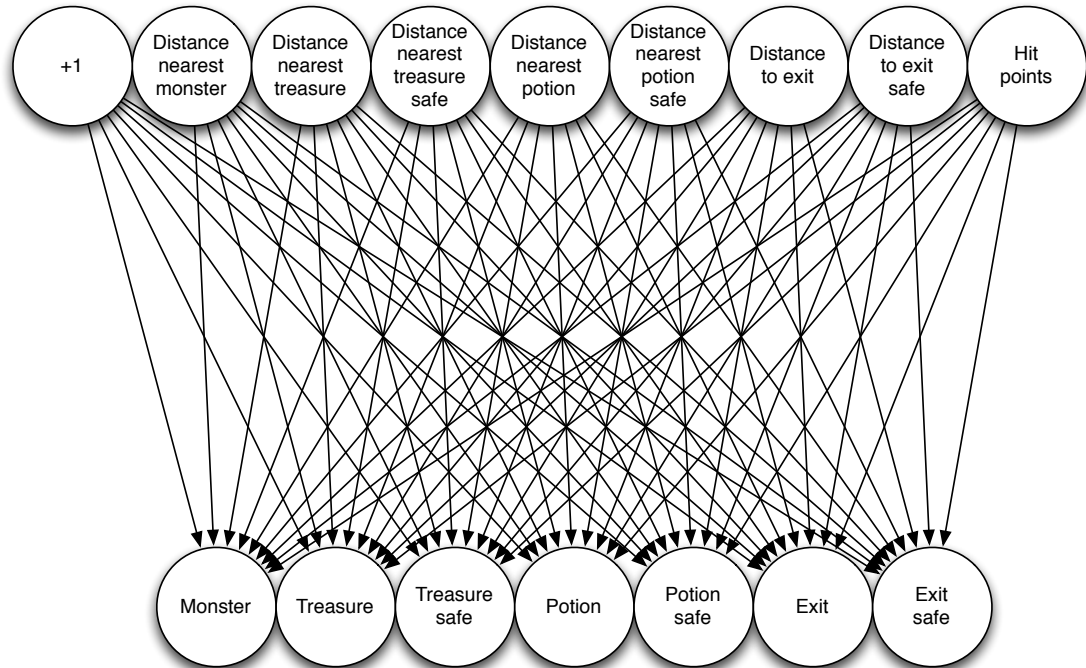
### 9.6.1 Evolving Agent Controllers

The controllers of the agents are represented as seven linear perceptrons. Each perceptron takes 8 inputs describing safe and risky path distances to the nearest affordances in the map as illustrated in Figure 9.2. By only considering the nearest affordances, the agent controller simulates bounded rationality in the sense that it only looks one affordance ahead. For the sake of simplicity this bound is not varied in these experiments, but potentially this horizon could be changed to represent different degrees of bounded rationality. Further details of the controller representation is given in Holmgård et al. (2014a). Controllers are evolved using a  $(\mu + \alpha)$  evolutionary strategy without self-adaptation. For each generation the top 2% performing elite individuals remain unchanged, the lowest performing half of the remaining population is removed, and single-parent offspring from the remaining individuals are produced to maintain the population size. Finally all individuals not in the elite are mutated. Mutation is accomplished by changing each connection weight in the network with a random number drawn from a Gaussian distribution centered around zero with a standard variation of 0.3, a value confirmed as useful for this game by informal experimentation. All experiments are done using a population size of 100 individuals, evolved for 100 generations. Controllers are initialized with random connection weights for all connections in the linear perceptrons. The topology shared by all controllers is illustrated in Figure 9.2.

### 9.6.2 Personas

For the purpose of the experiments five individual personas with different utility configurations were defined, based on designer interpretations of likely game-play in MiniDungeons. The personas were intended to represent five hypothetical extreme decision making styles in interacting with the game: an *Exit* (E) persona who simply tries to escape the level, a *Runner* (R) persona who tries to escape the level in as few steps as possible, a *Survivalist* (S) persona who tries to avoid risk, a *Monster Killer* (MK) persona who tries to kill all monsters and escape the level, and a *Treasure Collector* (TC) persona who attempts to collect all treasures and escape the level. The decision making styles are defined by the utility weights presented in Table 8.1, and serve as a metaphor for the relative importance of the affordances to the archetypal player represented by the

FIGURE 9.2: The controller network used for all agents: personas, clones, and specialized agents. The weights of the connections are determined through evolution as described in Section 9.6.1. For every action it takes in the game, a controller uses the current state of the game, represented by the 8 input nodes (disregarding the bias node), and the weights to select which of the 7 strategies represented by the output nodes to pursue.



persona. The values were assigned by the authors, as the designers of the game, by imagining how different archetypal player types would weigh the various affordances. When assigned to personas as a fitness score during the evolutionary process, utility points attained from a level are normalized by the maximally attainable utility for the same level. E.g. killing three monsters in a level with eight monsters will yield a Monster Killer persona 0.375 utility points, while yielding a Treasure Collector persona no points. During evolution, personas are exposed to and evaluated on 9 of the 10 MiniDungeons levels. The 10th level is subsequently reserved for comparison with human players. In total, 50 personas were evolved for this study.

TABLE 9.1: Utility weights for the five designed personas.

Affordances	E	R	S	MK	TC
Move	-0.01	-0.02	-0.01	-0.01	-0.01
Monster				1	
Treasure					1
Death			-1		
Exit	0.5	0.5	0.5	0.5	0.5

### 9.6.3 Clones

Clones, like personas, are evolved by exposing them to 9 of the 10 levels of MiniDungeons. Their fitness value is computed as the average normalized AAR, TAR or SAR across all 9 seen levels. One clone per player per map is evolved, yielding 380 agents per evaluation metric, in total 1140 clones. All subsequent tests are done comparing the clones to the players they were cloned from on their individually unseen levels.

### 9.6.4 Specialized Agents

In order to find the likely closest possible fit of the perceptron-based representation a set of specialized agents is evolved. Again, one agent for each human play-trace for each evaluation metric is evolved, resulting in 1140 total. These are evolved on a single level of MiniDungeons each. Their fitness scores are computed directly from AAR, TAR or SAR on that same level in an attempt to establish the closest fit to each human player the representation can achieve.

## 9.7 Results

This section compares the three presented evaluation metrics, and compares the ability of personas, clones, and specialized agents to represent human decision making styles in MiniDungeons. It presents a breakdown of how the application of different evaluation metrics change which personas are mapped to individual players. Finally, it provides an example of a single player's play-trace and the personas, clones, and specialized agents derived from that player on a particular level.

Table 9.2 shows the mean of the agreement ratios for each kind of agent evolved, using the AAR, TAR, and SAR metrics. In the case of personas, the best matching persona from the 5 persona gallery, meaning the one with the highest agreement ratio (for AAR, TAR, or SAR respectively) is identified for each player on each level and used in the analysis. The distributions of best matches are presented below in Table 9.3.

The achieved ratios are generally highest for AAR, followed by SAR, with TAR producing the lowest ratios. However, while the ratios share some semantic properties such as an upper perfect match represented by 1.0, it should be noted that the ratios are not directly comparable.

No agents, not even the specialized ones, attain a perfect agreement ratio. This indicates that the chosen agent control architecture seems incapable of matching players perfectly.



TABLE 9.2: Means and standard deviations (SD) of agreement ratios attained for best matching personas, clones, and agents evolved using the described fitness functions. It is worth noting that across all three metrics, personas generally exhibit performance close to that of the clones. Additionally, it is worth noting that while specialized agents evolved using TAR and SAR as fitness functions perform best in the metric they were evolved from, this is not the case for agents evolved using the AAR metric as a fitness function. Using the AAR metric, the TAR-evolved specialized agents exhibit the best performance.

Agent type	Fitness function	Evaluation metric					
		AAR		TAR		SAR	
		Mean	SD	Mean	SD	Mean	SD
Personas	Utilities	0.75	0.08	0.62	0.13	0.77	0.16
Clones	AAR	0.77	0.08	0.66	0.13	0.68	0.23
Clones	TAR	0.76	0.09	0.65	0.13	0.67	0.23
Clones	SAR	0.69	0.10	0.53	0.17	0.72	0.22
Specialized	AAR	0.81	0.09	0.73	0.17	0.68	0.24
Specialized	TAR	0.84	0.07	0.86	0.09	0.71	0.24
Specialized	SAR	0.73	0.10	0.57	0.18	0.85	0.21

This may be due to an inability of the seven-perceptron network to represent and hence learn the decision making preferences of actual players or it may be due to the fact that controllers are only provided with information about the distance to the nearest kind of each affordance. While the linear perceptron network has the advantage of having easily inspectable and interpretable weights other networks with greater representational power, such as non-linear multilayer perceptrons, may provide a better matching of individual player from the same information.

An interesting perspective is to compare across agent types within the different evaluation metrics. If we look at how fitness functions based on AAR, TAR, and SAR, respectively perform when cross-evaluated against each other within clones and specialized agents, we see that AAR and TAR produce comparable results. In other words, a clone or a specialized agent evolved using AAR as a fitness function also performs well when evaluated by TAR and vice versa. Agents evolved using SAR on the other hand, only perform relatively well when evaluated by SAR. For the other metrics, clones and specialized agents evolved from SAR perform worse than the other members of their groups. This may indicate that evolving agents using the high-level SAR as a metric, results in some loss of information about the decision making styles that players are enacting. When agents are only evolved from information about the aggregate outcome for each level, they might not learn about the order in which a player prefers to pursue affordances. SAR results also show higher standard deviations, which may be attributable to the loss of information from human play-traces making it harder to produce reliable matches. This suggests that AAR and TAR may be the most relevant metrics for modeling player decision making styles in MiniDungeons.

Next, we investigate the differences between how well personas and clones agree with player decision making styles. In the following analyses, specialized agents are omitted as they are assumed to represent the likely closest fit of the representation and indeed produce the highest agreement ratios when evaluated on the same metric from which they were developed.

Using the AAR evaluation metric across personas and clones, a one-way ANOVA shows significant differences ( $F(3, 1516) = 62.24, p < 0.001$ ) between the means of the four groups: the best matching personas and the three kinds of clones. Similar differences were established when using the TAR ( $F(3, 1516) = 68.16, p < 0.001$ ) and the SAR evaluation metrics ( $F(3, 1516) = 16.29, p < 0.001$ ), in spite of large standard deviations in the latter case.

For all evaluation metrics, Tukey HSD tests for post-hoc analysis reveal that the personas are significantly different ( $p < 0.05$ ) from all clones, regardless of the fitness function used to evolve the clones. The only exception is when personas are compared to clones evolved from TAR, evaluated by AAR. With AARs of 0.75 and 0.76, respectively, there is no significant difference between these two groups.

When examining the specific mean agreement ratios of the personas in contrast to the clones it is clear that the actual differences in the case of AAR, even though significant, are minor and that the personas for all practical purposes achieve the same performance as the clones, perhaps with the exception of clones evolved through SAR which perform the worst out of the group.

A similar pattern is repeated when examining the mean agreement ratios calculated using TAR, though the personas in this case rank lower than clones evolved from both AAR and TAR, while clones evolved using SAR again produce the lowest agreement ratio.

Finally, when applying the SAR metric, the personas outperform all clone types with a sizable performance difference between the personas and the worst performing clones, the ones evolved using SAR.

Taken together the above results indicate that a small gallery of five personas, defined using utility theory by game designers, are capable of representing a corpus of 38 players across 10 levels with roughly the same performance individually evolved clones. The personas do perform worse than specialized agents evolved to specifically copy player behavior on one particular level, but this is to be expected. Measured by the action and strategic metrics, the personas come relatively close to the specialized agents, while they lag further behind the specialized agents when applying the tactical metric.

Table 9.3 shows which personas exhibited the best ability to represent human play-traces, for each MiniDungeons level and in total. For each human play-trace, the personas with the highest AAR, TAR, and SAR respectively, are identified. All three metrics generally favor the Treasure Collector persona as the best match for most play-traces, although there is some discrepancy between the three measures in terms of which personas represent the human play-traces best. Notably, the SAR metric yields best matches that are quite different from the matches yielded by the AAR and TAR metrics. This underlines the fact that using an aggregate strategic level metric allows for larger degrees of variability in the decision order, since different orderings may lead to the same aggregate results.

TABLE 9.3: Best persona matches based on Action Agreement Ratio (AAR), Tactical Agreement Ratio (TAR), and Strategic Agreement Ratio (SAR), respectively

<b>AAR</b>											
Persona/Level	1	2	3	4	5	6	7	8	9	10	Total
Exit	0	2	5	1	0	1	5	1	2	3	20
Runner	0	0	0	0	0	0	0	0	0	0	0
Survivalist	0	1	0	0	0	0	0	0	0	0	1
Monster Killer	8	8	0	2	3	1	7	2	2	0	33
Treasure Collector	30	27	33	35	35	36	26	35	34	35	326
Total	38	38	38	38	38	38	38	38	38	38	380
<b>TAR</b>											
Persona/Level	1	2	3	4	5	6	7	8	9	10	Total
Exit	0	0	5	1	0	1	4	0	2	0	13
Runner	0	0	0	0	0	0	0	0	0	0	0
Survivalist	0	0	0	0	0	0	0	0	0	0	0
Monster Killer	5	15	3	3	4	3	4	5	0	0	42
Treasure Collector	33	23	30	34	34	34	30	33	36	38	325
Total	38	38	38	38	38	38	38	38	38	38	380
<b>SAR</b>											
Persona/Level	1	2	3	4	5	6	7	8	9	10	Total
Exit	7	7	10	11	2	15	11	1	7	0	71
Runner	0	0	0	1	0	2	2	0	0	0	5
Survivalist	0	0	0	0	0	1	0	3	0	1	5
Monster Killer	5	10	5	7	3	7	18	6	14	3	78
Treasure Collector	26	21	23	19	33	13	7	28	17	34	221
Total	38	38	38	38	38	38	38	38	38	38	380

Finally, Figure 9.3 shows a particular player’s original play trace along with the best matched personas, using each metric, and the derived clones and specialized agents. The player was matched by the Treasure Collector by all three metrics, as was indeed the typical case in the data set. The heat-maps illustrate how most agents provide a relatively close match to the visited areas of the level. The Treasure Collector comes close to matching the visitation pattern of the player, but fights several monsters the player did not fight, as monsters have no bearing on the utility attained by the Treasure Collector as long as they do not endanger the persona’s chances of completing the level.

The AAR metric does not consider the order of the affordances in the level directly, but only the immediate next atomic action. As a consequence, the clone and the specialized agent derived from this metric miss visiting tiles occupied by monsters and potions visited by the player, but in general exhibit a visitation pattern close to that of the player.

The TAR metric does consider the kinds of affordances it visits and the order of them and hence visits more of the same specific monsters, treasures and potions of the level as the player. As a consequence the visitation pattern also looks the same.

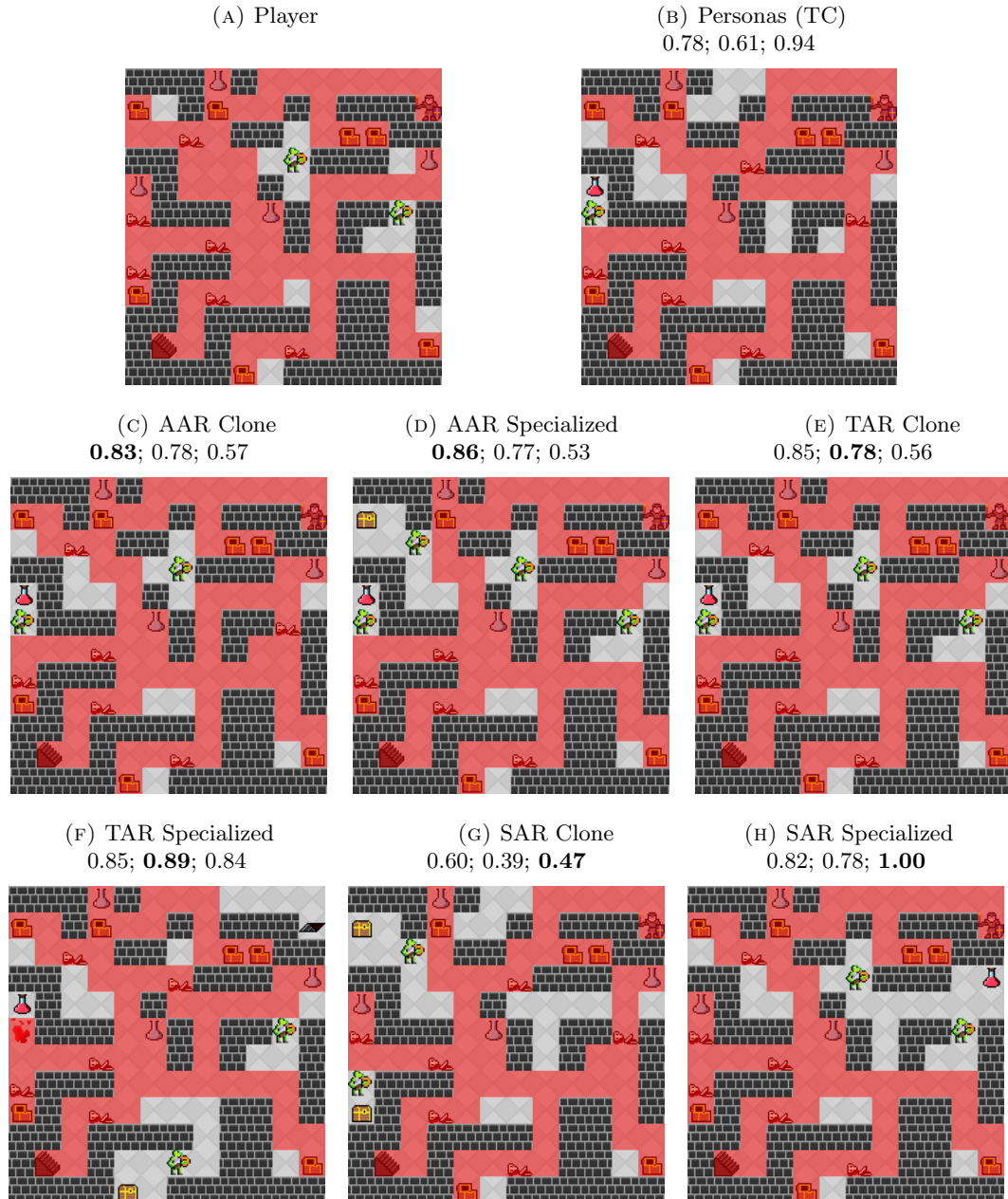
The SAR metric only considers outputting the same number of affordance interactions as the player and does not concern itself with neither the order, nor the location, of these affordances. The SAR based clone ends up with a relatively (and uncharacteristically) poor performance, possibly by choosing an early order of actions that makes it difficult for it to achieve the same statistics as the player. Meanwhile, the SAR specialized agent attains perfect performance, but does so through a quite different visitation pattern, demonstrating how the SAR metric allows for greater variation in the underlying implementation of actions leading to the same SAR value.

## 9.8 Discussion

In the following discussion we revisit and evaluate the results from the experiments. Two important observations must be made in relation to the experiment used to collect the play-traces: Firstly, all 38 players in the study were playing MiniDungeons for the first time. Though they were given unlimited plays on the tutorial level they were not familiar with the rules of the games or the distribution used to choose damage dealt by the monsters in the level. The play-traces collected are most likely subject to learning effects, as the players moved from being novices to developing some expertise, which in turn may have impacted their decision making styles. It might also have made players change decision making styles along the way, as their expertise increased. However, MiniDungeons is a relatively easy game to learn, making it likely that any learning effects beyond the first few levels would be limited. Future work could attempt to measure such learning effects or try to counter-act them by randomizing level order or operating with a block experimental design.

Secondly, each level in the game was played independently of the preceding levels. Each level let the player start with full hit-points. This may have lead players to favor the Treasure Collector decision making style, simply because risk was perceived as low. If

FIGURE 9.3: Heat-maps from one player and all best matching personas, all clones, and all specialized agents on Level 2. For each agent the maximally attained agreement on each metric is indicated in the order AAR; TAR; SAR. The metric used to drive the evolution of each agent, excepting the persona, is indicated in bold. The best matching persona, by all metrics, was the Treasure Collector.



a player died before reaching the exit, the player would start the next level with full health.

Below, we discuss the observed properties and performances of the three kinds of metrics used. We then briefly discuss the used controller architecture and whether its performance, as indicated by the metrics, suggests that it is sufficient for playing MiniDungeons. Afterwards, we discuss the implications of personas and clones attaining relatively

similar performances and whether either method hypothetically would be transferable to other games. Finally, we suggest future work, including adapting both personas and clones further to observed play-traces by fitting utility values through multi-objective evolution.

### 9.8.1 AAR, TAR, and SAR metrics

The three different metrics presented in this paper are not directly comparable as they operate on three different levels of analysis, relative to the mechanics and affordances of the MiniDungeons game. As the results above indicate, each metric yields somewhat different behavior when applied to the evolution of game playing agents representative of human players, each capturing different aspects of the original player's decision making style.

The metrics operate at three different levels of analysis, which is rooted in the theoretical notion that, even for simple games, decision making takes place at multiple levels simultaneously, focusing on the individual decision, the order of individual decisions, and the aggregate outcome of a set of decisions irrespective of the individual decisions or their order.

However, the specific implementation of the three metrics for MiniDungeons, meaning the identification of relevant affordances and related decisions, was pragmatically shaped by the game's design and not rooted in a theoretical system of analysis. This could be potentially be addressed by applying existing formal systems for identifying affordances, decisions, and actions in games, such as e.g. through the notion of game design patterns (Björk and Holopainen, 2004). The cost of taking this approach would be a longer, more involved analytical process before metrics could be defined for a particular game, but the gain would potentially be a greater validity of the chosen affordances. Depending on the purpose of the modeling, this may or may not be a desirable trade-off: An single developer looking to reduce time and resource consumption spent on play-testing may be perfectly content with using her own expertise as a valid foundation. In contrast, a team working in a larger organization or doing an academic study of one or multiple games might prefer grounding the selected affordances and decisions in a larger corpus of examples collected from other games.

Finally, while these metrics are specifically targeted at modeling decision making, other play-trace comparison metrics could be used to compare agent behavior to human player behavior: e.g. action/edit-distance based methods such as the *Gamalyzer* metric (Osborn and Mateas, 2014).

### 9.8.2 Controller Architecture Performance

The results indicate that a relatively small gallery of personas, crafted analytically by a game designer, may provide representational performance comparable to that achieved by the cloning approach. Still, the evolution of specialized agents that represent individual players on individual levels provides a reproductive fidelity that neither personas nor clones can match. On the other hand, specialized agents suffer from the problem that they are optimized for one particular level and are not well suited for generalizing to new levels, for which they have not been evolved.

The performances of the specialized agents reveal that the chosen representation using seven linear perceptrons is incapable of learning a human play-trace perfectly. This could motivate two changes to the controller to achieve better performance. Firstly, the use of a more complex controller, such as e.g. a controller based on multilayer perceptrons configured using Neuro-Evolution with Augmenting Topologies (Stanley and Miikkulainen, 2002), would most likely exhibit a greater ability to learn individual play-traces. Secondly, controller performance might be improved by increasing the environment sensing capabilities of the controller. In the current implementation, the controller is only informed about risky and safe path distances to the nearest affordance of each kind. As such, it can be interpreted as representing the assumption that players only look to the nearest affordances that they can reach when making decisions. This is most likely an over-simplification which, while having the advantage of simplifying the implementation and intelligibility of the model, may be too aggressive. Future work should explore more complicated agent control architectures.

### 9.8.3 Personas and Clones for MiniDungeons and Beyond

The fact that the top-down approach of persona construction and the bottom-up approach of cloning yield similar performances provides some indication that using procedural personas for synthesizing player models and play-test data for games might be feasible at least for games of the same scope as MiniDungeons.

We suggest that this might be useful in multiple situations: One example could be when game designers need quick access to potential interactions with new content during an iterative design process. By considering only the basic affordances of their game, and producing different preference orderings of these by assigning utilities, they may quickly sketch out different play-styles and see them in action. Another example could be for games using procedural content generation to an extent where full human curation of the generated content is impossible. Here, procedural personas can act as proxy critics of a human designer's intentions (Liapis et al., 2015).

The persona method is less play-trace-dependent and computationally expensive than the cloning method, but needs an expert game designer. Still, some players may exhibit decision making styles that cannot be captured by the designer's intuition, and would be captured better by the cloning approach.

MiniDungeons is arguably a game with a small scope, compared to most commercial games, even most made by single, independent developers. This begs the question of whether the method is extensible to larger games. While this is an open question at the moment, some strategies may be imagined. Any game which may be abstracted to a discrete graph of decisions may in principle be subject to the method. How this abstraction could be appropriately implemented would again be dependent on the design of the game in question and possibly subordinate modeling steps could be added to the method. If e.g. a first person shooter were being modeled, a first step could be to reduce the game's spatial areas into specific decision points, based on the affordances they contain. A collection of design patterns for the genre, such as those described in Hullett and Whitehead (2010), might aid in this process. Once the design patterns were identified, personas would be implemented to choose between these areas, based on their play style preferences, choosing e.g. between a sniper area or a close combat area. Decision making styles within each area could then be simulated using a separate application of the persona method, a probabilistic model, or simple scripted behavior. The major challenge in this case would be to arrive at a successful abstraction of the game's state space into appropriate affordances with an acceptable amount of effort. Future work will focus on formalizing this process.

#### **9.8.4 Combining Personas and Clones**

The fact that the persona and the cloning methods seem to perform equally well for MiniDungeons raises the question of whether the persona method has any value once a sufficient amount of human play-traces have been collected for modeling. We suggest that even in this situation, procedural personas may offer two advantages to game and level designers.

Firstly, each persona defines a decision making style grounded in the designer's expectations. Individual or groups of human players may be described by their distances to these a priori defined decision making styles. This means that the designer is not interpreting the play traces in terms of their observed decision making styles alone, but also in terms of how different they are from the designer's conception of styles catered to in the game's decision making space. This helps contrast what the designer expected to what the players actually did.



Secondly, once a relation between personas and clones has been established, it becomes possible to interpolate between personas and individual clones or points defined by clusters of clones. This allows a designer to define new personas that are more like the observed play-traces, but still informed by the designer's own observations, expectations and wishes for decision making styles within the game space.

Distances between personas and clones can be defined according to any of the three proposed metrics, depending on the designer's agenda, or according to an aggregation of all three metrics. While the designer could define these new adapted personas manually, a more efficient approach would be to fit utility values automatically to maximize one, two or all three of the defined agreement ratios. If a designer wanted to maximize agreement for all three metrics e.g. multi-objective evolutionary methods (Van Hoorn et al., 2009) could be used to arrive at suitable utility weight configurations. Once these had been determined computationally, the designer could then inspect these utility values to understand the differences between the original personas and the adapted ones and manually adjust them to her preference.

## 9.9 Conclusion

This paper presented a framework for modeling player decision making styles. This framework was implemented in three different manners: One was based on *personas*, evolved from designer expert knowledge, another was based on *clones*, based on human play-traces, while the third used *specialized agents* to replicate human play-traces while sacrificing generalizability. Three metrics were used to evaluate the agents' ability to represent human decision making styles, focused at the *action*, *tactical*, and *strategic* levels, respectively.

Personas and clones were shown to represent human decision making styles almost equally well when compared at the action and tactical levels. At the strategic level, personas were somewhat better at representing human decision making styles, compared to clones. The three different metrics showed how focusing on different analytical levels of a game can be used to characterize play traces differently and to generate variation in game playing agents.

Two advantages of using personas over clones is that we lose little accuracy from using personas instead of clones and we do not need to collect empirical data from players before we can start modeling. Clones, on the other hand, may learn examples of decision making styles that we as designers might not imagine on our own. The advantage of using either personas or clones over specialized agents is that even though they are less

accurate they may be used on novel content, which is critical in enabling procedural play-testing.

Based on the results, we conclude that using the top-down approach procedural personas for representing player decision making styles is comparable to using the bottom-up approach of cloning. Either approach may be useful for synthesizing play-test data for new content as it is being generated which may be of use to game designers authoring content as well as game designers building procedural content generation systems.

## 9.10 Acknowledgments

We would thank the players of the game for providing data and our reviewers for providing a number of observations and suggestions that significantly improved this paper. The research was supported, in part, by the FP7 ICT project C2Learn (project no: 318480) and by the FP7 Marie Curie CIG project AutoGameDesign (project no: 630665).

## Chapter 10

# Procedural Personas as Critics for Dungeon Generation

### Reference:

Liapis, Antonios, Christoffer Holmgård, Georgios N. Yannakakis, and Julian Togelius (2015). “Procedural Personas as Critics for Dungeon Generation”. In: *Applications of Evolutionary Computation*. Springer, pp. 331–343.

### 10.1 Abstract

This paper introduces a constrained optimization method which uses procedural personas to evaluate the playability and quality of evolved dungeon levels. Procedural personas represent archetypical player behaviors, and their controllers have been evolved to maximize a specific utility which drives their decisions. A “baseline” persona evaluates whether a level is playable by testing if it can survive in a worst-case scenario of the playthrough. On the other hand, a Monster Killer persona or a Treasure Collector persona evaluates playable levels based on how many monsters it can kill or how many treasures it can collect, respectively. Results show that the implemented two-population genetic algorithm discovers playable levels quickly and reliably, while the different personas affect the layout, difficulty level and tactical depth of the generated dungeons.

### 10.2 Introduction

The generation of dungeons is one of the first instances of procedural content generation (PCG) with *Rogue* (Toy et al., 1980). Since then, many games have used algorithms

to generate dungeons, e.g. *Diablo* (Blizzard North, 1996), *Daggerfall* (Bethesda Softworks, 1996) and *Daylight* (Zombie Studios, 2014). Generating dungeons has also been a fertile research topic as summarized by Linden et al. (2013); algorithmic approaches using constraints (Roden and Parberry, 2004), grammars (Dormans, 2010) and genetic algorithms (Hartsook et al., 2011) have been successfully applied to this task.

This paper introduces a method where procedural personas act as critics in a search-based procedural content generation (SBPCG) framework (Togelius et al., 2011). Procedural personas are artificial agents which represent archetypical player behaviors (e.g. rushing to the goal, killing monsters, collecting treasures). In this paper, the personas have been evolved on a set of authored dungeons, according to different fitnesses that match archetypical decisions-making priorities. The testbed game, named MiniDungeons, is a simple turn-based roguelike game; the game has been tested by human users and a close match between procedural persona playstyle and human playstyle was found (Holmgård et al., 2014a).

Using procedural personas to test the evolving dungeons situates the proposed method as a type of simulation-based SBPCG. However, the persona-critics are used not only to evaluate how appropriate a dungeon is for a particular playstyle, but also whether the dungeon is actually playable. The requirement that a dungeon can be completed by a simple “baseline” persona — despite any stochasticity of the gameplay — adds another constraint to the generative process. This paper uses a two-population genetic algorithm for the purposes of constrained optimization, which evolves both feasible and infeasible dungeons (Kimbrough et al., 2008). Dungeons are tested by a “baseline” persona based on whether it can complete a worst-case scenario of the dungeon; this persona also evaluates infeasible dungeons’ distance from feasibility. Playable levels are evaluated by a Monster Killer persona or a Treasure Collector persona based on how many monsters it can kill or how many treasures it can collect, respectively.

## 10.3 Previous Work

This Section covers the core background material (testbed game, procedural personas and evolutionary level design) on which the presented method is built.

### 10.3.1 MiniDungeons game

MiniDungeons is a simple turn-based roguelike puzzle game, implemented as a benchmark problem for modeling decision making styles of human players (Holmgård et al.,



FIGURE 10.1: The levels used for collecting player data and for evolving procedural personas.

2014b). MiniDungeons levels are laid out on a grid of  $12 \times 12$  tiles: tiles can be walls (which obstruct movement), empty, or contain monsters, treasure, the level's entrance or exit. The player has full information of the level except for monsters' damage, as discussed below.

In MiniDungeons, a hero (controlled by the player) starts at the level's entrance and must proceed to the level exit: stepping on the exit tile concludes a level and loads the next one. A hero starts each level with 40 hit points (HP) and dies at 0 HP. The hero can collect treasure by stepping on treasure tiles: treasures have no in-game effect but a treasure counter is shown on the user interface. The hero can drink potions by stepping on potion tiles: potions heal 10 HP, up to the maximum of 40 HP. Finally, the hero can kill monsters by stepping on monster tiles: monsters do not move and only engage the hero if the hero moves onto their tile. Combat is stochastic: a monster deals a random number between 5 HP and 14 HP of damage to the hero and then dies.

For the purposes of collecting player data as well as for evolving procedural personas, ten MiniDungeon levels were created in advance (see Fig. 10.1). These levels were designed in a mixed-initiative fashion (Liapis et al., 2013c) and had several patterns which allowed different decision making styles to be exhibited. The authored levels have many branching points, but usually include an easy path (with minimal combat) between the entrance and the exit. Moreover, treasures and potions are often "guarded" by monsters, although some treasures are easily accessible and some monsters do not obstruct any paths. These patterns allow for different ways of traversing the level, as will be seen in Section 10.3.2.

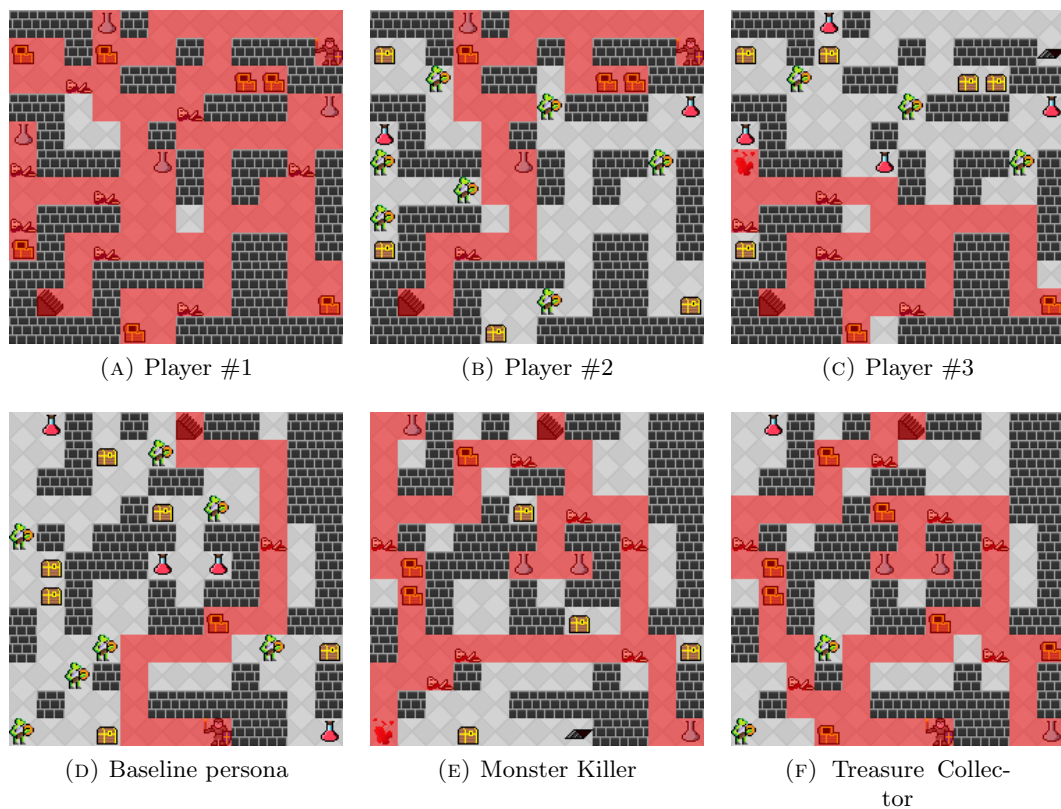


FIGURE 10.2: Playtraces of human players and evolved procedural personas of MiniDungeons.

### 10.3.2 Procedural Personas

The MiniDungeons game was created for two purposes: (a) to investigate how human players enact decision making styles in a simple game, and (b) to construct artificial agents able to represent such decision making styles.

A core assumption of decision theory (Kahneman and Tversky, 1979) is that human decision making under risk and uncertainty is shaped by *utility*. A utility function determines the decision maker’s willingness to take risks for an expected reward, and is considered idiosyncratic. In digital games, the game’s mechanics constitute affordances (Gibson, 1977) which are likely to be of utility to the player. Using the MiniDungeons game as a testbed, 38 participants played all 10 levels of Fig. 10.1, as covered in detail in Holmgård et al. (2014a): a few participants managed to collect all the treasures in every level (see Fig. 10.2a), while others rushed to the exit (see Fig. 10.2b) or miscalculated the risk of combat and died (see Fig. 10.2c). Such mechanics (treasure collection, death, reaching the exit) are thus likely sources of utility to players.

Procedural personas are artificial agents which represent archetypical decision making

styles. In MiniDungeons, procedural personas consider several gameplay and level elements as sources of utility: killing monsters, collecting treasures, reaching the exit, performing as few actions as possible, or avoiding death. Previous work identified five procedural personas: a Monster Killer, a Treasure Collector, a “baseline” persona, a Speedrunner and a Survivalist, respectively. For the purposes of this paper, generated dungeons will be evaluated by personas evolved on all dungeons of Fig. 10.1 as per Holmgård et al. (2014a). The controller for each persona is a combination of 7 linear perceptrons, with inputs being the hero’s HP and distance to different elements (e.g. closest potion, closest “safe” treasure) and outputs being the desirability of a strategy (e.g. go to closest potion, go to closest treasure that does not involve combat). The strategy with the highest value is selected by the agent; the decision is re-evaluated in every step rather than upon completion of the strategy. The perceptrons’ weights were evolved via an  $(\mu + \lambda)$  evolutionary strategy without self-adaptation. The fitness of each agent was calculated from the utilities collected after all 10 levels were played. Focusing on the personas used in this paper, the baseline persona received a boost to its fitness for every exit it reached; the Treasure Collector received a fitness boost for every treasure collected and a smaller boost for every exit reached; the Monster Killer received a fitness boost for every monster killed and a smaller boost for every exit reached. Optimizing the controllers for these fitnesses resulted in personas exhibiting very different behaviors (see Fig. 10.2d–10.2f).

The evolved procedural personas were compared to the human playtraces, in terms of persona-player agreement ratio. In every step a human took when playing, the persona was queried “what would be your next action given this game state?”: if the persona’s chosen action matched the human’s, the agreement ratio increased. Summarizing the results of Holmgård et al. (2014a), most players had the highest agreement ratio with the Treasure Collector persona, while a smaller number of players matched the Monster Killer persona.

### 10.3.3 Constrained Optimization of Game Levels

Previous experiments on the constrained optimization of game levels focused on generating *map sketches*, i.e. low-resolution, high-level abstractions of complete levels (Liapis et al., 2013c). Map sketches contain a small number of tiles which represent the most significant features of a level of a specific genre (e.g. weapon pickups in shooter games, player bases in strategy games). The simplicity of a map sketch allows it to be evolved in a straightforward and computationally lightweight manner. Map sketches of strategy games, rogue-like dungeons and first-person shooters have been evolved according to a generic set of objectives which can be customized to the game genre at hand (Liapis

et al., 2013d). The constraints of such map sketches revolve around the connectedness between level features: for instance, in a map sketch for a strategy game all bases must be connected (via passable paths) with each other and with all of the map's resources. In order to ensure constraint satisfaction, evolution has been carried out via a FI-2pop GA (Kimbrough et al., 2008) which can discover feasible individuals quickly and reliably even in highly constrained spaces (Liapis et al., 2013b).

MiniDungeon levels differ from map sketches in the fact that, despite a similarly small map size, they are directly playable. This introduces additional constraints on MiniDungeon levels in that they must be completable by procedural personas. Moreover, previous experiments optimized map sketches according to hard-coded objectives inspired by game design patterns (Björk and Holopainen, 2004), while MiniDungeon levels are evolved according to the play experience of the procedural personas that playtest them. In that regard, the procedural personas act as critics both on the playability and on the quality of the generated level: how this affects the evolutionary process will be explored in Section 10.5.

## 10.4 Methodology

This Section describes the two-population genetic algorithm used to evolve MiniDungeon levels, as well as the methods for assessing playability (the infeasible fitness function) and level quality (the feasible fitness function) via procedural personas.

### 10.4.1 Evolving MiniDungeon levels

A MiniDungeon level consists of 144 tiles, which can be empty or contain walls, monsters, treasures, potions, the level entrance or the level exit. In the genotype, a MiniDungeon level is represented directly as an array of integers: each integer describes the contents of a single tile in the level.

Due to the constraints on playability (discussed in Section 10.4.2), MiniDungeon levels are evolved via a feasible-infeasible two-population genetic algorithm (FI-2pop GA). The FI-2pop GA separates feasible individuals from infeasible ones (which do not satisfy one or more constraints), placing the former in a feasible population and the latter in an infeasible population (Kimbrough et al., 2008). The feasible population evolves to optimize the domain-specific measure of quality, while the infeasible population evolves to minimize its members' distance from the feasible border. As infeasible individuals approach the border of feasibility, the chances that their offspring will be feasible increase. Feasible offspring of infeasible parents migrate to the feasible population, and



vice versa: this indirect form of interbreeding may increase the size and diversity of the feasible population. In order to ensure that the feasible population is sufficiently large for efficient optimization, the *offspring boost* mechanism is applied to the FI-2pop GA. The offspring boost is applied in cases where the feasible population is smaller than the infeasible population, and forces both feasible and infeasible populations to produce an equal number of offspring regardless of the number of parents in each population.

In the experiments described in this paper, evolution of MiniDungeon levels is driven by asexual mutation alone; preliminary experiments showed that recombination is slower to discover feasible individuals and can result in multiple entrances or exits in the same dungeon. Mutation may transform an empty tile to a wall tile and vice versa, a level feature (non-wall, non-empty tile) may swap places with another level feature chosen randomly, or any tile may swap places with an adjacent one. Every offspring has 5% to 20% of its tiles (chosen randomly) mutated in the above fashion. By evolving content solely via this mutation scheme, an offspring is ensured to contain the same number of monsters, treasures, potions, level entrances and level exits as its parent. Parents are chosen via fitness-proportionate roulette wheel selection; the same parent may be chosen multiple times to generate offspring. In each population (feasible and infeasible), the best individual is transferred to the next generation unchanged.

#### 10.4.2 Assessing Playability with Personas

In order for a MiniDungeons level to be playable, a number of constraints need to be satisfied: (a) the level must contain a specific number of tiles of certain types, e.g. one entrance and one exit, (b) all features of the level (monsters, potions, treasures, exit) must be accessible via passable paths to the hero, and (c) the hero must be able to reach the exit without dying. Constraints of type (a) are automatically satisfied by seeding the initial population with levels containing the desired number of level features: since mutation does not add or remove features, the number of features in the initial population will remain constant throughout the evolutionary process. Constraints of type (b) require that a passable path exists between the level entrance and all other features in the level: levels that fail this constraint are evaluated based on how many features are inaccessible. Finally, constraints of type (c) require that an agent simulates a playthrough of the level. In order to ensure that the level can be completed regardless of the stochasticity of combat, a ‘worst-case’ scenario is constructed by assigning maximum damage (14 HP) to all monsters of the level. The agent chosen to perform the playthrough is the baseline persona, whose affordance is only to reach the exit: this persona does not get “distracted” by treasure or monsters, and is likely to finish the level quickly. If the baseline persona dies then this constraint is failed: however, an additional check for the

number of tiles explored by the persona is performed. This additional constraint was added after preliminary experiments in order to ensure that the entrance and exit are not close to each other, so that even speedrunners face at least a minimal challenge. If a baseline persona completes the level having explored less than 12 tiles, the level fails to satisfy the constraint of type (c) and is evaluated based on how many tiles the baseline persona explored, or a worse score if the baseline persona died.

Combining constraints (b) and (c) into a fitness measure for infeasible content, the distance to feasibility is calculated via  $d_{inf}$  of eq. (10.1). The infeasible population evolves to minimize  $d_{inf}$ , which increases the chances of feasible content being discovered. Observing  $d_{inf}$ , there is a clear priority between constraints: levels that fail constraints of type (b) automatically fail constraint (c) and assume that the baseline persona died without even testing for it. Moreover, if a baseline persona dies then the level receives a much worse score than if it completes the level, even within a very small number of steps. This aims to guide infeasible content towards first becoming well-formed (with all features accessible to the hero), then minimally playable for the baseline persona.

$$d_{inf} = \begin{cases} 1 + \frac{u_N}{N} & \text{if } u_N > 0 \\ 1 & \text{if baseline persona died} \\ \frac{1}{2}(1 - \frac{s_B}{C_s}) & \text{if baseline persona completed the level with } s_B < C_s \end{cases} \quad (10.1)$$

where  $N$  is the total number of level features (monsters, potions, treasures, exit) and  $u_N$  is the number of features which are not accessible from the level entrance;  $s_B$  is the number of tiles explored by the baseline persona in the worst-case scenario (all monsters dealing maximum damage) and  $C_s$  is the minimum number of explored tiles for a level to be considered feasible ( $C_s = 12$  in this study).

### 10.4.3 Assessing Level Quality with Personas

The main contribution of the procedural personas is towards the evaluation of feasible, playable game levels. However, it is not obvious what a persona (or indeed the human players it represents) looks for in a level. Granted that the decisions of procedural personas are shaped by their own utility functions, only events which affect their utility should be considered. This paper will consider the two most dominant (and distinct) procedural personas of past experiments: the Monster Killer (with a utility for killing monsters and reaching the exit) and the Treasure Collector (with a utility for collecting treasure and reaching the exit). Most playtraces of the 38 human players who tested

MiniDungeons matched the Treasure Collector persona (86%), while the Monster Killer was second (8%). When evaluating a level it has just finished playing (either by reaching the exit or by dying), the Monster Killer assigns the score of eq. (10.2) while the Treasure Collector assigns the score of eq. (10.3). The values of  $C_m$ ,  $C_t$  and  $C_r$  are taken directly from the fitness function which guided the evolution of each persona's controller<sup>1</sup>; the persona was evolved on 10 authored levels (see Fig. 10.1) and was evaluated on how it represents an archetypical decision making style (a Monster Killer that kills most monsters, a Treasure Collector that collects most treasure) (Holmgård et al., 2014a). Inversely, the scores of eq. (10.2) and (10.3) evaluate whether the level provides the desired utilities to personas that play optimally towards attaining them.

$$S_{MK} = \frac{(d_m C_m + C_r r)}{(N_m C_m + C_e)} \quad (10.2)$$

$$S_{TC} = \frac{(d_t C_t + C_r r)}{(N_t C_t + C_e)} \quad (10.3)$$

where  $N_m$  and  $N_t$  is the number of monsters and treasures in the level respectively;  $d_m$  and  $d_t$  is the number of dead monsters and collected treasures respectively;  $r$  is 1 if the hero reached the exit and 0 if not;  $C_m$ ,  $C_t$  and  $C_r$  are constants expressing the priority of monsters, treasures and level completion (respectively) in each persona's utility; for these personas  $C_m = C_t = 1$  and  $C_r = 0.5$ . The denominator normalizes the score of eq. (10.2) and (10.3) between 0 (no affordances acquired) and 1 (all affordances acquired).

Intuitively, a persona prefers levels that allow it to maximize its utility function: i.e. a Monster Killer prefers levels that allow it to kill all monsters and a Treasure Collector prefers levels that allow it to collect all treasure. Due to the stochastic nature of combat, the same level is played by a persona multiple times (10 in this paper) with damage for each monster randomized in each playthrough. When maximizing the level's utility for a persona, the simulations'  $S_{MK}$  and  $S_{TC}$  scores are averaged in the fitness of eq. (10.4) for a Monster Killer, and eq. (10.5) for a Treasure Collector, respectively.

$$F_{MK} = \frac{1}{R} \sum_{i=0}^R S_{MK}(i) \quad (10.4)$$

$$F_{TC} = \frac{1}{R} \sum_{i=0}^R S_{TC}(i) \quad (10.5)$$

---

<sup>1</sup>The fitness function of all personas' controllers included a penalty for taking extraneous actions. Since this penalty was a control mechanism to avoid playthroughs taking too long rather than an explicit utility, it is omitted for the purposes of level evaluation.

Maximizing the utility function of a persona, however, may be somewhat naive considering the decisions taken within MiniDungeons. Maximizing the utility of a Treasure Collector, for instance, can be trivially solved by placing all the treasure in a straight path between the level entrance and the level exit. In such cases, the player does not take a decision at any point during play; there is no risk/reward where the idiosyncratic utility function would shape the decision. In order to provide an element of risk, and thus require that the persona makes *meaningful* decisions, the level can be evaluated on how different a playthrough is from the next. Due to the randomness of combat, different playthroughs by the same persona may result in a premature death, in more or fewer treasures collected or monsters killed. Using the standard deviation of  $S_{MK}$  and  $S_{TC}$  among the 10 simulations, eq. (10.6) (for a Monster Killer) and eq. (10.7) (for a Treasure Collector) aim to maximize the levels' risk involved in personas' decisions.

$$D_{MK} = \sqrt{\frac{1}{R-1} \sum_{i=0}^R (S_{MK}(i) - F_{MK})^2} \quad (10.6)$$

$$D_{TC} = \sqrt{\frac{1}{R-1} \sum_{i=0}^R (S_{TC}(i) - F_{TC})^2} \quad (10.7)$$

## 10.5 Experiments

The experiments described in this section test how the different procedural personas (Monster Killer and Treasure Collector) and different fitness functions of eq. (10.4)-(10.7) affect the evolutionary process and the final generated dungeons. Dungeons generated in this paper have the same properties as those of Fig. 10.1: a  $12 \times 12$  tile grid containing one entrance, one exit, 8 monsters, 7 treasures and 4 potions (21 level features in total). All experiments in this paper were performed with a population size of 20 (including feasible and infeasible levels), and evolution runs for 100 generations; results were averaged from 20 independent evolutionary runs and each level is evaluated by a procedural persona via 10 playthroughs.

### 10.5.1 Discovery of feasible content

Despite the small map size of MiniDungeons, the constraints of connectivity of 21 level features and that of baseline persona survival were expected to make discovery of feasible individuals by random chance highly unlikely. Out of  $10^6$  randomly initialized levels, 360 were feasible (for all constraints) and 958 satisfied the constraints of connectivity, i.e.  $u_N = 0$  in eq. (10.1). Evolving infeasible individuals allowed the FI-2pop GA to discover playable levels quickly despite the limited population size: the first feasible

individual was discovered on average after 14.39 generations<sup>2</sup> (standard error: 1.40). This performance of the FI-2pop GA can be compared with a single population approach which handles infeasible individuals by applying the death penalty (i.e. fitness of 0). Using the same parameters as the FI-2pop GA and performing 20 evolutionary runs with each of eq. (10.4)–(10.7) (80 runs in total), the single population approach did not discover any feasible individuals in 21 of 80 runs (while all runs of the FI-2pop GA discovered playable levels). Moreover, among those runs where feasible individuals were found when using the death penalty, discovery of playable levels occurred after 35.42 generations (standard error: 0.84). As the difference in generation of discovery between FI-2pop GA and single-population GA is statistically significant ( $p < 10^{-6}$  via two-tailed Student's  $t$ -test assuming unequal variances), it is clear that the FI-2pop GA can discover playable MiniDungeon levels faster and more reliably.

### 10.5.2 Quality of feasible content

Figure 10.3 displays the best final evolved levels of 20 evolutionary runs, for each fitness function of eq. (10.4)–(10.7). To better demonstrate the levels' gameplay, each level is accompanied by a visualization of different playthroughs of the persona that evaluates it. Levels evolved towards  $F_{MK}$  tend to allow access from the entrance to the exit as well as to most potions (i.e. no monsters guard those level features); therefore it is the players' decision to pursue combat without it being forced upon them. Levels evolved towards  $F_{TC}$  tend to leave most treasures unguarded (in Fig. 10.3f only one treasure, near the exit, is guarded by a monster) and therefore collecting all treasures is not a risky choice for the player. Levels evolved towards  $D_{MK}$  tend to place more monsters at chokepoints, therefore guarding many of the level's features such as the exit, potions and treasure: in Fig. 10.3k the hero must face a minimum of two monsters in order to reach the exit, and a minimum of three monsters to reach the treasures in the middle of the map (the exit tile can not be crossed as it ends the level). Levels evolved towards  $D_{TC}$  similarly place monsters at chokepoints: in Fig. 10.3l two monsters must be fought to reach the exit as well as the treasures in the middle of the map. While the Treasure Collector persona could theoretically have fought those two monsters and gained access to the 6 otherwise unguarded treasures, it opted to go for the bottom right treasure which often caused it to die. This odd decision demonstrates the bias introduced by the representation of the personas' controllers (the Treasure Collector went for the closest guarded treasure in this case) and by the levels they were evolved on (which rarely had so many monsters clustered in a map corner). This issue will be further discussed in Section 10.6.

---

<sup>2</sup>Since the infeasible fitness ( $d_{inf}$ ) is the same for all experiments, discovery of the first feasible individual is calculated based on all four sets of experiments ( $F_{MK}$ ,  $F_{TC}$ ,  $D_{MK}$ ,  $D_{TC}$ ).

To evaluate the quality of a generated level, the utility function of its persona-critic is a straightforward performance metric. Expanding on that, the quality of the persona's playthroughs in each level can be captured by other gameplay metrics, such as number of tiles explored, actions taken or times the persona died. Table 10.1 contains the gameplay metrics of the best final evolved levels as evaluated via 10 playthroughs of its persona-critic. Values in parentheses represent the deviation between playthroughs of the same level (rather than deviation between levels). For comparative purposes, Table 10.1 includes the gameplay metrics of the authored levels of Fig. 10.1, on which the personas were evolved. Observing Table 10.1, there is a clear difference between Monster Killer personas and Treasure Collector personas: Monster Killers kill far more monsters (unsurprisingly), drink more potions, die far more often and take much more damage than Treasure Collectors. Comparing between levels evolved towards  $F_{MK}$  and  $D_{MK}$ , the former can be played by a Monster Killer persona more efficiently: more monsters are killed, more potions drunk and less deaths occur than with  $D_{MK}$ . The high death ratio of  $D_{MK}$  is a direct result of the fitness computation: the most straightforward way to achieve a larger deviation in monster kills is by dying prematurely. This is achieved in the map design by “hiding” potions behind multiple monsters, whereas maps evolved towards  $F_{MK}$  allow the hero to heal at any time (see Fig. 10.3). Comparing between levels evolved towards  $F_{TC}$  and  $D_{TC}$ , it is obvious that the former present minimal challenge to the Treasure Collector persona: with  $F_{TC}$ , all 7 rewards are always collected — without the hero ever dying — in every simulation and in every best final level. In contrast, with  $D_{TC}$  the hero collects less treasure with a high deviation in treasure collected between playthroughs, and has some chance of dying. Interestingly, the chance that the Treasure Collector dies is lower for maps evolved towards  $D_{TC}$  than for authored maps on which it was evolved; this is different than with maps evolved towards  $D_{MK}$ , where the death ratio is higher than for authored maps. Observing the Treasure Collector's actions in maps evolved for  $D_{TC}$ , its cautious tactics (compared to the Monster Killer) led it to rush to the exit when at low HP, since unguarded treasures were rarely available.



FIGURE 10.3: Best evolved levels for the different fitness functions of eq. (10.4)-(10.7). The levels shown have the highest fitness among 20 independent runs. Above each level are two playthroughs of the persona for which the level is evolved (Monster Killer or Treasure Collector), with randomized damage values for each monster.

	Monsters	Treasures	Potions	Explored	Actions	Death Ratio	Damage
Monster Killer							
Auth.	7.47 (0.47)	0.95 (0.21)	3.77 (0.15)	45.19 (4.57)	70.64 (10.34)	0.68 (0.42)	68.64 (5.25)
$F_{MK}$	7.91 (0.23)	2.90 (0.28)	3.99 (0.05)	44.34 (2.32)	78.81 (9.11)	0.43 (0.47)	72.63 (5.64)
$D_{MK}$	6.16 (1.15)	2.39 (0.45)	3.58 (0.51)	35.66 (6.09)	53.70 (12.75)	0.92 (0.13)	54.23 (6.16)
Treasure Collector							
Auth.	5.93 (0.42)	6.52 (0.61)	2.47 (0.49)	52.03 (6.12)	84.25 (14.42)	0.29 (0.42)	55.72 (4.18)
$F_{TC}$	2.68 (0.03)	7.00 (0.00)	3.57 (0.02)	43.57 (0.15)	75.89 (0.68)	0.00 (0.00)	25.29 (4.97)
$D_{TC}$	3.30 (0.83)	4.59 (2.07)	2.34 (1.29)	30.63 (12.39)	41.97 (21.55)	0.17 (0.19)	30.84 (9.26)

TABLE 10.1: Metrics of the best final levels, derived from simulations with procedural personas. Each level is simulated 10 times, and the value in the table represents the average of those 10 simulations, averaged again across the 20 independent runs of the GA. The value in parentheses represents the standard deviation of that metric within the 10 simulations (on the same level), and is also averaged across the 20 runs of the GA. Included are the gameplay metrics of the authored levels of Fig. 10.1: the values are averaged from 10 simulations, with deviation between simulations (on the same level) in parentheses.



## 10.6 Discussion

The experiments in Section 10.5 demonstrated the impact of the FI-2pop GA in the swift and reliable discovery of playable MiniDungeon levels. Moreover, the influence of the persona-critic was shown in the evolved dungeons' design patterns: levels evolved according to a Monster Killer had many unguarded potions while levels evolved according to a Treasure collector had many unguarded treasures. However, optimizing for most monsters killed or treasures collected resulted in MiniDungeon levels of limited interest, especially for  $F_{TC}$  where there was no risk of dying when collecting all treasure. In contrast, maps evolved towards deviations between monsters killed ( $D_{MK}$ ) or treasures collected ( $D_{TC}$ ) featured a higher chance of dying for either persona, and therefore interesting risk/reward decisions. Maps evolved towards either  $D_{MK}$  or  $D_{TC}$  are superficially similar, as both fitnesses result in levels with more monsters guarding potions and treasure; the difference in gameplay metrics, therefore, is introduced by the different decisions and utility functions of the personas playtesting them. It may be worthwhile in future work to explore the potential of evolving maps based on how different the playthroughs between these two personas are.

However, the design patterns of evolved levels were biased by the personas' architecture as well as the levels that they were evolved on. Using two inputs for estimating the utility of treasure (closest treasure and closest unguarded treasure) works well for the authored levels the personas were evolved on (which had several unguarded treasures) but fell short when all treasures were guarded e.g. in Fig. 10.3l. Additionally, following a strategy such as "collect closest treasure" should avoid monsters when possible by using more sophisticated planning approaches than the ones currently in place. Finally, future work can explore how dungeons can be evolved according to personas with more elaborate utilities (e.g. a completionist persona targeting both monsters and treasure), or according to clones of human players, i.e. artificial agents evolved to match the decisions of a specific human player (Holmgård et al., 2014d), thus providing personalized dungeons.

The algorithms covered in this paper can be applied to any problem that includes search in constrained spaces using simulations to evaluate content quality. Within games, procedurally generated content usually has to satisfy certain constraints; such constraints can be tested via planning (Horswill and Foged, 2012), ensuring that a "perfect" or "worst-case" player can finish the game. However, in games with high stochasticity (e.g. roguelike games), with emerging tactics (e.g. multi-player strategy games) or where players don't always play optimally (e.g. sandbox games), simulations using one or more artificial agents to test the game can be useful both for playability checks (assuming more

human-like perception, cognitive load and response times) and for evaluating the quality of completed playthroughs. Beyond games, constrained optimization is extensively applied in evolutionary industrial design (Michalewicz et al., 1996) where simulations are often used to test robot locomotion or the performance of a machine part. The results of these simulations can act as constraints (e.g. minimal distance covered by a robot or lifetime of a machine part) in order to divide the search space into feasible and infeasible, allowing the FI-2pop GA to explore it using simulation-based fitnesses on the feasible and infeasible population.

## 10.7 Conclusion

This paper described a method for using procedural personas to evaluate the playability and quality of generated levels for the MiniDungeons game. Playability is determined by a “baseline” persona playing through a worst-case scenario of the level, with monsters dealing maximum damage. Using a two-population genetic algorithm to distinguish between feasible and infeasible content, discovery of playable levels is fast and reliable despite the highly constrained search space. To test the level’s quality, a procedural persona simulates multiple playthroughs: a good level may require that the persona maximizes its utility or that the decisions taken by the persona affect its utility significantly. This paper tested two procedural personas, the Monster Killer and the Treasure Collector, and the final evolved levels demonstrated different map designs appropriate for each. Future work aims to improve the persona-critics, explore other simulation-based level evaluations, and increase the complexity of MiniDungeons.

## 10.8 Acknowledgements

The research was supported, in part, by the FP7 ICT project C2Learn (project no: 318480) and by the FP7 Marie Curie CIG project AutoGameDesign (project no: 630665).

## Chapter 11

# Monte-Carlo Tree Search for Persona Based Player Modeling

### Reference:

Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis (2015). “Monte-Carlo Tree Search for Persona Based Player Modeling”. In: *Workshop Technical Reports from the Eleventh AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (Player Modeling Workshop at AIIDE-15)*.

### 11.1 Abstract

Is it possible to conduct player modeling without any players? In this paper we use Monte-Carlo Tree Search-controlled procedural personas to simulate a range of decision making styles in the puzzle game *MiniDungeons 2*. The purpose is to provide a method for synthetic play testing of game levels with synthetic players based on designer intuition and experience. Five personas are constructed, representing five different decision making styles archetypal for the game. The personas vary solely in the weights of decision-making utilities that describe their valuation of a set affordances in *MiniDungeons 2*. By configuring these weights using designer expert knowledge, and passing the configurations directly to the MCTS algorithm, we make the personas exhibit a number of distinct decision making and play styles.

## 11.2 Introduction

This paper investigates new methods for automatic play-testing in games. Play-testing is an integral step of iterative game development. It allows game designers to test their assumptions about player behavior and to observe dynamics of the game system (Fullerton et al., 2004). In a sense, it is a partial mapping of the game space itself through observing where human players are capable of and interested in going within that space. As crucial as human play-testing is, it is also time consuming and potentially expensive. Further, it does not necessarily support a quick iterative loop when game designers are creating or fine-tuning new content for a game (e.g. game levels). Level designers can only test their levels with human players so many times, and for many developers minor changes cannot realistically mandate human play-testing. Instead, the designer informally infers or formally analyses the expected atomic and holistic impacts on each minor change in a level design, imagining what players might do, observing her own behavior in the level, or testing internally with her team. While this works for current game development practices we propose that there is a potential for supporting the level design process with generative player models. Generative player models acting as agents which play the game in lieu of players may provide game designers with surrogate play-traces to inform their design decisions or to integrate in their content creation systems. When a designer defines such an agent for a particular game we call them *procedural personas*. As player types akin to those described by Bartle (1996) and the play personas described by Tychsen and Canossa (2008) and Canossa and Drachen (2009), they describe archetypal ways of interacting with the game. They are formal representations of the game designer’s assumptions about her players.

Each persona may be used interactively or automatically in the level design process. Interactively, a level designer can inspect different interaction patterns (e.g. play-traces or completion statistics) in the level and iteratively adapt either the level or the persona behavior (Yannakakis et al., 2014). Automatically, a procedural content generation system can use the personas as critics that evaluate and change the generated content (Liapis et al., 2015) as part of a search based procedural content generation loop (Togelius et al., 2011). The purpose of procedural personas is thus to create easily configurable artificial game playing agents that believably simulate a variety of human decision making styles.

Inspired by decision theory the behaviors of procedural personas are controlled with simple utility functions (Mark, 2009) that designers can easily interpret, change and use to describe archetypal decision making styles. By assigning utility weights to game affordances, the designer prescribes what the different personas should prioritize. Methods for modeling player behavior from observed play traces and methods for creating AI agents that function as believable opponents are research areas that have seen much

attention and progress in the last decade (Hingston, 2012). However, methods that encode and model designer’s notions and expectations of player behavior from simple parameters, and realize these as observable behavior in-game, is still an under-explored part of Game AI (Smith et al., 2011; Yannakakis and Togelius, 2014).

In this paper we contribute to the development of AI agents as generative models of expected player behavior by demonstrating the use of procedural personas, configured by utility weights and controlled by Monte-Carlo Tree Search (MCTS), in our test-bed game *MiniDungeons 2*. We start out by grounding our work in decision theory, followed by a description of the *MiniDungeons 2* test-bed game. Afterwards, we describe our specific implementation of an MCTS controller for *MiniDungeons 2*. This controller is then used with 5 different procedural persona configurations. The personas are used to play a number of *MiniDungeons 2* levels, and their behaviors are recorded and contrasted to test whether meaningful differences arise from the different persona configurations.

### 11.3 Related Work

Two fundamental assumptions about games, drawn from decision theory, underlie the method suggested here.

The first assumption is that players’ decisions in games can be understood as being driven by utility. Whenever a player is making a move in a game she is trying to maximize her expected utility. The utility function is personal in the sense that different players will have different preferences when playing games. Their preferences need not be perfectly aligned with the rules of the game and may even change over the course of the game — although this problem is not treated here, as we try to express static utility functions. The utility function is expressed through play style or, more specifically, decision making style.

The second assumption is that players try to optimize their utility in a boundedly rational fashion, using analytic thinking to solve parts of the decision making task while using heuristics to solve other parts of the decision making task (Gigerenzer and Selten, 2002; Kahneman, 2011). The extent to which the player applies each kind of thinking depends on the player’s ability to do so — the player’s analytic skill and experience. It also depends on the player’s interest in devoting cognitive resources to solving the decision task in the first place, and the extent to which the player possesses specialized heuristics (from expertise) that can replace analytic effort (Gigerenzer and Gaissmaier, 2011). MCTS may be a well-suited algorithm for approximating this decision making process in a simulation, since it supports utility weights directly, is composed of an

analytical and a heuristic part, and its performance can be varied by changing its computational budget (Lorentz, 2011; Bourki et al., 2010; Zook et al., 2015). In this paper we explore the feasibility of using MCTS for generating different decision making styles in our test-bed game MiniDungeons 2. Before describing MiniDungeons 2, we briefly describe its predecessor that guided its design: MiniDungeons 1.

## 11.4 MiniDungeons 1 Game Design

Earlier attempts at modeling human decision making were made using the game *MiniDungeons 1* (MD1) (Holmgård et al., 2014b). As a predecessor to the MiniDungeons 2 game, described in this paper, MD1 had a much simpler rule set and a smaller set of affordances which affected human decision making. MD1 levels were laid out on a grid of 12 by 12 tiles, and tiles could be impassable walls, or passable tiles which could be empty or contain potions, treasures, or monsters. Unlike MiniDungeons 2, its predecessor only had one type of monster, which did not move. Combat in MD1 was stochastic: if a hero moved into the position of a monster, the monster would deal a random number of hit point (HP) damage and then die (with the hero moving to its tile). MD1 therefore revolved around the calculated risk of combat: could the hero survive another monster fight and what reward did the monster guard? Personas in MD 1 attempted to model how players pursued various affordances while deciding under uncertainty.

In contrast, MiniDungeons 2 acts as a test-bed for modeling combinations of analytical and heuristic decision making, moving closer to the puzzle genre. For this reason, monsters in MiniDungeons 2 move in a completely deterministic manner and their combat damage is easily predictable. Therefore, MiniDungeons 2 (with larger levels and more complicated mechanics) challenges the decision making skill of a player not in her decision making under uncertainty, but on the long-term risk of the level play-through, which is harder to analyze than in the smaller, simpler MD1 levels.

## 11.5 MiniDungeons 2 Game Design

MiniDungeons 2 is a deterministic one-and-a-half player game with full game state information available to the player (Elias et al., 2012). The game is designed specifically to have a high decision density, meaning that every action matters, while requiring little to no manual skill. It is a turn-based puzzle game where a *hero* travels from the *entrance* of a dungeon level to the *exit* (which loads the next level), similarly to many games in the rogue-like genre. Within the level, there are *potions*, *treasures* and *monsters* of different

types. The monsters move in response to the hero's action. Fighting happens when characters collide or when the hero or certain monsters conduct ranged attacks by throwing a *spell* attack or, in the case of the hero, a *javelin*. The hero may kill most monsters by fighting, but crucially, a special monster called the *minitaur* cannot be killed, but can only be knocked out for 3 turns through fighting. Every turn the minitaur will move directly toward the hero along the shortest path as determined by A\* path-finding, if it is not knocked out. This helps drive the game toward a terminal state, though it does not guarantee it. A state in MiniDungeons 2 will typically have 3-4 actions available, depending on the level in question. A typical level in MiniDungeons 2 takes 15-30 turns to play, depending on which goals a player chooses to pursue. Depending on monster setup some levels allow the player to play indefinitely, if she so chooses, by running away from the minitaur forever. Figure 11.1 shows an example MiniDungeons 2 level; the detailed rules of MD2 are described in Holmgård et al. (2015).

FIGURE 11.1: Map 7 from MiniDungeons 2.

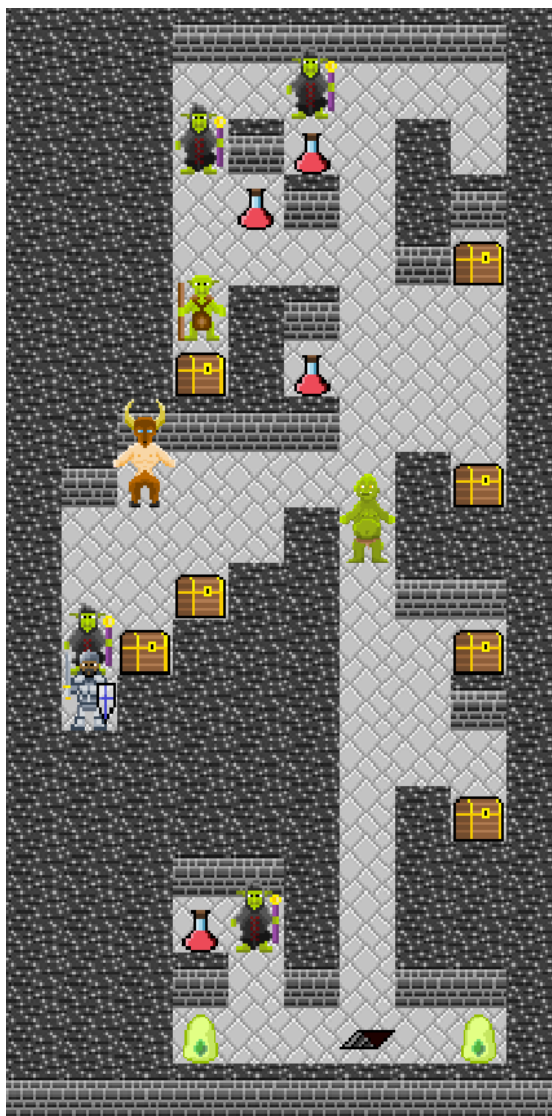


TABLE 11.1: All personas tested in the experiments. The five different personas are awarded utility from seven affordances in MiniDungeons 2: Taking a turn (Tu), reducing the distance to the exit (Di), killing a monster (M), collecting a treasure (Tr), drinking a potion (P), dying (D) or reaching the exit of a level (E).

Persona	Utility weights of affordances						
	Tu	Di	M	Tr	P	D	E
Exit	-0.01	1					0.5
Runner	-0.02	1					0.5
Survivalist	-0.01	0.5			0.5	-1	0.5
Monster K.	-0.01	0.5	0.5				0.5
Treasure C.	-0.01	0.5		0.5			0.5

## 11.6 Persona Design for MiniDungeons 2

As described above, the personas for MiniDungeons 2 are defined by their individual utility functions and mediated by their computational budget.

The sources of utility in MiniDungeons 2 are defined by the authors, acting as the designers of the game. They are comprised of what we consider to be the most important seven events that can occur in the game: spending a turn, moving toward the exit, killing a monster, collecting a treasure, drinking a potion, dying (and losing the level), and exiting (and completing the level). Table 11.1 shows the configurations of the five personas that were defined for MiniDungeons 2. The personas represent different imagined play styles: *Exit* (E) simply tries to finish the level. So do the other personas, but they have auxiliary goals shaping their decisions: *Runner* (R) tries to complete the level in as few turns as possible, *Survivalist* (S) tries to avoid damage and collect potions, *Monster Killer* (MK) tries to kill as many monsters as possible, and *Treasure Collector* (TC) tries to collect as many treasures as possible.

## 11.7 Monte Carlo Tree Search for Persona Control

Monte Carlo Tree Search (MCTS) is a stochastic tree search algorithm that has seen considerable success in some board games and digital games (Browne et al., 2012; Jacobsen et al., 2014; Champandard, 2012; Perez et al., 2015). It works by expanding each state depending on how promising it is (based on its reward) and on an estimate of how under-explored it is. Reward is calculated by repeatedly playing the game from the given state until a terminal state, and averaging the terminal state reward (win or loss). When using the algorithm for games where random play-outs are not guaranteed (or even likely) to lead to a terminal state within a reasonable time (e.g. many arcade games), the algorithm needs to be modified (Jacobsen et al., 2014). A common modification is to only perform the play-out for a set number of actions, and then evaluate



the end state using some heuristic. We use the standard UCB1 formulation of MCTS (Browne et al., 2012), but only perform play-outs for a maximum length of 10,000 actions and then use the utility function of the persona to evaluate the state. In practice, however, very few play-outs reach the 10,000 action limit and reach a terminal state long before. The utility function is applied to terminal states as well, and terminal states are specifically flagged as such by the game logic. The intent of our work is not only to develop agents that play the game well, but to develop agents that can model player preferences and skill as personas. Skill can be represented through the computational budget allocated to MCTS play-outs. We investigate this by varying the computation time available to the personas. It is important to note that the different utility functions require different information and therefore have marginally different complexities. However, all necessary information for all utility functions is always calculated by the game logic during game-play and play-outs. Therefore, we assume that the personas have the same amount of computation time available and are playing at similar “skill” levels. In spite of this we do not assume that the different persona preferences are equally easy or hard to enact. Being a Monster Killer may be a more difficult decision making style than being a Runner. While this is an interesting question in itself, this paper focuses on behavioral differences between personas with identical computational budgets and performance differences within personas across different computational budgets. Below, we briefly outline our strategy for comparing the behaviors of the various personas and their performances under varying computational budgets.

## 11.8 Metrics

In this section we describe the three different types of metrics we use to evaluate the implemented personas as procedural representations of imagined decision making styles: Action agreement ratios, summary statistics, and heat-maps.

### 11.8.1 Action Agreement Ratio

In order to establish that the personas are in fact enacting different decision making styles, we apply an agreement metric across individual actions. The metric we use to evaluate persona likeness, developed in previous work (Holmgård et al., 2014a) on persona/human comparison, is the *action agreement ratio* (AAR). AAR considers each step of a play-trace a distinct decision, in line with the high decision density of MiniDungeons 2. To produce the AAR between two personas, all distinct game states of a persona play-trace, the original, are reconstructed. For each game state, the other persona being compared for agreement is inserted into the game state and queried for the next

preferred action, essentially asking: “What would you do?”. If the two personas choose the same action, one point is registered. Finally, the AAR is computed by dividing the number of points with the number of decisions in the original play-trace. A perfect AAR score of 1.0 represents two personas that agree on every single decision.

In *Results* we describe the experiments we ran to generate persona behaviors and examine their performances and differences through summary statistics and through AAR.

### 11.8.2 Summary Statistics and Heat-Maps for Persona Identification and Skill Evaluation

Since the different personas are constructed to pursue different affordances in MiniDungeons 2 it is difficult to evaluate all of them using a simple unidimensional score system. Instead, we summarize the number of affordances reached during game-play and interpret these as indications of whether a persona is enacting a desired style. While this arguably is a subjective approach to determining whether personas enact a desired decision making style, the fundamental persona concept is subjective to the designer. Future work will focus on determining if personas are recognizable across observers, but here we directly interpret the summary statistics from a game design perspective. Further, we examine a number of heat-maps representing persona play-traces to identify patterns that characterize the behavior of the various personas.

For evaluating whether varying the computational budget has an impact on persona performance we apply a straightforward operationalist approach: For each persona we determine whether variation in the computational budget impacts the interaction with affordances that factor into the particular persona’s utility function. If a Monster Killer manages to kill more monsters or if a Treasure Collector manages to collect more treasure, we interpret this as an indication of greater skill. In the following section we first present the experiments we ran to obtain results and then analyze each of the metrics.

## 11.9 Results

The five MCTS personas (Exit, Runner, Survivalist, Monster Killer and Treasure Collector) were tested on 10 hand-crafted maps in MiniDungeons 2. The levels were crafted to represent varying degrees of difficulty and to allow for the expression of all five decision making styles. The levels were informally tested to ensure that this was the case. Five test conditions were defined with 10ms, 100ms, 1s, 10s, and 60s of computation time, respectively. Each controller was run 20 times on each map in order to take into

TABLE 11.2: Action Agreement Ratios (AAR) between all personas. The AARs range from 0 to 1 and indicate the extent to which two personas agree on atomic in-game decisions. The results indicate that personas agree with themselves to a larger extent than with other personas. However, it is also evident that some variation happens within personas, likely due to the stochasticity of the MCTS algorithm. All AARs were calculated with 1s of decision making time per action.

	E	R	S	MK	TC
Exit	<b>0.65</b>	0.49	0.42	0.40	0.43
Runner	0.43	<b>0.65</b>	0.59	0.46	0.56
Survivalist	0.42	0.56	<b>0.61</b>	0.45	0.51
Monster Killer	0.47	0.49	0.48	<b>0.64</b>	0.46
Treasure Collector	0.44	0.58	0.52	0.40	<b>0.68</b>

account the effect of stochasticity, and the means of the scores are reported here. All experiments were run on an Intel Xeon E5-2680 Ivy Bridge CPU at 2.80GHz.

### 11.9.1 Action Agreement Ratios

AAR values were obtained by cross-comparing all personas against each other with 1s of computation time. For each persona, the other four personas as well as a new instance of the same persona were presented with all 1s play-traces from the original persona (20 per map) and asked to evaluate each decision, yielding an AAR value. Each play-trace was evaluated 20 times to account for stochasticity. The average AAR values resulting from this process are presented in Table 11.2.

The results indicate a number of interesting things. No personas achieve perfect agreement, not even with themselves. This may be due to the stochasticity of the MCTS algorithm. The state space of MiniDungeons 2 is too large to be fully explored by MCTS. Hence, each instance of a persona may end up exploring slightly different parts of the state space and reach different conclusions about the best course of action. In particular, sometimes an agent walks back and forth while still implementing an overall optimal path. Secondly, all personas agree more with themselves than with any other persona, with a mean AAR within personas of 0.65 in contrast to a mean AAR between personas of 0.48. This difference is significant as measured by Welch’s t-test ( $t = 128.3, p < 0.01$ ), showing that the designer-defined utility weights for the different affordances clearly and consistently affect persona behavior.

### 11.9.2 Summary Statistics and Heat-Maps

Table 11.3 shows summary statistics for each persona, summed across all levels when different computational budgets are allocated. The results show similar patterns across

all computational budgets and highlight the similarities and differences in the behaviors of the personas.

In general the personas fail at completing the levels. In a few instances they do succeed in reaching the exit, mainly in the case of the Exit persona, but in general they die before completing any levels. This may be due to a general weakness in the MCTS controller design. The controller does not implement domain knowledge in the play-outs, which are completely random. This can lead to most play-outs providing little information which in turn can hamper performance. It can, however, also be due to the fact that the levels are somewhat hard. Future work comparing persona skill to human player skill should investigate this.

In relative terms, the Exit persona is by far the most successful of the personas. Depending on the computational budget, it completes between 15% and 39% of the levels. However, it may be argued that the Exit persona has the simplest decision making style, as it only has to optimize for reducing the distance to the exit, reaching the exit, and taking as few turns as possible; all of these goals align well within the rules of the game. The Runner persona, on the other hand, is typically unsuccessful in reaching the exit. The only difference to the Exit persona is a higher cost to spending a turn, but this changes the behavior to be unsuccessful. Notably, the Runner pursues almost no potions which stands in contrast to the Exit persona which seems to exploit potions to make it all the way to the exit. The Survivalist seems to focus shortsightedly on potions, which it collects to a great extent, but the number of potions in the levels seems to be too low to make this a viable strategy for also reaching the exit. The Monster Killer kills by far the most monsters of any persona, and collects a large number of potions too, to enable this mission. Finally, the Treasure Collector collects more treasure than anyone else, but typically fails at reaching the exit. This could either be because it prefers treasures over finishing the levels or the controller can not look ahead far enough to avoid trapping itself in fatal situations.

Figure 11.2 shows persona behaviors on various maps. The maps are chosen to highlight the behavior of each persona as different maps cater to different personas' decision making styles. The green color indicates tiles visited by the persona and the red color indicates tiles onto which the persona threw the javelin. Each map is shown in its final state.

All together the results indicate that the personas exhibit variation in behavior based on their utility functions. The differences in AAR values and summary statistics support this and the heat-maps serve as demonstrations. At the more general level, the personas exhibit a somewhat weak ability to play the game successfully, although this might be

TABLE 11.3: Play summary statistics summed across the 10 maps under different computational budgets: 10ms, 100ms, 1s, 10s, and 60s. Each condition totaled 200 runs per persona, 20 per map.

10 milliseconds					
	E	R	S	MK	TC
Turns	5368	3853	4106	4386	3982
Deaths	170	195	197	200	197
Monsters	721	531	571	741	531
Minitaurs	718	589	634	627	631
Treasures	111	90	78	86	175
Potions	57	11	34	32	14
Javelins	359	354	317	327	311
100 milliseconds					
	E	R	S	MK	TC
Turns	5017	2645	3328	4210	2778
Deaths	144	194	194	196	196
Monsters	800	481	554	931	490
Minitaurs	580	415	507	539	427
Treasures	113	77	84	92	210
Potions	75	0	57	65	3
Javelins	291	274	282	278	249
1 second					
	E	R	S	MK	TC
Turns	4116	2263	2831	3907	2513
Deaths	138	195	200	197	199
Monsters	761	482	570	969	476
Minitaurs	416	358	418	461	377
Treasures	110	57	68	88	203
Potions	45	0	70	63	5
Javelins	283	224	237	287	235
10 seconds					
	E	R	S	MK	TC
Turns	4066	2250	2576	3648	2603
Deaths	123	197	196	200	195
Monsters	769	455	544	1002	478
Minitaurs	459	372	375	403	417
Treasures	106	56	60	59	190
Potions	58	0	67	62	5
Javelins	243	233	225	237	217
60 seconds					
	E	R	S	MK	TC
Turns	3885	2110	2579	3611	2336
Deaths	122	197	199	198	192
Monsters	759	473	577	967	493
Minitaurs	408	337	392	406	349
Treasures	89	58	49	70	201
Potions	53	0	85	52	3
Javelins	230	202	215	253	223

addressed by changing details in the MCTS controller implementation and by fine-tuning the personas' utility functions.

## 11.10 Discussion

In this paper we demonstrated that combining the idea of varying simple utility functions with Monte-Carlo Tree Search for agent control allows us to express archetypally different decision making styles in MiniDungeons 2. The decision making styles modeled in this paper do not represent models of observed human players but instead represent formal models of archetypal players imagined by game designers. On the one hand, it is possible to criticize this as being altogether different from the usual conception of data-driven player modeling. On the other hand, it is our belief that game designers always work with informal, imaginary player models when designing games, and that helping the designer formalize these assumptions procedurally may be valuable to the design process, in line with play persona theory (Canossa and Drachen, 2009). The use of MCTS itself brings new possibilities to embedding synthetic play-testing in iterative level design processes. In previous work focusing on MiniDungeons 1 we have shown how other agent control methods, including reinforcement learning (Holmgård et al., 2014b) and neural networks configured through evolution (Holmgård et al., 2014a), can be used for the same purpose. However, the methods used for that game would not perform well in MiniDungeons 2, which has a dynamic environment (monsters move); the models learned for MiniDungeons 1 presumed a static world. Other recent work has shown how game playing agents that play across multiple games (while not incorporating any explicit notions of style or preference) can be produced using a combination of reinforcement learning and neural networks (Mnih et al., 2015) or MCTS (Perez et al., 2015). While off-line machine learning-based player modeling methods perform well when trained for a sufficiently long time and with a sufficient amount of training material, the need for training also constitutes their major disadvantage. If a game designer fundamentally changes the rules of a game, previously trained agents or personas may be rendered invalid in an instant. In contrast, on-line search based methods like MCTS are capable of adapting to unseen problems within the rules of a game (e.g. new maps) or to rule changes. If we add a new monster with an entirely new behavior to MiniDungeons 2 and include this in the simulations of the MCTS personas, it is likely that the personas would retain their individual characteristics while responding to the new element of the game. Prior work in MCTS for general game playing supports this assumption (Finnsson and Björnsson, 2008), but future work should focus on investigating this for the particular case of MiniDungeons 2. The scalable aspects of MCTS allow for a straightforward way of simulating player skill for personas. The results in this paper suggest that skill may

be represented through computational budgets, and recent work has provided deeper insight into MCTS and skill for other games (Zook et al., 2015). It is an open question to what extent MCTS based personas can replicate the behavior of a human player on an unseen map in MiniDungeons 2. The current personas are defined by hand, and are in a sense extremes within the space of strategies that can be represented with the current set of primary utilities. When sufficient player data has been collected from players of MiniDungeons 2, we should try to learn personas directly from player behavior, by varying the utility weights of the MCTS agents until the best possible fit is obtained. Future work revolves around using these personas as level design critics in procedural content generation tools for MiniDungeons 2.

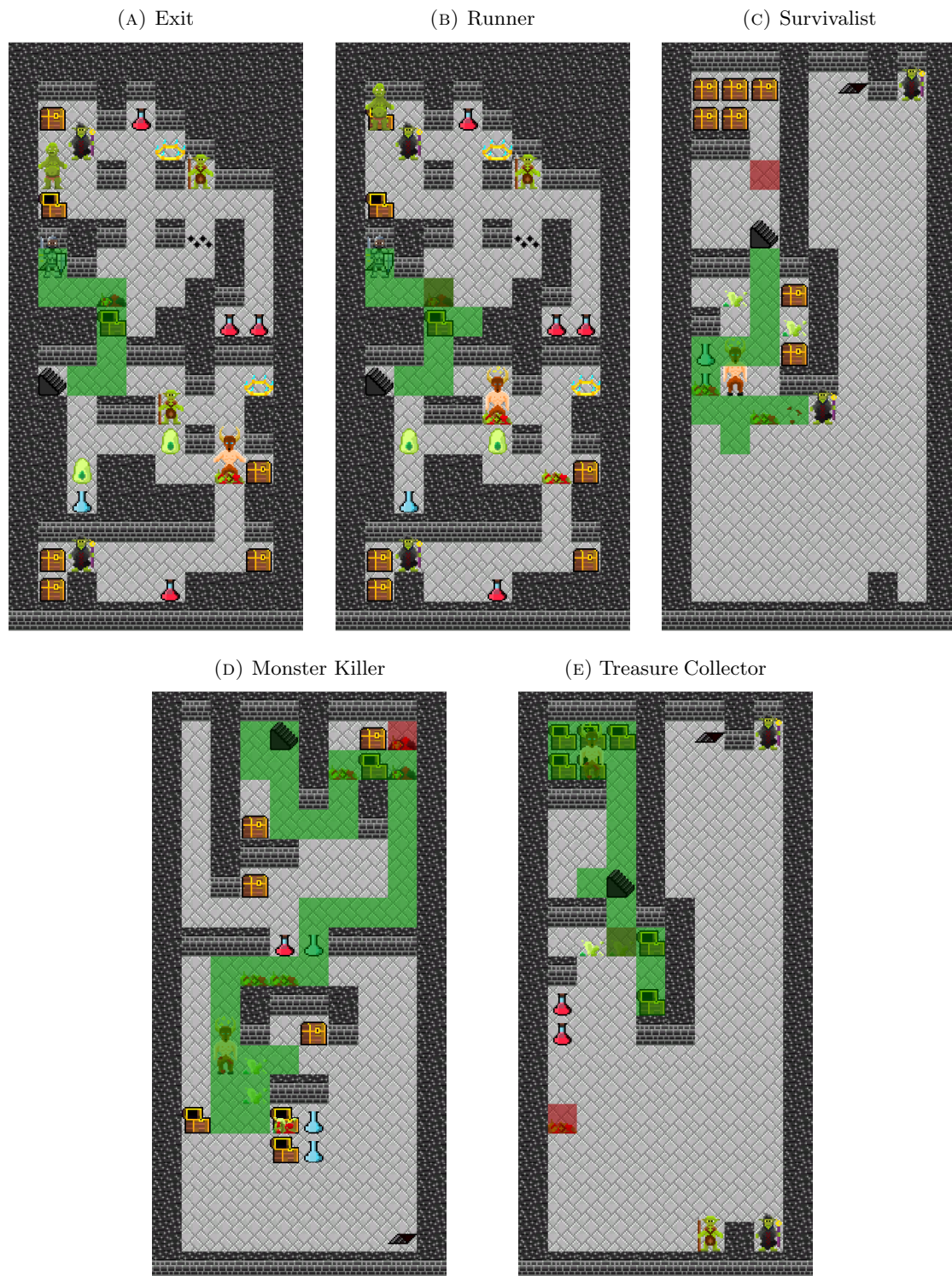
## 11.11 Conclusion

In this paper we have demonstrated how a number of simple utility functions, representing game designers' assumptions about player behavior, can be used with a Monte-Carlo Tree Search implementation to provide game playing personas that enact a gallery of decision making styles. We also argued that MCTS has a number of advantages for modeling player decision making styles: It matches contemporary high-level models of the human decision making process, it accepts utility functions in a straightforward manner, and can be adjusted to represent different skill levels by changing its computational budget. These characteristics make MCTS a good candidate for constructing generative player models across a wide range of games. Future work will focus on building MCTS-based data-driven generative player models from human play-traces, comparing these to the expert-knowledge-driven ones built from game designer expectations, and eventually applying these player models as critics for procedural level generation.

## 11.12 Acknowledgments

The research was supported, in part, by the FP7 ICT project C2Learn (project no: 318480), the FP7 Marie Curie CIG project AutoGameDesign (project no: 630665), and by the Stibo Foundation Travel Bursary Grant for Global IT Talents.

FIGURE 11.2: Heat-maps of end game states exemplifying differences in the personas' behaviors, taken from personas given 1s of decision making time. Different maps are chosen to best showcase the personas' decision making styles. Green indicates tiles a persona moved to and red indicates a javelin attack. In sub-figures (a) and (b) the Exit and Runner personas manage to finish the level, while in sub-figures (c), (d), and (e) the personas die before reaching the end of the level. The Survivalist pursues two potions, but ultimately dies, and the Monster Killer manages to kill all monsters in the level, but does not survive. Finally, the Treasure Collector gets all the treasure, but dies in the process.





## Chapter 12

# Discussion

In this chapter, we review the findings from the six preceding chapters, discuss the contributions and limitations of the procedural persona concept as well as the general attempt to model human decision making in games.

First, we discuss the fundamental notion of the thesis: that procedural personas can provide useful reference processes for making sense of potential and actual decision making styles in games.

Second, we discuss the validity of the findings from the studies of *MiniDungeons 1 and 2*, examining whether the reported results support the claim that procedural personas represent decision making styles.

Third, we go deeper into the operationalization of decision making applied in the papers, critique the use of affordances to identify decision making points in games, and discuss whether other indicators could be established and how. We also discuss whether the manual part of defining decision points could be supported by computational tools or even fully automated, pointing to related work that shows some progress in this direction.

Fourth, we discuss the metrics used throughout the papers to compare decisions among personas and human players and ask whether other metrics would be more appropriate.

Fifth, we examine how well the procedural persona method can be expected to extend beyond the *MiniDungeons* games and whether the method could be useful in real-world game development.

The above discussions are summarized in the last section which serves as the vantage point for the following chapter, which outlines future work for developing the procedural persona concept.

## 12.1 Establishing Reference Processes in State Spaces

One of the fundamental assumptions of this thesis is that the state spaces of most games, save for perhaps the simplest ones, are too large for an analyst, designer or player to mentally contain all at once or to practically explore completely. Not every possible state of a game or every possible sequence of states can be visited by a human observer. In some cases all game states may be calculated and visited by a computer and stored for inspection. This is demonstrated by Sturtevant (2013) who also argues that the range of games that may be exhaustively searched is rapidly increasing with computational power and storage. However, even then, these states cannot be grasped all at once by a human observer.

The procedural persona method suggests that valuable information can be obtained by instead working with reduced abstractions of games' state spaces and player behaviors, derived from a priori readings of the rules, to understand what is possible and interesting in a game. The concept is a way of establishing known reference processes in a game's state space, by defining affordances and player motivations and reducing the game's state space into a decision space. Projections back from this decision space onto the game's state space, through simulation, then allows a designer, analyst or a procedural content generation system to explore what is possible and interesting in the game, according to these definitions.

Of the six papers presented in the previous six chapters the first one, concerned with player behavior in the *Mario AI Benchmark*, may be considered the odd-one-out, since it does not explicitly contain the concept of procedural personas. The study of deviations from a reference agent in the *Mario AI Benchmark* is the first exploration of using agents as reference processes to make sense of a game. Being the first paper in the development of the concept, it suffers from a number of weaknesses: The distance metrics applied to measure the deviation between the reference agent and the human players operates only in one dimension and has trouble accounting for play-traces of varying lengths. Also, it has a hard time capturing and accounting for the tactical aspects of player decision making in platform games, such as exploiting patterns in scripted enemy behaviors or moving both back and forth in the level. Additionally, the agent is not a persona with multiple goals, defined by a utility matrix. It represents a rational player who acts strictly in accordance with the primary goal of the game, getting to the end as efficiently as possible. In a sense it solves the levels, but does not play them. Still, the method shows that comparisons with an agent with a known set of motivations and methods for enacting these can provide information about a game and its players and thus supports the idea of reference processes. The following five papers all build on this notion of reference processes in *MiniDungeons 1 and 2* instead, which are tailored to investigate

the idea of representing decision making styles. Their designs remove potential sources of noise that could be difficult to filter out in other game genres by aligning the state space with the decision space.

It should be noted that other methods than the procedural persona one could be considered equally relevant for establishing reference processes for making sense of the dynamics of a game or for guaranteeing properties of procedurally generated content. Examples could include methods for discovering a range of possible play styles by generating random actions instead of taking the controlled persona approach (Bauer et al., 2013; Isaksen et al., 2015), generative approaches based on constraint satisfaction (Smith et al., 2013), grammar based agents (Xu et al., 2014) or reasoning, analytic agents (Shaker et al., 2013a), to name a few successful examples. The procedural persona approach may be added to this gallery of methods where the appropriate approach will depend on the particular interest of the analyst or designer. This begs the question of whether the procedural persona method actually models decision making styles or just play styles: do procedural personas validly represent *decision making* styles?

## 12.2 Validity: Do Procedural Personas Represent Decision Making Styles?

The first argument for the validity of the procedural personas being models of decision making is that they are well-grounded in decision science. Decision theory, bounded and adaptive rationality, and recognition primed decision making are complementary and account for different parts of human decision making: The rational, the adaptive, and the immediate.

Insights from the three perspective are integrated into the personas through the use of utility functions and through the agent controller architectures that match the overall processes described by the theories. Still, the personas are limited generative models human decision making and they do not represent human decision at a detailed level. A detailed, accurate representation falls closer to the goals of elaborate cognitive architectures which comprise a whole research direction within cognitive science. Such models are most likely unnecessarily complex for the purpose of understanding game state spaces or generating game content and would probably not be helpful as player modeling tools due to their complexity. The procedural personas are aimed to be as simple as possible while still representing decision making styles in games.

A second question is whether the utility functions are a valid way of representing the motivations that underlie different play styles in games. One answer is to, again, point

to observations from the decision sciences where behavior can be explained from the notion of utility. A second answer is that other work within games also shows that in-game actions can be related to in-game affordances via variations in motivations (Drachen et al., 2009a; Canossa, 2012) and constructs like personality type (Yee et al., 2011b; Canossa et al., 2015). Affective, objective information would be another source of information for inferring player motivations. As touched upon in Chapter 2, there is evidence that signals like heart-rate and skin conductance, to name a few, can be used to identify what decisions players find interesting and important (Bechara et al., 1997; Yannakakis and Hallam, 2008; Ahn, 2010). Establishing correspondence between physiological indicators of engagement in decision making and the learned utility functions would provide stronger support for the validity of procedural personas as player models. Since utility functions indicate what affordances players care most about, physiological indicators should match this.

Another way of evaluating the validity of the procedural persona concept is to examine its capacity as a player classifier. The papers included in this thesis shed some light on this question, although more research is needed. Results from the papers relating to *MiniDungeons 1* show that the human players, classified by the personas, predominantly ended up in one class. This classification imbalance could be influenced by several factors: The controller implementations, the chosen utility functions, the level designs, or they could truly represent differences in player decision making preferences and styles. More studies, including wider samples of players across more levels, preferably across several games, and with the added collection of external measures such as intelligence tests, personality tests, and objective measures from physiology, are needed to provide a deeper evaluation of the procedural persona method as a player classifier.

The aspect of procedural personas that has seen the most comprehensive investigation in the papers in this thesis is the method's ability to predict individual decisions of individual players. Across all the different approaches to predicting player decisions, the best matching procedural personas seem to be able to reproduce player decisions with an accuracy of 60 – 70% in Chapter 9, depending on the metric used. While this is clearly better than a random agent, which never attained an accuracy greater than 43%, it leaves room for improvement as predictive player models.

At least three reasons for why the personas do not attain better accuracies seem plausible:

Firstly, the chosen controller architectures may not be able to represent the decision making styles of players. In the paper featuring personas and clones, the specialized agent results indicate that the evolved perceptron-based controller is capable of learning

to predict up to 85% of players' decisions on average. While this seems a more acceptable performance it does indicate that a more complex controller architecture might be able to learn a greater accuracy. Unfortunately, player data does not exist for *MiniDungeons 2* yet, so the accuracies of the Monte-Carlo Tree Search based personas are not known. Regardless, there is no question that the validity of the method depends on the performance of the chosen controller.

Secondly, the utility functions that are used for controlling the performance of the personas, and hence predicting the actions of individual players, are in no way fitted to the properties of the sample that they are modeling. Players are assigned to whichever persona is the best fit and this persona is then used as the best predictive model. The predictive performance of the personas could most likely be enhanced significantly by applying an iterative process of classifying players and adjusting utility functions, until a desired distribution of player classifications and a maximal level of predictive performance was attained. Each of these clusters could then be characterized by their distance to the original personas and be labeled anew by the persona designer. As such, it seems reasonable to assume that the predictive performance of the method could be improved through further research.

Thirdly, the predictions derived from the procedural personas is dependent on the metrics that were defined for measuring decision agreement. The action, tactical, and strategic agreement ratios depend on the affordances that are chosen for a particular game, in our case *MiniDungeons 1 and 2*. The metrics are subjective in the sense that they are defined by the game designers, who use them to describe what they consider important. Other designers might have other evaluations and define the metrics differently. On one hand, this is intentional as the method aims to enable sense-making of specific games' state spaces. On the other hand it makes it hard to compare performance across games: There is no way to guarantee that it will work well in a game before it has been tested. An approach to partially solving this problem could be to explore the performance of the method across groups of games that are classified as being similar within some taxonomy. A framework such as game design patterns (Björk and Holopainen, 2004) might be well suited for this, but there is no data to support this assumption yet.

### 12.3 Metrics and Operationalization

In terms of the three metrics developed in this thesis, they too are grounded in the theoretical basis of decision sciences. While the implementation of each is metric is and game specific, the abstract definition of each level of decision characterization in games is general and applicable across games. The best way, perhaps, to evaluate

the validity of the metrics, would be to compare them to other measures of decision making in games. At this point in time, to the best of my knowledge, no other general metrics for characterizing and predicting player actions in games focus specifically on decision making. Still, as described in the introductory chapters and the papers here, the interpretative jump from an action to a decision can be considered relatively small, as long as the action is significant in the context of the game. At least one other framework in the literature addresses the problem of classifying players based on their inputs and actions: the Gamalyzer method by Osborn and Mateas (2014). While not included in this thesis, the validity of the three decision agreement metrics developed here could be improved by comparing classification results from the persona method with classification results from the Gamalyzer method.

## 12.4 Extensibility and Usefulness

Another in relation to the value of procedural persona concept is to which extent it generalizes to different kinds of video games. The *MiniDungeons* games, although different in mechanics and dynamics, share many similarities by both being turn-based dungeon roaming games that take place on relatively small maps where the whole level layout is visible to the player. Other research working with the notion of procedural personas for different games has recently emerged (Brown, 2015), but does not yet have results on the performance of the method. The *MiniDungeons* games were characterized by a tight alignment between the games' state spaces and decision spaces. Being turn based and discretized, every action in the games significantly changed the game state and even at the action level the personas were making meaningful decisions.

A way of investigating the extensibility of the method would be to identify a game where the game's state space is loosely aligned with the decision space. This could be a game where many inputs or actions are needed to generate few meaningful decisions. One example could be a Massively Multi-player On-line Role Playing Game, like e.g. *World of Warcraft* (Blizzard Entertainment, 2014). *World of Warcraft* takes place in a continuous three-dimensional world where a relatively large amount of time is spent traveling between encounters and destinations. Many encounters with enemies and other challenges in *World of Warcraft* are similar and repetitive, yet they still require significant tactical decision making when they take place. *World of Warcraft* also features a number of highly technical challenges called *instances*, which require teamwork and coordination, social interaction, and crafting and trading, among many other features. The game's decision space is vast, but due to the continuous and real-time nature of the game environment, the game's state space is many orders of magnitude larger. Still, it is

well documented that different kinds of players exhibit different play styles in *World of Warcraft* (Yee et al., 2011a), and it would follow from our previous argument that these players exhibit variation in decision making styles when faced with the various challenges in the game. Does the procedural persona concept have any usefulness or applicability to a game of this scope and scale? While a thought experiment, we can imagine how procedural personas might be helpful for designers creating new content for a game like *World of Warcraft*.

Creating personas that could operate within the actual state space of the game would most likely be very difficult, as an agent capable of playing *World of Warcraft* like a human player in every capacity is beyond the current state of the art in game artificial intelligence. If, however, certain aspects of the game were extracted and modeled as reduced decision spaces, abstractions over the actual game of *World of Warcraft*, the persona method might be useful. An *instance* in *World of Warcraft* (a certain, delineated challenge in the game) might quickly be modeled as a graph of decisions by the level designers. From there, assumptions of designers about how players of different decision making styles could be formalized as utility functions and simulations could be run over this abstract graph representing the *instance*. Changing the probabilities of events occurring in the instance, the difficulty of enemies or the availability of other kinds of affordances, and re-running persona-based simulations, could relatively quickly inform designers about the consequences of design changes, before they were implemented as costly alterations to the game content.

The example remains a thought experiment and unproven, but suggests that for modern games of high complexity, a simulation of players operating on a reduced simulation of the game might be a useful support tool for game development. To the extent that procedural content generation tools could be interjected to generate content from abstract representations of decision space, it might even be possible to automatize further steps of this process. Obviously, significant improvements upon the current state-of-the-art in procedural content generation methods for games would be needed, but work on mixed-initiative tools for content creation (Liapis et al., 2013c) and automatic game generation (Cook et al., 2014) point to the future feasibility of such methods.

## 12.5 Chapter Summary

In this chapter we revisited the core assumptions, strengths, and weaknesses of the procedural persona concept and discussed its validity, extensibility, and potential. In the following chapter, we outline the future work that could be undertaken to expand upon

the concept and developed new methods for player decision modeling and procedural content generation.



## Chapter 13

# Future Work

In the previous chapter we discussed the main assumptions of this thesis and visited some of the main limitations in the shown validity, generalizability, and applicability of the procedural persona method. We identified a number of areas where further research is needed if the method is to be considered a valid and general approach for modeling and simulating player decisions in video games. Here, in this chapter, we expand on the identified future work.

First, we discuss the prospective of learning player utility functions and suggest one approach for accomplishing this in practice. Then, we visit the question of the performance of applied agent control architectures and whether other architectures would be better suited and what benefits and trade-offs other approaches might imply. Further building on the player modeling perspective, we suggest that procedural personas, used as individual player models, should be correlated and examined for correspondence to external psychometric measures. Also, objective measures, such as affective information collected during play-testing, could advantageously be used as supporting data sources for building procedural personas. Relatedly, we discuss the possibility of conducting user studies with game and level designers to further investigate the claim that procedural personas can produce information that is useful in iterative content creation for games, enabling computer-assisted content design. We also outline a future research agenda for implementing procedural content generation for *MiniDungeons 2*, and for expanding the testing of the method to other games and genres, and imagine how a more general game design support tool could be built around procedural personas.

### 13.1 Learning Utility Functions for New Personas

As touched upon in the discussion, a limitation in this thesis is the fact that all the personas used for player classification and modeling had their utility functions predefined by expert game designers. Human player were characterized by the extent to which the personas agreed with them, but no attempt was made to adapt the personas to match the players by way of adjusting their utility functions. Instead, the clones of Chapter 9 tried to learn directly from the human play-traces. Future work should focus on learning the decision making style of human players indirectly, by adjusting the utility functions of the personas until the agreement ratios between personas and individual players, or clusters of players, are maximized. This could allow for the generation of new personas, adapted to play-traces, that are intelligible by way of their distances to the original a priori defined personas. If all three agreement ratios are targeted simultaneously, while the utility functions of personas are directed at multiple affordances, this becomes a multi-parameter, multi-objective indirect learning problem. One approach for finding solutions to such problems is applying multi-objective evolution. Multi-objective evolution can find probable solutions along high-dimensional pareto-fronts and are an active research topic within game artificial intelligence (Schrum and Miikkulainen, 2008) and have also seen use for modeling decision making preferences in other domains (Marler and Arora, 2004). Applying these methods might allow us to create hybrid personas with utility functions shaped by play-traces as examples, but interpretable in relation to a priori defined personas.

A second approach to learning the utilities of players could be preference learning (Fürnkranz and Hüllermeier, 2005), useful both for defining the input to the personas and for generating output. In terms of input, a persona designer might find the relative ranks of different affordances obvious, but could have a hard time meaningfully assigning scalar values to the affordances. Here, describing the utility to players in terms of preferences would bypass this problem. In terms of output, preference learning could work for personas where decisions can be reduced to matters of preferences, i.e. where the desired model output is of the ordinal or class kind. When personas select between different options, and the scalar outputs of utility functions in any instance have no other meaning than sorting the different decision options, preferences provide the same information. Preference learning can deliver strong performance for supervised learning problems (Yannakakis et al., 2009), can be used with evolutionary methods (Martínez et al., 2010), and a large body of work addresses the integration of affective information into preference learning (Yannakakis, 2009).

Related to this discussion is the question of whether the implemented controllers are capable of sufficiently representing the human players' decision making. In the next section, we address this question in greater detail.

## 13.2 Alternative Agent Control Architectures

One of the main issues with the implementations of the procedural persona concept in the papers of this thesis, visited in the discussion, is that at least the evolved agents based on linear perceptrons seem to be incapable of fully learning player decision making styles, as documented in Chapter 9. Additionally, this architecture does not specifically implement two decision making steps, an analytic and a heuristic one, but represents the analytic part in the weights of the perceptron and the heuristic and bounded part of decision making in the amount of game state information that is considered for each decision. The Monte-Carlo Tree Search based personas of *MiniDungeons 2*, described in Chapter 11, improve on this by having a distinct analytic step and a distinct heuristic step, each of which can be controlled, tuned, and adapted individually. However, a number of other agent control architectures could be considered and should be tested in conjunction with the procedural persona concept, including expanded off-line training methods such as neuro-evolution with augmenting topologies (NEAT) (Stanley and Miikkulainen, 2002), ensemble methods combining several approaches like reinforcement learning and neural networks (Mnih et al., 2015), a revisiting of the A\* in state space approach used in Chapter 6, or even variations of bounded simple breadth-first search with end-state heuristic evaluation (Sturtevant, 2013).

### 13.2.1 Emphasizing Model-Free Aspects of Procedural Personas

The first two options, NEAT and ensemble machine learning methods, represent moving away from the model-driven emphasis of the procedural persona concept and moving closer to model-free personas, where more of the behavior is learned from interacting with the game rules in simulations and from observing human play traces. This could provide the benefit of a lower reliance on designer analysis of affordances, but could also lower the interpretability of the personas as reference processes in the game's state space. Some intelligibility might be recovered, however, by extracting rules (Krishnan et al., 1999a) from the learned neural networks or mapping them to decision trees (Krishnan et al., 1999b; Setiono and Leow, 1999).

NEAT has been shown to be an effective method for configuring agents in games and beyond to learn and perform a variety of behaviors (Stanley and Miikkulainen, 2002).

This provides reason to believe that the method would allow for a closer matching of personas to individual player play-traces. The method would also be flexible in the sense that it would, depending on how the controller is constructed, be usable as a direct action selector (similar to the approach taken with linear perceptrons in this thesis), an action evaluator, or a state evaluator choosing between the outcomes of search based methods. One downside of using a more complicated approach such as NEAT is that each individual persona would contain a significant amount of the player model information in the configuration of the controller neural network, rather than in an easily inspectable matrix of weights directed at the affordances of the game in question. That means that any greater reproductive performance would most likely come with the cost of lower intelligibility of the model. This would complicate the measurement of differences between personas using their utility functions, but would still allow for the characterization of their relative differences through simulation-based metrics like the action, tactical, and strategic agreement ratios.

Applying ensemble methods such as a combination of different reinforcement learning techniques, e.g. Q-learning and deep learning combined have been shown to perform well across multiple games (Mnih et al., 2015), would most likely bring many of the same benefits and costs as the application of NEAT. One argument in favor of applying such methods could be that once the (relatively) simple intelligibility of controlling agents through linear combinations of weights in utility matrices is left behind, we may as well apply any method that produces a high quality reproduction of player decision making styles. If we decide on pursuing a model-free approach and simply use the machine learned model as a black-box actor outputting the next preferred action, we should exploit this to the largest possible extent. We might accomplish this by simply relying on the observed behaviors of the personas, and their relative differences as measured by the agreement ratios, to understand their differences.

Both of the approaches suggested above have the advantage that they can learn indirectly from objectives given through utility functions, and as such satisfy the basic requirements of procedural personas, being able to learn from a utility matrix. This makes them suitable candidates for alternative agent controllers. However, they suffer from the fact that they must be trained through off-line simulation. Depending on the decision space of the game, this can be a resource and time consuming process and may be hard to combine with the iterative nature of game analytics, game design, or procedural content generation. An analyst, a designer or a procedural content generation system might not have the time to wait for agents with various utility functions to be retrained through extensive simulation.

### 13.2.2 Emphasizing Model-Based Aspects of Procedural Personas

A different approach could be to pursue agent control methods that are more model-driven by combining on-line search in state space with hand-crafted heuristics that provide estimates of attaining the various kinds of utility derived from the affordances of the game. This would mean putting further emphasis on the analysis of the affordances of the game in question and building accurate heuristics for each of them. For the Monte-Carlo Tree Search implementation shown in Chapter 11, this would most likely enable us to boost the performance of the controller significantly by injecting expert information into the roll-out phase through search-guiding heuristics.

Another relevant on-line-search based method, driven by expert knowledge, that could be explored would be the evolution of action sequences with a rolling-horizon, a method which recently has been shown to perform well for coarser-grained actions, which could be promising for decisions at the tactical and strategic levels (Perez et al., 2013).

Relatedly, the A\* in state-space approach used for playing the *Mario AI Benchmark* in Chapter 6 could relatively easily be ported to a game like *MiniDungeons 2* which has few affordances. Each affordance would have to be incorporated into the A\* heuristic in manner allowing for weighing the contribution of each affordance. Finally, this approach could be even further simplified by simply searching as far in the game tree as possible within a given computational budget and evaluating the value of the nodes at the end of the search using the same hand-crafted heuristics, weighed by the utility matrix of the persona.

While these on-line-search based methods would require a greater degree of design and configuration from expert knowledge they would, on the other hand, be immediately tunable by analysts or game designers and could easily be adapted to varying computational budgets.

The alternative agent control architectures suggested above represent plausible candidates for improving the performance of procedural personas for the test-beds presented in this thesis, and possibly other test-beds and games. The first are directed toward increasing the model-free aspects of method while the latter increase the model-based aspects. Depending on their application, either approach may be better. Future work should focus on implementing these and possibly other agent control architectures to explore their suitability for the suggested uses of procedural personas.

Regardless of the agent control architectures used, the player modeling aspects of the work presented in this thesis is limited by the fact that only in-game behavior is considered. In the next section we outline possible ways of relating decision making styles as modeled by procedural personas to other measures of individual differences.

### 13.3 Correspondence with Other Measures

As noted in Chapter 3, a large component of many kinds of player modeling is player profile information. In this thesis, player profile information is integrated indirectly in the personas, by identifying the personas that are the best matches for human players, and is learned directly in the form of clones. In both these cases, the only player profile information that is learned is in relation to decision making styles within the respective games. However, as mentioned in Chapters 3 and 12, other psychological measures exist that describe decision making styles or constructs that are known to be correlated with differences in decision making styles. These include decision making style questionnaires (Scott and Bruce, 1995), measures of general intelligence, personality measures (Bruin et al., 2007) and measures of life motives (Canossa, 2012). Establishing relations between decision making styles and external measures would support the validity of procedural personas as models of decision making. Future research should collect these kinds of measures from participants.

In the same vein, research on the emotional components of decision making suggest that objective measures such as affective signals collected from physiology could contain information indicative of decision making styles (Ahn, 2010), useful for identifying individual or group differences in players. Future research should try to integrate such objective measures in the experimental paradigms.

### 13.4 Future Implementations and Studies with Procedural Personas

Another aspect of the procedural persona method which is not explored fully in this thesis is the integration of personas in mixed-initiative content generation tools and fully automated procedural content generation systems. Even though Chapter 10 provides an initial investigation into this, much work remains to be done to outline the limits and possibilities of the approach. Future work should include studies with game designers using mixed-initiative content generation tools supported by the personas, monitoring

how they interact with the system and to which extent they use the information generated. This will provide further information about the usefulness and applicability of procedural personas.

A first step in this direction will be the implementation of personas in a level editing tool for *MiniDungeons 2*, analogous to the one that was implemented for *MiniDungeons 1*. In the same vein, experiments with human players making decisions in *MiniDungeons 2* would be a natural next step, since this game is prepared for telemetric crowd-sourced collection of play data. Future work will see *MiniDungeons 2* launched as a publicly available research game for collecting decision making data from the game. This version of the game will feature automatic generation of game levels, following the same principles that were applied in *MiniDungeons 1*, using procedural personas as critics evaluating the generated content.

Finally, the procedural persona method should be applied to other games than *MiniDungeons 1 and 2*, which were specifically designed to support the procedural persona research agenda. Ideally, future work should include the construction and testing of personas for commercial games of other genres and of higher complexity. This could provide an opportunity to test the notion of running personas not in the actual state spaces of these games, but in abstract representations of their decision spaces.

## 13.5 Chapter Summary

In this chapter we explored the immediate future work that could extend the work presented in this thesis. While this thesis presents the theoretical foundation and early empirical investigations into extending play personas into procedural personas, many unanswered questions remain. From the theoretical perspective we still need to investigate the limits of how well we can model player decision making styles, which agent control architectures are appropriate for procedural personas, and what the validity of the concept as a player modeling tool is. We also need to conduct further investigations into the potential value of the method in game content creation processes, whether interactive or automated. From the practical perspective, the test-bed games, in particular *MiniDungeons 2*, have potential to support a range of new experiments which have yet to be conducted. Additionally, the procedural persona concept should be applied to other games than the ones developed in this thesis.

## Chapter 14

# Conclusion

This thesis introduced the concept of procedural personas. Extending the concept of play personas (Tychsen and Canossa, 2008; Canossa and Drachen, 2009), procedural personas are operationalizations of the former, realized in generative agents, modeling player decision making styles through a combination of utility functions and automatically learned or manually defined heuristics. Procedural personas are an attempt to represent player decision making processes in games in abstract generative models that capture the essential components of human decision making in order to support game analysts and designers in understanding the possibility spaces of games.

The procedural persona concept was derived from combining psychological decision science with play persona theory, and was operationalized and realized into practical implementations through a number of agent control methods drawn from game artificial intelligence. Though exhaustively investigating the validity, generalizability, and applicability of the framework turned out to be beyond the scope of this thesis, the empirical results substantiate that the concept was viable and applicable for two specific test-bed games, and a significant amount of potential future work was identified.

The concept presented in this thesis represents an addition to the gallery of methods for player modeling in games and for controlling search-based procedural content generation processes found in the game artificial literature. Moreover, it represents an addition to the literature on agent based modeling of psychological processes, albeit for the very specific domain of decision making in games, and to the general game artificial intelligence literature on agent control methods.



## Appendix A

# Epistemological Challenges for Operationalism in Player Experience Research

### Reference:

Pedersen, Christoffer Holmgård and Georgios N. Yannakakis (2012). “Epistemological Challenges for Operationalism in Player Experience Research”. In: *Foundations of Digital Games, Play Experience Workshop - Reconciling PX Methods*.

In this abstract we argue that employing an operationalist epistemology can limit the potential of player experience research. This argument is drawn from modern psychometrics where a movement questioning fundamental assumptions of classical test theory has been gaining increasing attention and has been supplying increasing amounts of evidence in recent years. We argue that these criticisms should be observed and that this could have important questions to player experience research in terms of theory and methods.

In line with the psychometrics research, we propose the use of a realist latent variable epistemology that we believe provides a stronger vantage point for the quantitative investigation of player experience and allows for more powerful methods of analysis. The concern that we put forward in this paper takes its outset in recent psychometric work of Borsboom (2005) and Borsboom et al. (2004). Their position is that the part of psychological research that employs classical test theory suffers from a number of problems. The problems span across the epistemological, theoretical and methodological levels, but are interconnected and are ultimately related to the question of validity in empirical studies. They label the combined classical test theory approach ‘operationalism’.

The fundamental problem of operationalism, according to Borsboom, stems from its notion of true and error scores. The idea is borrowed from the theory of errors which is used e.g. in astronomy and, crucially, assumes that the true score is constant across measurements and that the error score is only introduced by the limitations of psychological instruments of measurement, typically tests or questionnaires, or random aspects of the measurement situation. In classical test theory, the true score is approximated by assuming that the error score is random. Then one proceeds by using various techniques to measure the same phenomenon multiple times in different ways across individuals, while assuming that these measurements are perfectly parallel and governed by the same conditions. Given this procedure, the true score emerges as the average value across the multiple measurements; however, the assumptions underlying this approach are problematic, since human beings and measurement conditions will change in myriad ways between trials. This means that repeated measures cannot be considered perfectly parallel and from a strictly epistemological view, classical test theory cannot concern itself with repeated measures since they do not conform to the assumptions of the theory (Borsboom, 2005). Classical test theory's solution to this problem is, however, to assume this parallelism anyway. The end result of this assumption is that the measurable true score itself becomes an assumption, rendering the method tautologous, but this is seldom recognized in the application of classical test theory. If accepted, this critique of operationalism poses major questions to most user experience assessment tools such as the Game Experience Questionnaire (Ijsselstein et al., 2008) that uses the same approach across games and across subjects, not taking the measurement situation into account.

Recent research on ranking-based questionnaire schemes and non-linear predictive statistics (Yannakakis and Hallam, 2011; Yannakakis et al., 2010) (among many) indicates that approaches alternative to the operationalist one in player experience research can provide models that have a strong grounding in reality and better predictive capacity than e.g. measurements based on rating-based questionnaire methodology and linear frequentist statistics. We argue that these strains of epistemology and methodology can lead to new unexplored areas of player experience research and yield more accurate predictors of player experience.

## Appendix B

# Ethical Considerations in Designing Adaptive Persuasive Games

### Reference:

Pedersen, Christoffer Holmgård, Rilla Khaled, and Georgios N. Yannakakis (2012). “Ethical Considerations in Designing Adaptive Persuasive Games”. In: *Persuasive Technology*.

### B.1 Abstract

In this poster, we describe an ongoing project concerning the development of an Adaptive Treatment Game (ATG) for treating Post Traumatic Stress Disorder. The ATG uses biofeedback and computer game technology to enable multiple treatment techniques and goals. We examine how a multidisciplinary approach shaped the prototype and we discuss the ethical implications of creating a self-adaptive, semi-autonomous treatment game.

### B.2 Introduction

Post Traumatic Stress Disorder (PTSD) can be a severely disabling syndrome. It is sometimes developed after exposure to extreme stress in situations that include experiencing or witnessing mortal danger or extreme terror. Research into the efficacy of

different treatments for PTSD has been ongoing since the 1980's and a variety of treatment approaches have been identified (Foa et al., 2009; Nemeroff et al., 2006). One of the most recent developments in treatment approaches is the use of Virtual Reality Therapy (VR-T). Studies of the efficacy of VR-T are cautiously positive, though more research is needed (Parsons and Rizzo, 2008).

Meanwhile, advances in affective computing have enabled the creation of systems that use psychophysiological and behavioral data to reliably infer emotions experienced by users, including stress and anxiety (Haag et al., 2004; Picard, 2000; Popović et al., 2009). Drawing together threads of earlier research initiatives, we have reason to believe that including ludic and diegetic aspects in VR-T universes will enhance their efficacy, along with their ability to promote attitude and behavior change. To explore this hypothesis, we are developing a prototype of a multi mode Adaptive Treatment Game (ATG) that brings together three Cognitive Behavioral Treatment techniques in one coherent game universe. The ATG prototype will be completed and undergo clinical testing in Spring 2012.

### **B.3 The ATG prototype.**

The multidisciplinary team behind the ATG included multiple game designers and developers, computer game, affective computing and artificial intelligence researchers and three PTSD therapists (two psychologists and a psychiatrist) with decades of treatment experience between them. Based on the recommendations and experience of the therapists, Relaxation Training (RT), Stress Inoculation Training (SIT) and Exposure Therapy (ET) were chosen as the treatment approaches at the outset of the project. As such, our tool is multi modal, in that it supports these three treatment types. Avenues of adaptive persuasive design that were outlined by Fogg (2003) almost a decade ago have now been used in a plethora of tools and products as discussed by Kaptein et al. (2009) and Kaptein and Eckles (2010). Drawing on persuasive design strategies, including *tunneling*, *tailoring*, and *conditioning* (Fogg, 2003), we designed a treatment tool that uses adaptive biofeedback technology to learn an individual patient's response patterns and adjust the presented stimuli relative to reaction data from previous treatment sessions (Popović et al., 2009). In addition, the tool uses game design to create a convincing, seamless world. The three modes of the ATG are displayed in Figure B.1.

We decided to create our own development method in order to support the multidisciplinary collaboration process and structure the contributions from the different areas of expertise. Since we wanted to create a game that could be used in real world psychological practice, we needed to ensure that the ATG was feasible, useful and safe outside the

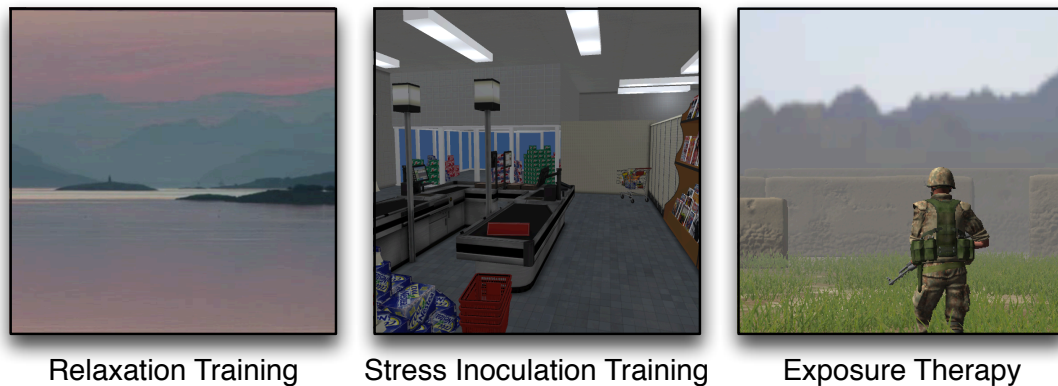


FIGURE B.1: Screenshots from the three modes of ATG

laboratory. To solve this task, we started by forming a hierarchy of design concerns, in the following priority: *functional design*, *treatment design*, *technology design*, and *game design*. This *design hierarchy* was used to resolve any design conflicts - e.g. treatment design concerns would always take precedence over game design concerns.

## B.4 Discussion

A design incorporating input from many sources of reference must become an amalgam of priorities from all the different fields, which are not necessarily compatible. This means that hard decisions and prioritization was necessary in order to make the different constituents of the ATG fit together.

It resulted in an underdeveloped game design, since this was at the lowest tier of the design hierarchy. It might have been fruitful to give game design a higher priority, or to abandon the idea of prioritized concerns altogether to ultimately make a more compelling tool.

However, we believe that the most interesting and pressing questions that the ATG raises, fall under the area of ethical persuasive design. Making any form of semi-autonomous system that interacts with patients in clinical settings entails a major ethical responsibility on the part of the designers of the system, as does the construction of any piece of persuasive technology. The responsibility of imbuing the system with these adaptive properties is not whisked away by providing the therapist as a safety measure; the constructors of the system still carry a responsibility for its subsequent effects on end users (Friedman and Kahn Jr, 2002). Berdichevsky and Neuenschwander (1999) describe in their decision tree for ethical evaluation of persuasive technologies that a system designer's work is ethical if her system's outcome is intended and good, but she

is not responsible if an undesirable outcome is unintended and not reasonably predictable. In the case of adaptive persuasive technology it becomes more difficult to imagine all possible use scenarios and thus all the possible unintended side-effects. This blurs the line of reasonable predictability as also Kaptein and Eckles (2010) point out in their treatment of persuasive profiles. Indeed, using adaptivity and profiling might put an even greater responsibility on the designer. In our case, we identified the following risks:

**Black-boxing** of the ATG's inner workings could make the links between experience and evaluation opaque to the patient and the therapist. This may in term result in alienation from the platform and demotivate the patient from engaging with the ATG more than once. The answer to this was exposing the evaluations of the system to the therapist as well as the patient, making the ATG a tool that the two use in an egalitarian and transparent manner.

**Objectification** of the patient to a level where the ATG's evaluations take precedence over phenomenological experience. A special responsibility lies with the therapist to emphasize the experience of the patient as valid.

**Erroneous profiling** where short-comings of the applied AI lead to misclassifications and possible misinterpretations of the patient's reactions to certain stimuli, potentially leading to the exposure of the patient to unduly stressful or completely inappropriate stimuli. This is handled by the fact that the therapist may always override the system.

**Second-order conditioning** where fear reactions to cues in the virtual environment are not extinguished, but rather generalized, making hitherto unproblematic elements of experience into cues eliciting stress and/or anxiety. This risk is handled in conjunction by the therapist and the ATG.

**Re-traumatization** could be considered the worst-case consequence of the combination of erroneous profiling and second-order conditioning. If the ATG presented a patient with a wrongly graded, too intense, stimulus, it could set off a fully fledged anxiety attack or a flashback. The consequence could be conditioning adverse responses to the therapy situation itself and have destructive consequences for the therapeutic alliance. To minimize this risk, the stimuli in the ATG undergo testing with expert therapists, users drawn from the general public and as well as veteran cohorts, and carefully selected PTSD patients.

## B.5 Conclusion and Future Work

With the ATG, we designed and built a prototype that points to a new way of applying virtual reality for PTSD in particular, but perhaps also cognitive behavioral therapy in general. While we have yet to investigate the efficacy of the ATG as a treatment tool (it will undergo clinical trials in Spring 2012) the process of making the prototype yielded a number of valuable insights.

Bringing a hierarchical set of concerns into an iterative design process turned out to be limiting. With this approach some areas of a project may receive too little attention or be inappropriately bounded by concerns with higher priority. This was partially the case with game design in our project and it remains an open question whether the ATG would be a better tool if game design had been allowed to influence functional or treatment design.

Our research and development efforts so far suggest that adaptive and goal-directed VR-T tools can make psychological therapy not only more engaging, but also more effective at treating debilitating anxiety disorders. It shows that making adaptive and profiling tools raises important ethical questions with responsibilities for the designers and creators – and that handling these challenges is worth the effort, when it allows us to make future cognitive behavioral therapy a more personal, immersive and effective experience.

## Appendix C

# The Games for Health Prototype

### Reference:

Holmgård, Christoffer, Georgios N. Yannakakis, and Rilla Khaled (2013). “The Games for Health Prototype”. In: *Foundations of Digital Games, Research and Experimental Games Festival*, pp. 396–399.

### C.1 Abstract

In this paper we present a prototype developed to explore the application of game design and technology to the treatment of Post Traumatic Stress Disorder (PTSD). We describe the design process that led to the development of the prototype and the included aspects of game design and game technology, how the approach and the prototype differ from previous work in using virtual environments in the treatment of PTSD, and we outline the first clinical trials of the prototype.

Demo and video links:

<http://itu.dk/people/holmgard/gfh/gfh.html>

<http://www.gamesforhealth.dk>

### C.2 Introduction

Post Traumatic Stress Disorder (PTSD) is a psychiatric diagnosis describing an often severely disabling syndrome that is sometimes developed after being exposed to highly stressful situations. Veterans from military operations are a high-risk group for developing this syndrome (Thomsen et al., 2011). With the Games for Health project we set out



to investigate the usefulness of game design and technology to support the psychiatric treatment of PTSD with veteran soldiers from the currently ongoing Danish military engagement in the conflict in Afghanistan. One treatment approach for PTSD, favored because of strong evidence for its therapeutic efficacy, is the *cognitive behavioral therapy* technique of *exposure therapy*. In exposure therapy, the therapist confronts the patient with anxiety provoking stimuli in a controlled setting in order to extinguish reactions to said stimuli and/or allow the patient to reprocess the memories cued by the stimuli. Three common variations are the use of real life stimuli i.e. *in vivo*, representing stimuli via media i.e. *mediated*, or having the patient imagine the stress provoking situations and thus self-generate the stimuli i.e. *imaginal* (Foa et al., 2009; Nemeroff et al., 2006). With the help of a multidisciplinary team, we designed and implemented a novel game that expands upon the principles of exposure therapy with simple game mechanics and uses detection mechanisms to infer the user's responses to in-game events.

### C.3 Background

Prior research has demonstrated the usefulness of virtual environments for treating veterans' PTSD with *virtual reality therapy* (Parsons and Rizzo, 2008). The developed systems often focus on outfitting the therapist with a sand-box type environment for the patient to explore that the therapist manually configures during the therapeutic session. The interaction is centered on the conversation between the user and the therapist, and the user's primary role is to explore and perceive the virtual environment, rather than directly interact with it. The approach also typically seeks a high degree of verisimilitude and employs specialized equipment such as head mounted displays or custom-built interfaces like e.g. vibrating platforms. Finally, virtual reality therapy most often focuses on exposing the patient to the original stressful, traumatizing situation. Notable examples are the *Virtual Iraq* and *Virtual Afghanistan* applications that show promising results in clinical testing (Reger et al., 2011; Rizzo et al., 2009b; Rizzo et al., 2009a). No approaches for virtual reality therapy have yet, to the knowledge of the authors, combined exposure therapy with affective computing (Picard, 2000). Various physiological phenomena have been demonstrated to allow for the measurement of emotional states, finding use in everything from the measurement and modeling of players' reactions to veterans' stress levels (Perala, 2007; Popovic et al., 2005; Westland, 2011; Yannakakis and Hallam, 2008). Additionally, studies have shown that veterans suffering from PTSD exhibit response patterns significantly different from those of non patients. It has been suggested that these differences could be used to support diagnostic differentiation (Blechert et al., 2007).

## C.4 Game Design Considerations

The design of the Games for Health prototype was guided by the joint design efforts of a multidisciplinary team consisting of researchers and practitioners from the fields of psychotraumatology and digital games. We set out to address the PTSD condition in a manner that differed from the prior work outlined above in a number of ways:

**Ease of use** was an overarching design goal. We aimed to create a tool that would fit into most psychological or psychiatric clinical practices with a minimum of technical expertise needed from the mental health practitioner. Hence, it was decided to develop a tool that could run on any reasonably modern consumer computer equipment. To this end the prototype was developed using the game engine Unity (Unity Technologies, 2005). We did, however, deviate from this design goal to the extent that it was necessary to enable the physiological readings for the affective computing component of the prototype, but only employed consumer-grade psycho-physiological measurement equipment (*Wild Divine*. <http://www.wilddivine.com>).

**Exposure to everyday life situations** was central to the design of the prototype. Whereas existing solutions (Reger et al., 2011; Rizzo et al., 2009b; Rizzo et al., 2009a) focus on addressing the memory of the traumatizing situation, the mental health professionals on the design team stressed the potential value of being able to expose patients to mundane, but stressful situations. We hypothesized that this approach could help PTSD patients improve their functioning in everyday tasks with direct benefits to their quality of life as a form of *systematic desensitization* (Foa et al., 2009). The task of going shopping in a supermarket was quickly identified as a common situation that is severely challenging to many veterans suffering from PTSD. Supermarkets are highly stimulating environments with many social interactions and unpredictable auditory and visual experiences, which PTSD patients find stressful; some to the extent that they avoid going shopping or only do so with a helper present for emotional support. Consequently, we built our prototype to take place in a virtual supermarket and focused on reproducing the experience of the stressfulness of shopping in a supermarket. Since many veterans suffering from PTSD report re-experiencing memories of the originally traumatizing situation when cued by elements in the environment, we also included short *flashbacks*. These momentarily change the environment of the game to an Afghan theater of operations, before changing back into the supermarket. The two modes of the game are depicted in Figure C.1 and Figure C.2.

**Goal driven interaction** with the virtual environment was determined to be a priority area where game design could support the use of virtual environments for treating



FIGURE C.1: The supermarket. A man is walking rapidly and angrily down the aisle.

PTSD. The underlying idea was informed by research in presence and immersion (McMahon, 2003) suggesting that the user's involvement with the diegetic aspects of a virtual environment has a high impact on the level of immersion the user feels which we, in turn, hypothesized to influence the stressfulness of the experience. Additionally, adding a mission to the patient's interaction with the prototype would quite simply provide a reason for interacting with as much of the virtual environment as possible, allowing us to expose the patient to as many different stimuli as possible. Hence, we added the very basic mission of having to gather a number of items indicated on a shopping list, before proceeding to the register, standing in line and paying the store clerk before leaving. The patients were asked to complete this task within a set time frame and were shown a timer counting down while they were completing the mission.

**Tracking responses through affective computing** was included for a number of reasons. As noted above, earlier studies (McFall et al., 1990) have shown that patients suffering from PTSD have autonomic responses to stressful events that differ from those of control individuals. Consequently, the autonomic responses would allow us to investigate whether veterans did indeed respond differently to the prototype than a control group, which would indicate that the prototype had a specific relevance to the patients.



FIGURE C.2: A flashback scene presented immediately after the game state of Figure C.1. The angrily walking man bled into a vision of a man running towards the player.

Conversely, once a baseline dataset had been established, it could also potentially allow the prototype to function as a diagnostic support tool, by using autonomic responses to distinguish between patients and non-patients. Finally, the ability to track and evaluate patients' responses to individual events in the virtual environment could allow for the construction of individual models of stress reactivity, which in turn could allow for the tailoring of event-configurations at the individual level.

## C.5 Playing the Game

Below, we briefly describe how the game was played by the participants in the study. Since the game is specifically constructed to elicit stress responses with the player an experimenter and trained psychologist, capable of intervening, is present in the room with the player throughout the session to minimize the risk of subjecting the player to any excessive stress. Before initiating the first round of play, the severity of the patient's PTSD symptoms is assessed using a structured interview and a questionnaire:

The PTSD Module of the Structured Clinical Interview for the DSM (SCID) (First et al., 2002) and the PTSD Checklist (Weathers et al., 1993).

### C.5.1 The Game Environment

As noted above, we assume that the player's level of immersion into the game environment will influence the extent to which he responds to the stressors in the game. To support immersion, the simulation is presented from a first-person-perspective, inviting the player to identify himself onto the unseen avatar of the game. The player starts at the entrance of the supermarket, navigates through the supermarket collecting items on the list and concludes the mission by standing in line and paying at the cash register. The player can move freely around the supermarket and collects or activates goods or items by centering the view on them and clicking. In order to ensure that the player experiences as much of the supermarket as possible, the object of the game is to collect a number of goods presented on an on-screen shopping list within a given time frame. The items to collect and the remaining time are displayed on screen during the game. The goods are placed in locations that make it probable that the player will be exposed to all sections of the supermarket if he manages to collect all items. The supermarket environment includes a number of stressors that aim at eliciting stress in the player. These are designed around three typical symptoms of PTSD, namely agoraphobia, hyper-arousal/heightened startle response, and the re-experiencing of traumatic events upon cueing by an outside stimulus or general stress (Foa et al., 2009).

Stressors targeting agoraphobia include the following design elements: The layout of the supermarket is designed to include hidden angles and preventing the player from attaining a full overview of the location. An aisle is blocked by a shopping cart making it difficult for the player to pass. *Non-Player-Characters* (NPCs), adults as well as children, wander around the supermarket, sometimes blocking the way of the player. Two NPCs, engaged in conversation, will stop talking and stare at the player if he approaches. An NPC walks angrily down the aisle toward the player, expressing aggression through his body-language. A family of NPCs are engaged in a discussion, the father scolding the child aggressively. An NPC pockets goods from the shelves of the supermarket.

Stressors targeting hyper-arousal include a dog barking at the entrance to the supermarket and the sound of crashes and glass breaking suddenly playing at random locations in the supermarket.

Finally, stressors targeting re-experiencing are included in the form of three different flashbacks. The purpose of these is to elicit the feeling of recalling and re-experiencing a traumatic memory. Only one is shown per mission. In the first flashback the player is

walking on a foot patrol in a typical Afghan theater of operations. In the second flashback the player sees a man running directly toward him, possibly with hostile intentions. In the third flashback the player sees a fellow soldier hit by an explosion, clutching the remains of his leg.

The game features three different configurations of missions assumed to elicit stress to different degrees. The missions vary in terms of the number of items the player must collect within the time limit and the apparent threat in the presented flashback. We assume that increasing the number of items the player has to collect within the same time frame and increasing the degree of threatening content in the flashbacks will increase the stressfulness of the experience accordingly. The missions are played consecutively from the least stressful to the most stressful.

### C.5.2 Hardware and Setup

For continuous measurement of *skin conductance* (SC) and *blood volume pulse* (BVP) the IOM biofeedback device (*Wild Divine*. <http://www.wilddivine.com>) is used. The IOM biofeedback device samples these two signals at a rate of 300 Hz and downsamples them to 30 Hz in firmware before transmitting them to the recording computer (*Wild Divine Developer Support*. <http://www.wilddivinedev.com>). The device is attached to the distal phalanges of the little, ring, and middle fingers of the player's non-dominant hand. To ensure maximum exposure to the content, while still using typical consumer-grade hardware, the game is presented on a 25" LCD monitor placed roughly 35 cm from the face of the player. For providing auditive stimulation, while still allowing the player to communicate with the experimenter, supra-aural headphones are used to deliver the sounds of the game. The audio level is adjusted to be experienced subjectively as loud, but pleasant. Since frustration with the control scheme of the game might introduce unwanted variation into the results of the experiment (Yannakakis et al., 2010) the game is configured to use what we consider to be standard controls for first-person-perspective computer games which should be familiar to most players. The mouse, operated with the player's dominant hand, controls the perspective and the keyboard controls movement. In order to minimize the risk of movement artifacts in the physiological readings, participants operate the keyboard (W, A, S, D or arrow keys) with only the index finger of their non-dominant hand, keeping the other fingers still.

## C.6 Clinical Trials

Clinical trials were completed in December 2012. A total of 15 veterans suffering from PTSD and 20 comparable veterans, who were screened for but not diagnosed with PTSD, played the game three to six times over one to two sessions (three rounds per session), with a fourteen day break between sessions. The analysis of the collected data is currently ongoing, however, preliminary findings indicate that the prototype has a strong potential for stimulating stress in veterans suffering from PTSD, based on both qualitative data gathered from observation and interviewing, as well as subjective ratings of experienced stress levels collected from the participants. Preliminary analyses of physiological responses support this observation. Several patients commented that the experience of playing through the prototype made them aware what elements of the supermarket they found particularly stressful and that this provided them with insight into their everyday challenges of going shopping.

## C.7 Future Work

The first analysis of the collected data material is expected to be completed during January 2013. If the collected data supports the hypothesis that physiological responses to the prototype can be used to differentiate between those veterans who qualify for the PTSD diagnosis and those who do not, further testing is planned to expand the body of evidence. The next step for the prototype will be the inclusion of adaptive functionality, enabling it to identify the most potent stimuli at the personal level. The assumption is that this will allow sessions to be increasingly customized to the individual patient, based on the individual's previous responses to the game.

## C.8 Conclusion

This paper has given a short introduction to the Games for Health prototype and the key considerations that went into its design and development. The developed prototype advances the areas of game design and technology applied for health purposes. Preliminary results indicate that there is indeed a value in this alternative approach of simulating everyday situations to provide PTSD patients with a tool that allows them to train their coping skills for these situations in collaboration with their therapist. Furthermore, preliminary analyses suggest that the bodily responses of PTSD patients are

substantially different from the responses of non-patients. This in turn makes it plausible that a virtual environment featuring affective detection functionalities could have potential as a future diagnostic tool supporting psychiatric practice.

## **C.9 Acknowledgments**

This research has been supported by the Danish Council for Technology and Innovation project *Games for Health* and by the EU FP7 ICT project *SIREN* (project no: 258453).



## Appendix D

# Stress Detection for PTSD via the StartleMart Game

### Reference:

Holmgård, Christoffer, Georgios N. Yannakakis, Karen-Inge Karstoft, and Henrik Steen Andersen (2013). “Stress Detection for PTSD via the StartleMart Game”. In: *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*. IEEE, pp. 523–528.

### D.1 Abstract

Computer games have recently shown promise as a diagnostic and treatment tool for psychiatric rehabilitation. This paper examines the positive impact of affect detection and advanced game technology on the treatment of mental diagnoses such as Post Traumatic Stress Disorder (PTSD). For that purpose, we couple game design and game technology with stress detection for the automatic profiling and the personalized treatment of PTSD via game-based *exposure therapy* and *stress inoculation training*. The PTSD treatment game we designed forces the player to go through various stressful experiences while a stress detection mechanism profiles the severity and type of PTSD via skin conductance responses to those in-game stress elicitors. The initial study and analysis of 14 PTSD-diagnosed veteran soldiers presented in this paper reveals clear correspondence between diagnostic standard measures of PTSD severity and skin conductance responses. Significant correlations between physiological responses and subjective evaluations of the stressfulness of experiences, represented as pairwise preferences, are also found. We conclude that this supports the use of the simulation as a relevant treatment tool for

stress inoculation training. This points to future avenues of research toward discerning between degrees and types of PTSD using game-based diagnostic and treatment tools.

## D.2 Introduction

Post Traumatic Stress Disorder (PTSD) is a psychiatric diagnosis describing an often severely disabling syndrome that is sometimes developed after being exposed to highly stressful situations. Veterans from military operations are a high-risk group for developing this syndrome (Hoge et al., 2004). A number of psychiatric treatments for PTSD are based on cognitive-behavioral approaches and include *exposure therapy* and *stress inoculation training*. Among the possible ways of treating PTSD computer games and virtual environments appear to have a great potential for eliciting stress in a controlled fashion and provide an immersive medium for PTSD treatment facilitating exposure therapy and stress inoculation training.

In this paper we investigate the usefulness of game design incorporating affect detection to support the psychiatric treatment of PTSD-diagnosed veteran soldiers. For that purpose, we designed and developed a game — *StartleMart* — that expands upon existing principles of PTSD treatment techniques with game mechanics and uses stress detection to infer the user’s physiological responses to in-game events. In this initial study we run a test with 14 veterans diagnosed with PTSD and examine the impact of their PTSD psychiatric profile on the arousal responses — measured via *skin conductance* (SC) — they manifest through in-game stress elicitors. In addition we examine the relationship between self-reported stress levels and SC signal features. Results show that physiological responses correlate with both PTSD profile features and self-reports of stress, supporting the relevance of the StartleMart game for PTSD diagnosis and treatment.

This work is novel as it uniquely combines real-time stress detection with a game (virtual) environment aimed at PTSD treatment. Diverging from and innovating upon earlier work in the use of simulations for treating PTSD (Wood et al., 2011), we argue that simulating everyday-life situations can help PTSD patients improve their functioning in everyday tasks with direct benefits to their quality of life as a form of *stress inoculation training* (Foa et al., 2009). The present study expands on previous research and approaches by constructing a desensitization and exposure paradigm consisting of a virtual world taking place in a home-like setting with integrated game mechanics. The result is a hitherto unexplored midpoint between *mediated* and *in vivo* exposure paradigms aimed at addressing issues in the everyday-life of the patient. We believe that by interweaving appropriate game design and efficient stress profiling we can provide a

personalized therapeutic environment that allows therapists, for the first time, to detect and address common PTSD symptoms across individuals with varying etiologies behind their PTSD. For instance, a veteran soldier and an assault victim may exhibit similar responses to stressful everyday-life situations and a simulation addressing these situations would be relevant to both.

### **D.3 Background: Stress Detection and PTSD Treatment**

This section covers related work on affect detection, the relationship between stress disorders and physiology, treatment types for PTSD and our approach to PTSD diagnosis and treatment, and, finally, the use of virtual environments and computer games for the treatment of PTSD.

#### **D.3.1 Stress Detection**

A wide range of approaches exist for capturing stress using physiological, behavioral, and self-report data or combinations thereof. Earlier work on stress detection (Calvo and D’Mello, 2010) has demonstrated how features extracted from raw physiological signals can be used to discern between a variety of emotional states in general (Picard et al., 2001) and in games (Martínez et al., 2011). SC has been identified as a useful indicator of stress elicited from tasks (Healey and Picard, 2005; Hernandez et al., 2011) and with soldiers (Perala, 2007). Innervation of the sweat glands is caused solely by the autonomic nervous system, making it a well suited source for measuring specifically arousal and, by extension, stress (Boucsein, 2011). Thus, SC is an obvious physiological indicator of player stress. It has been indicated that it is necessary to mediate the interpretation of physiological measurement data with information from self reports of phenomenal experience (Tognetti et al., 2011). For this reason we also use self-reports to collect information about the stressfulness of using StartleMart. Because self-reports have been shown to be unstable over time and hard to anchor to fixed scales between sessions (Yannakakis and Hallam, 2011), we treat self-reports as expressions of preference rather than directly comparable indications of subjective experience. By collecting physiological responses synchronously with log data from the game environment, we build on previous attempts to use affective computing to link responses to presented stimuli with PTSD symptom severity (Popović et al., 2009; Wood et al., 2011).

### D.3.2 Physiology of PTSD

In mediated stimulus exposure paradigms, PTSD-patients exhibit physiological responses to stressful visual and auditive stimuli that are significantly different from the responses of non-patients (Perala, 2007). Their responses are generally characterized by high sympathetic activity as measured by SC. In experimental studies, slower SC habituation, elevated resting SC, and greater SC responses to startling stimuli, have been found to be robust identifying characteristics of PTSD-patients (Pole, 2007). This indicates the higher base levels of arousal and heightened sensitivity to stress that are typical of the disorder. It has been suggested that these differences could be used to support diagnostic differentiation between PTSD patients and non-patients as well as between different degrees of PTSD symptom severity (Blechert et al., 2007) guiding treatment strategies or allowing for adaptive treatment tools (Wood et al., 2011). In the present study we innovate by investigating the relationship between PTSD profiles, self-reports of stress and SC signal features of arousal.

### D.3.3 Treating Stress Disorders

Two well-known treatment approaches for PTSD are the *cognitive-behavioral therapy* techniques of *exposure therapy* and *stress inoculation training*. In exposure therapy, the therapist confronts the patient with anxiety provoking stimuli in a controlled setting in order to extinguish reactions to the stimuli and/or allow the patient to reprocess the memories cued by the stimuli. Three common variations are the use of real life stimuli i.e. *in vivo*, representing stimuli via media i.e. *mediated*, or having the patient imagine the stress provoking situations and thus self-generate the stimuli i.e. *imaginal* (Foa et al., 2009). In stress inoculation training, the therapist exposes the patient to stimuli and situations that are not directly linked to the original trauma of the patient, but that cause problematic anxiety responses that are difficult for the patient to cope with (Foa et al., 2009). In the present study we utilize StartleMart as a game facilitator of exposure therapy and stress inoculation training.

### D.3.4 Games for Mental Health

Games and game-like worlds have successfully been used as mental health interventions by appropriating commercial games (Holmes et al., 2009) and by developing specialized solutions (Hoque et al., 2009). Earlier research has demonstrated the usefulness of virtual environments for treating veterans' PTSD with *virtual reality therapy*, an extension of exposure therapy (Parsons and Rizzo, 2008; Wood et al., 2011). Virtual reality

therapy most often focuses on exposing the patient to the original stressful, traumatizing situation, in the vain of classic exposure therapy, rather than appropriating principles from stress inoculation training. Notable examples are the *Virtual Iraq* and *Virtual Afghanistan* applications that show promising results in clinical testing (Reger et al., 2011; Rizzo et al., 2009a). In the StartleMart game, instead, we adopt a hybrid approach coupling exposure therapy and stress inoculation training which is informed by real-time stress detection for personalized treatment.

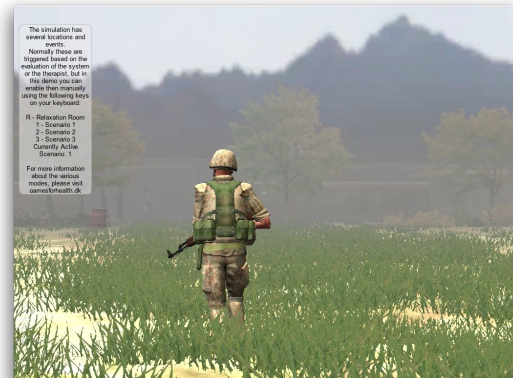
## D.4 The StartleMart Game for PTSD Treatment

The design of StartleMart is guided by the joint efforts of a multidisciplinary team consisting of researchers and practitioners from the fields of psychotraumatology, game design and game technology. Informed by the mental health professionals on the design team, we assume that being able to expose patients to simulations of mundane, but stressful, situations will be effective for diagnosis and treatment while maintaining flexibility in terms of possible patient groups that can benefit from the game. The task of going shopping in a supermarket is as a common situation that is severely challenging to many patients suffering from PTSD (Kashdan et al., 2010). Supermarkets are highly stimulating environments with social interactions and unpredictable auditive and visual experiences which PTSD patients find stressful; some to the extent that they avoid going shopping or only do so with a helper present for emotional support. Consequently, the game is built to primarily take place in a virtual supermarket (see Fig. D.1). Since many veterans suffering from PTSD report re-experiencing memories of the originally traumatizing situation when cued by elements in the environment, the game includes short *flashbacks* (see Fig. D.1). These momentarily change the environment of the game to an Afghan theater of operations, before changing back into the supermarket, embedding elements of *exposure therapy* (Foa et al., 2009) in the *stress inoculation training* (Foa et al., 2009) approach<sup>1</sup>. We assume that the player's level of immersion into the game environment will influence the extent to which he responds to the stressors in the game (McMahan, 2003). To support immersion, the simulation is presented as a first-person-shopper. The player starts at the entrance of the supermarket, navigates through the supermarket collecting items on the list and concludes a mission by standing in line and paying at the cash register. The player can move freely around the supermarket and collect or activate goods or items. In order to ensure that the player experiences as much of the supermarket as possible, the objective of the game is to collect a number of goods presented on an on-screen shopping list within a given time frame. The items to

<sup>1</sup>As the flashbacks constitute a minor part of the content they can easily be omitted or exchanged for groups of patients that are not war veterans.



(A) Sound of ventilator blowing overhead.



(B) Sound of wind blowing.



(C) Man walking angrily toward player.



(D) Man running toward player.



(E) Man staring at player.



(F) Wounded soldier staring at player.

FIGURE D.1: The three flashbacks of the game (b, d, f) and the immediately preceding supermarket scenes (a, c, e). Elements of the supermarket bleed into the flashbacks, simulating re-experience.

collect and the remaining time are displayed on screen during the game (see top-right of the left-hand images in Fig. D.1). The goods are placed in locations that make it probable that the player will be exposed to all sections of the supermarket if he manages to collect all items.

#### D.4.1 Game Stressors

The supermarket environment includes a number of stressors that aim at eliciting stress in the player. These are designed around three typical symptoms of PTSD, namely *fear-avoidance behavior*, *hyper-arousal* (i.e. heightened startle response), and *re-experiencing* of traumatic events triggered by an outside stimulus or general stress (Foa et al., 2009). Stressors targeting *fear-avoidance behavior* include the following design elements: The layout of the supermarket is designed to include hidden angles and preventing the player from attaining a full overview of the location. An aisle is blocked by a shopping cart making it difficult for the player to pass. Non-Player-Characters (NPCs), adults as well as children, wander around the supermarket, sometimes blocking the way of the player. Two NPCs, engaged in conversation, will stop talking and stare at the player if he approaches. An NPC walks angrily down the aisle toward the player, expressing aggression through his body-language. A family of NPCs are engaged in a discussion, the father is scolding the child aggressively. An NPC pockets goods from the shelves of the supermarket. Stressors targeting *hyper-arousal* include a dog barking at the entrance to the supermarket and the sound of crashes and glass breaking suddenly playing at random locations in the supermarket. Stressors targeting *re-experiencing* are included in the form of three different flashbacks. Their purpose is to elicit the feeling of recalling and re-experiencing a traumatic memory. Only one is shown per mission. In the first flashback the player is walking on a foot patrol in a typical Afghan theater of operations (see Fig. D.1b). In the second flashback the player sees a man, apparently of Afghan origin, running directly toward him (see Fig. D.1d). In the third flashback the player sees a fellow soldier hit by an explosion, clutching the remains of his leg (see Fig. D.1f). The game features three different configurations of missions assumed to elicit stress to different degrees. The missions vary in terms of the number of items the player must collect and the apparent threat in the presented flashback. We assume that increasing the number of items the player has to collect and increasing the degree of threatening content in the flashbacks will increase the stressfulness of the experience. The missions are played consecutively going from the least stressful to the most stressful.

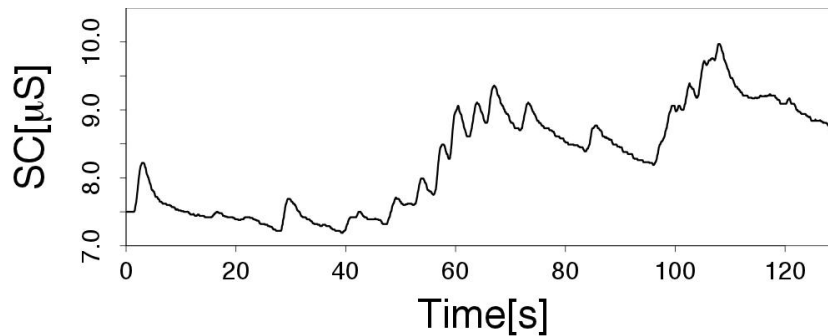


FIGURE D.2: An example of an SC signal recorded from a single game session: Subject no. 5, game session 3 (high stress intensity).

## D.5 Experimental Protocol and Data Collection

In this section we provide details about the participants of our experiment and the experimental protocol followed for the clinical trials of the game. Fourteen male PTSD patients, veterans from Danish military operations in Afghanistan, are included in the study presented in this paper. The participants are in psychiatric treatment for PTSD and qualify for the PTSD diagnosis. All subjects in the sample are medicated with *Selective Serotonin Re-uptake Inhibitors (SSRI)* which is known to generally lower sympathetic activity and in particular SC (Siepmann et al., 2003). This clearly adds a challenge to the detection of SC stress responses to game stimuli since patients are expected to manifest responses that are pharmacologically suppressed to an unknown degree. Our approach, thus, is based on a within-subject analysis to eliminate such effects. Each patient participates in the experiment twice, engaging in a total of 6 game play sessions, 3 per participation (11 patients have participated in both sessions, while 3 participated in the first session only). The experimenters, trained psychologists, welcome the participant, complete a diagnostic interview with the patient and collect various instances of demographic and background data from either the patient himself or the patient's medical records (see section D.6.1). The participant is introduced to the experimental setup and seated in front of the controls and monitor. The biofeedback device is attached to the participant's fingertips (see more details in section D.5.1), and a brief introduction to the game rules and how to control game is given. Following a short waiting period, collecting baseline SC data, the participant is asked to play three sessions of the game. Subjective data (self-reports) are collected over the course of the experiment (see section D.6.2). Finally, the experimenter debriefs the participant, responding to any concerns or issues the patient might have.



### D.5.1 Physiological Sensors and Setup

For continuous measurement of SC the IOM biofeedback device<sup>2</sup> is used. The IOM biofeedback device samples SC at a rate of 300 Hz and down-samples them to 30 Hz in firmware before transmitting them to the recording computer. The device's measuring electrodes are attached dryly to the distal phalanges of the little and middle fingers of the patient's non-dominant hand. A sensor measuring blood volume pulse is attached to the ring finger, but is not used for the study and analysis presented here. To ensure maximum exposure to the content the game is presented on a 25" LCD monitor placed roughly 35 cm from the face of the patient. To provide auditive stimulation, while still allowing the player to communicate with the experimenter, supra-aural headphones are used to deliver the sounds of the game. The audio level is adjusted to be experienced subjectively as loud, but pleasant. Since frustration with the control scheme of the game might introduce unwanted variation and artifacts to the results of the experiment (Yannakakis et al., 2010) the game is configured to use standard controls for first-person-perspective computer games which should be familiar to most patients. The mouse, operated with the patient's dominant hand, controls the perspective and the keyboard controls movement. To minimize the risk of movement artifacts in the physiological readings, patients operate the keyboard (W, A, S, D or arrow keys) with only the index finger of their non-dominant hand, keeping the other fingers still. An example of a SC signal collected from a single session is illustrated in Fig. D.2.

## D.6 User Data Features

This section details the three types of data obtained from, or extracted for, each experiment participant considered in this study. These include the PTSD profile of the patient, the subjective self-reports of stress during the experiment and the the set of features extracted from the SC signal.

### D.6.1 PTSD profile

Each participant is subjected to the PTSD Module of the Structured Clinical Interview for the DSM (SCID) (First et al., 2002) and completes the military version of the PTSD Checklist-IV (PCL-M) (Blanchard et al., 1996), a 17-item questionnaire that yields a PTSD symptom severity score in the interval 17–85. Then all patients are profiled in terms of *age*, PTSD checklist score *PCL*, number of deployments (i.e. war missions)

---

<sup>2</sup><http://www.wilddivine.com/>

TABLE D.1: PTSD profile features

Feature	Average	Standard deviation	Range
Age	26.8	2.5	22–32
PCL	58.0	4.9	50–65
$N_{dep}$	1.77	0.67	1–3
$N_{day}$	1001.2	432.4	113–1685

experienced  $N_{dep}$ , and the number of days since their return from their latest deployment  $N_{day}$ . The average, standard deviation and range values of the PTSD profile features across all 14 patients are presented in Table D.1. For the veteran PTSD patients, traumatized by experiences during deployment in this study, we assume that  $N_{day}$  may be considered an adequately precise measure of the time passed since the traumatizing experience. The deployment situation as a whole may be considered a highly stressful experience and as such part of the traumatizing situation. This means that the age of the trauma for all purposes here is assumed to be equivalent to  $N_{day}$ .

### D.6.2 Self-Reports of Stress

Before, immediately after, and following a short break after each of the three sessions, the patient is asked to provide a rating of his subjectively experienced stress level on the *Subjective Units of Distress Scale* (SUDS) (Wolpe, 1973) in a range from 0 to 100 with 0 representing complete absence of stress and 100 representing the most stressful experience the patient can recall.

### D.6.3 Features extracted from Skin Conductance

SC features are extracted from complete game sessions. Session data is procedurally and visually inspected for outliers and other indications of artifacts. Session data instances containing artifacts are either reconstructed, if possible, or removed from the data set. Following this data cleaning process — that removed 7 (9%) of all possible 75 game sessions resulting in a total of 68 (91%) sessions — all signals are adjusted for baseline readings, subtracting the individual session mean baseline value from the raw signal. Prior to feature extraction all signals are normalized via *min-max normalization* within individuals and across sessions from the same day. In order to account for any day-variation effects, signals from the same patients, but taken on different days, are treated as separate individuals. In accordance with recommendations from earlier studies on SC signal processing (Picard et al., 2001; Yannakakis and Hallam, 2008; Martínez et al., 2011), a number of features of the SC signal are extracted: Mean SC value  $SC_{\bar{x}}$ , standard deviation of the SC signal  $SC_{\sigma}$ , minimum SC value  $SC_{min}$ , maximum SC value  $SC_{max}$ ,

the difference between the maximum and minimum SC value  $SC_{max-min}$ , the correlation between recording time  $t$  and SC values  $R_{SCt}$ , the value of the first SC sample  $S_{init}$ , the value of the final SC sample  $S_{last}$ , the difference and absolute difference between final and first SC value  $SC_{last-init}$  and  $|SC_{last-init}|$ , the time of the minimum SC value  $t_{SC_{min}}$ , the time of the maximum SC value  $t_{SC_{max}}$ , the time  $t$  difference between the minimum and maximum SC values  $|t_{SC_{max-min}}|$  the means of the absolute values of the first and second differences of the SC signal  $SC_{|\delta_1|}$  and  $SC_{|\delta_2|}$ . An uncommonly used feature: the mean of the absolute first difference of the absolute first difference  $|SC_{\delta_\delta}|$  is added in an attempt to describe the tendency toward weak habituation in the signal.

## D.7 Results

We assume there is a relationship between PTSD profile and manifestations of stress on the SC signal and, thus, investigate the effects of PTSD profile features on SC signals. We also investigate the relationship between self-reported levels of stress and features of the SC signal. On that basis, we follow a correlation analysis for examining both relationships and present the key findings of this initial analysis.

### D.7.1 Correlation analysis between PTSD profile and SC features

To investigate the impact of a PTSD profile to manifestations of stress via SC we compute correlations between the two sets of features using Spearman's rank correlation coefficient  $\rho$  (Kendall, 1970). The results are presented in Table D.2. Results suggest that patients suffering from more severe degrees of PTSD (higher PCL values) respond with higher  $SC_{max}$  and a higher increase across the sessions as indicated by  $SC_{last-init}$ . This corresponds to findings that PTSD patients are more responsive to stressful stimuli. They also complete the session with a higher  $SC_{last}$  which corresponds to findings that PTSD patients are more responsive and habituate slower than non-patients. Patients with more severe PTSD exhibit higher values of all typical measures of local variation. The correlations between PCL and  $SC_{|\delta_1|}$ ,  $SC_{|\delta_2|}$  indicate that patients with more severe PTSD exhibit more variation. We hypothesize this is due to the relation between the severity of the syndrome and the hyper-responsiveness and hyper-arousal of the patient, meaning the patient responds more often to stimuli in the game.  $SC_{|\delta_\delta|}$  also correlates with symptom severity suggesting PTSD patients' slower habituation compared to non-patients (Pole, 2007). Significant positive correlation is observed between  $N_{dep}$  and  $SC_{min}$ . No clear explanation can be given for this, since more deployments should mean a higher degree of exposure to potentially highly stressful situations, but it should be noted that the range of the number of deployments in the sample is limited to 1 – 3.

TABLE D.2: Correlations  $\rho$  between SC signal and PTSD profile features in the left section of the table. Correlations  $c(z)$  between SC signal and self-reported stress in the right section of the table. Statistically significant correlations appear in bold (p-value < 0.05) and italics (p-value < 0.10).

	Age	PCL	$N_{dep}$	$N_{day}$	Day pairs	Adjac. pairs
$SC_{\bar{x}}$	0.10	0.10	0.01	0.08	<i>0.15</i>	<i>0.14</i>
$SC_{max}$	0.22	<b>0.29</b>	0.05	<i>-0.25</i>	<i>-0.15</i>	0.00
$SC_{min}$	-0.16	0.03	<b>-0.31</b>	0.05	<i>-0.15</i>	0.00
$SC_{max-min}$	0.23	<i>0.24</i>	0.13	<i>-0.26</i>	<b>-0.25</b>	<b>-0.19</b>
$SC_{\sigma}$	0.26	0.17	0.13	<i>-0.23</i>	<i>-0.15</i>	0.00
$R_{SCt}$	0.10	0.02	0.15	-0.06	<i>0.15</i>	<i>0.14</i>
$SC_{init}$	0.11	0.08	-0.13	0.10	<b>0.25</b>	0.10
$SC_{last}$	0.08	<b>0.35</b>	-0.17	<b>-0.30</b>	-0.02	-0.05
$SC_{last-init}$	-0.08	<b>0.31</b>	-0.03	<i>-0.25</i>	-0.02	0
$ SC_{last-init} $	0.09	<b>0.32</b>	-0.01	<b>-0.35</b>	-0.02	0.05
$t_{SC_{min}}$	0.06	0.02	-0.13	-0.02	<i>-0.12</i>	-0.10
$t_{SC_{max}}$	-0.17	0.06	0.10	-0.12	-0.02	0.00
$ t_{SC_{max-min}} $	-0.04	-0.07	0.11	-0.15	0.08	0.05
$SC_{ \delta_1 }$	0.15	<b>0.29</b>	0.13	<i>-0.26</i>	<i>-0.12</i>	<i>-0.14</i>
$SC_{ \delta_2 }$	0.15	<b>0.28</b>	0.14	<i>-0.25</i>	<i>-0.12</i>	<i>-0.14</i>
$SC_{ \delta_{\delta} }$	0.15	<b>0.28</b>	0.14	<i>-0.25</i>	-0.02	0.00

One could speculate that individuals who were only diagnosed with PTSD after several deployments were less susceptible to contracting the hyper-aroused state of PTSD. It would follow that they would exhibit lower SC bounds than their more susceptible colleagues, but the explanation remains speculation. A negative correlation is observed between  $N_{day}$  and the last SC value recorded in session; PTSD symptoms typically abate as a function of time (Foa et al., 2009), so this relation matches the literature on PTSD. The literature also matches the relation between  $N_{day}$  and  $PCL$ :  $PCL$  and  $N_{day}$  correlate negatively ( $\rho = -0.51, p < 0.01$ ) indicating the symptom severity decreases over time. It seems plausible that  $N_{day}$  is an inverse indicator of symptom severity and that less severe cases of PTSD exhibit lower bounds of SC, most likely due to a less elevated mean SC level and faster (closer to normal) habituation. Altogether, we argue the results indicate a positive relationship between symptom severity and physiological stress responses to StartleMart.

### D.7.2 Correlation analysis between self-reports and SC features

To investigate relations between self-reported levels of stress and signal features another measure of correlation is computed. As noted in section D.3.1 there is reason to believe that pair-wise preference analysis is a useful approach for examining self-reports. For this purpose, preference pairs are created with each pair containing a self-report value

and a feature value. Correlation values are calculated via the following test statistic (Yannakakis and Hallam, 2008)

$$c(z) = \sum_{i=1}^{N_s} \{z_i/N_s\} \quad (\text{D.1})$$

where  $z_i = 1$  if the highest stress value is from the same observation of the pair as the highest feature value; otherwise  $z_i = -1$ . In cases where the SUDS ratings are equal the stress preference pair is considered ambiguous and discarded. The relations between the self-reported SUDS ratings collected from the patients are expected to become increasingly vague over time. This, in turn, affects the quality of self-reported ratings. Episodic memory traces that form the basis of self reports fade over time, but the precise rate at which this memory decay occurs is unknown in this case and most likely individual (Robinson and Clore, 2002). Ideally, memory decay is so slow that the patient will have a clear feeling of the first session when rating the final session, but it is possible that only comparisons between immediately adjacent sessions are valid. To account for this uncertainty, two different correlation analyses are attempted: one combining data from sessions on the same day and one combining data only from sessions that took place immediately adjacent to each other. The results are included in Table D.2. Two significant effects are identified across the two approaches: A negative correlation between self-reports of stress and the range of the SC signal ( $SC_{max-min}$ ) and a positive correlation between reported stress and initial SC values. Both effects are consistent with the fact that patients with severe PTSD symptoms exhibit high SC values and weaker habituation. This means their SC values stay higher and their signals are subject to quick stabilization at the individually higher baseline. The correlations indicate that patients feeling stressed by interacting with StartleMart exhibit matching physiological responses and supports the relevance of the game to the target group.

## D.8 Discussion

The *PCL* score of the patients served as the first measure of ground truth describing symptom severity in this study. The *PCL* instrument is well-validated and the de facto standard for PTSD severity screening (Foa et al., 2009), but is nonetheless based on self-reports of personal experience retrieved from memory. This is an inherent weakness of the presented study, but one we suspect is innate and difficult to overcome in any study involving a syndrome defined partially by personal experience. The negative correlation between *PCL* values and  $N_{day}$ , which matches expectations according to the literature, strengthens the validity of the measure. The second measure of ground truth is the SUDS values collected during the game-play sessions. These are subject to the concerns

described in sections D.3.1 and D.7.2, but these concerns are sought mediated by the use of pair-wise preferences as the basis for correlation. In Table D.2 negative correlations are present between self-reports and  $SC_{max}$  and  $SC_{min}$  when pairs are constructed across all sessions in a day. These correlations trend in the opposite direction of what we would expect from theory. When pairs are limited to adjacent sessions these effects disappear and only effects matching expectations from theory remain. We consider this an indication of the psychometric properties of self-reports of stress. Future work using StartleMart might benefit from including stress evaluations as preferences at the report level. The analyses presented in this paper are limited to correlating features. Recent work in the literature (Zhai et al., 2005; Hernandez et al., 2011) describes how applications of non-linear techniques of analysis and machine learning can support stress detection and the data set described here could advantageously be analyzed by these methods in the future. Finally, the application of SC signal deconvolution could allow us to separate tonic and phasic components of the SC signal, identifying phasic drivers underlying responses to in-game events (Benedek and Kaernbach, 2010b). This could allow us to develop personalized, event-based PTSD profiles that integrate information from the simulation context into the stress detection process.

## D.9 Conclusion

The results of the analyses in this paper indicate that physiological responses to StartleMart are highly correlated with PTSD symptom severity and subjective experience expressed through self-reports of stress. That StartleMart elicits stress responses with PTSD patients lends credence to the general idea of using game-based stimuli of everyday life situations for stress inoculation training for PTSD patients. However, any treatment efficacy is unknown at this point and would require a randomized study. Nonetheless, the fact that physiological responses seem to scale with measures of symptom severity, self-reports and an indicator of recovery over time, indicates a promise to using StartleMart for diagnosis and treatment of PTSD. Future work will focus on leveraging these findings to refine profiling and adaptive game-based solutions supporting diagnosis and treatment in psychiatric work.

## Acknowledgments

This research was supported by the Danish Council for Technology and Innovation under the *Games for Health* project and by the FP7 ICT project *SIREN* (project no: 258453).

## Appendix E

# Multimodal PTSD Characterization via the StartleMart Game

### Reference:

Holmgård, Christoffer, Georgios N. Yannakakis, Héctor P. Martínez, Karen-Inge Karstoft, and Henrik Steen Andersen (2014). “Multimodal PTSD Characterization via the StartleMart Game”. In: *Journal on Multimodal User Interfaces*, pp. 1–13.

### E.1 Abstract

Computer games have recently shown promise as a diagnostic and treatment tool for psychiatric rehabilitation. This paper examines the potential of combining multiple modalities for detecting affective responses of patients interacting with a simulation built on game technology, aimed at the treatment of mental diagnoses such as Post Traumatic Stress Disorder (PTSD). For that purpose, we couple game design and game technology to create a game-based tool for *exposure therapy* and *stress inoculation training* that utilizes stress detection for the automatic profiling and potential personalization of PTSD treatments. The PTSD treatment game we designed forces the player to go through various stressful experiences while a stress detection mechanism profiles the severity and type of PTSD by analyzing the physiological responses to those in-game stress elicitors in two separate modalities: skin conductance (SC) and blood volume pulse (BVP). SC is often used to monitor stress as it is connected to the activation of the sympathetic nervous system (SNS). By including BVP into the model we introduce

information about para-sympathetic activation, which offers a more complete view of the psycho-physiological experience of the player; in addition, as BVP is also modulated by SNS, a multimodal model should be more robust to changes in each modality due to particular drugs or day-to-day bodily changes. Overall, the study and analysis of 14 PTSD-diagnosed veteran soldiers presented in this paper reveals correspondence between diagnostic standard measures of PTSD severity and SC and BVP responsiveness and feature combinations thereof. The study also reveals that these features are significantly correlated with subjective evaluations of the stressfulness of experiences, represented as pairwise preferences. More importantly, the results presented here demonstrate that using the modalities of skin conductance and blood volume pulse captures a more nuanced representation of player stress responses than using skin conductance alone. We conclude that the results support the use of the simulation as a relevant treatment tool for stress inoculation training, and suggest the feasibility of using such a tool to profile PTSD patients. The use of multiple modalities appears to be key for an accurate profiling, although further research and analysis are required to identify the most relevant physiological features for capturing user stress.

## E.2 Introduction

Post Traumatic Stress Disorder (PTSD) is a psychiatric diagnosis describing an often severely disabling syndrome that is sometimes developed after being exposed to highly stressful situations. Veterans from military operations are a high-risk group for developing this syndrome (Hoge et al., 2004). A number of psychiatric treatments for PTSD are based on cognitive-behavioral approaches and include *exposure therapy* and *stress inoculation training* (Foa et al., 2009). Among the possible ways of treating PTSD computer games and virtual environments appear to have a great potential for eliciting stress in a controlled fashion and provide an immersive medium for PTSD treatment facilitating exposure therapy and stress inoculation training. If enhanced with affect detection capabilities, these systems would be able to aid psychiatric evaluation of patients and automatic personalized treatments.

In this paper we investigate the combination of multiple modalities for stress detection in games designed to support the psychiatric treatment of PTSD-diagnosed veteran soldiers. For that purpose, we designed and developed a game — *StartleMart* — that expands upon existing principles of PTSD treatment techniques with game mechanics and profiles users based on their stress levels, which are inferred from physiological responses



to in-game events. In this study, we examine results gathered from 14 veterans diagnosed with PTSD and examine the relation among their PTSD psychiatric profile (measured via standard clinical tools), their perceived stress levels while playing the game (measured via post-experience self-reports), and their physiological responses to in-game stressors (measured via *skin conductance* (SC) and *blood volume pulse* (BVP) sensors). Results, building upon and expanding an initial analysis of SC features reported in previous work (Holmgård et al., 2013b), show that not only SC, but also BVP physiological responses correlate with both PTSD profile features and self-reports of stress. More importantly, results further show that features extracted from the two modalities can be combined into two underlying linear components which are related to measures of PTSD symptom severity. In all, the results demonstrate that capturing user stress responses from multiple physiological modalities enables a more nuanced understanding of patient responses compared to using a single modality. While one could argue that SC is enough to monitor stress because it is modulated only by the sympathetic nervous system (SNS) — which controls the responses of the body to events perceived as threats — the connection of BVP to both sympathetic and para-sympathetic nervous systems — which in contrast to SNS is linked to relaxation responses — has several advantages. The combination of both signals provides more complete information about the stress responses (e.g. the stress activation and the following relaxation or lack thereof) and more robust monitoring of SNS activations (e.g. motion artifacts, day-to-day changes or effects of drugs more prominent in a single modality (Siepmann et al., 2003)).

From the perspective of PTSD treatment, StartleMart represents a novel approach as it uniquely combines real-time stress detection with a game (virtual) environment simulating everyday-life situations. Diverging from and innovating upon earlier work in the use of simulations for treating PTSD (Wood et al., 2011), we argue that simulating everyday-life situations can help PTSD patients improve their functioning in everyday tasks with direct benefits to their quality of life as a form of *stress inoculation training* (Foa et al., 2009). The present game design expands on previous research and approaches by constructing a desensitization and exposure paradigm consisting of a virtual world representing a home-like setting with integrated game mechanics. The result is a hitherto unexplored midpoint between *mediated* and *in vivo* desensitization and exposure paradigms aimed at addressing issues in the everyday-life of the patient. Our experiments show the viability of this approach as both physiological responses and experience self-reports suggest that in-game events can significantly stress soldiers diagnosed with PTSD.

We believe that by interweaving appropriate game design and efficient stress profiling we can provide a personalized therapeutic environment that allows therapists, for the first time, to detect and address common PTSD symptoms across individuals with varying

etiologies behind their PTSD. For instance, a veteran soldier and an assault victim may exhibit similar responses to stressful everyday-life situations and a simulation addressing these situations would be relevant to both. Unsurprisingly, the utilization of multiple input modalities appears to be fundamental to empower these solutions with efficient profiling capabilities across individuals.

### **E.3 Stress Detection**

A wide range of approaches exist for capturing stress using physiological, behavioral, and self-report data or combinations thereof. Earlier work on stress detection (Calvo and D’Mello, 2010) has demonstrated how features extracted from raw physiological signals can be used to discern between a variety of emotional states in general (Picard et al., 2001) and in games (Martínez et al., 2011), and previous work has presented designs and studies that build affective loops for PTSD treatment by coupling presented stimuli with PTSD symptom severity (Popović et al., 2009; Wood et al., 2011). Informed by this previous research, our configuration captures indications of stress responses by continuously recording SC and BVP and by requesting self-reports from the player.

SC has been identified as a useful indicator of stress elicited from tasks (Healey and Picard, 2005; Hernandez et al., 2011) and with soldiers (Perala, 2007). Innervation of the sweat glands is caused solely by the sympathetic nervous system (SNS) whose activation is linked to reaction to threats (Andreassi, 2000). By extension, SC activity is related to emotional states such as fear, anger and anxiety, and more generally arousal (Boucsein, 2011). Thus, SC is an obvious physiological indicator of player stress.

BVP is a measure of blood flow in body appendices such as finger tips, and it is directly related to heart rate (HR). HR increases with activation of the SNS, but in contrast to SC, HR is also affected by a second control system; HR decreases with activation of the para-sympathetic nervous system (PSNS). This reaction is associated to states of rest and enjoyment (Andreassi, 2000). Thus, variability on HR as observed from BVP can reveal changes across both states of stress and relaxation, adding information not easily identified in the SC signal.

Self-reports can provide valuable ground truth (Tognetti et al., 2011) for interpreting recorded physiological responses, though they have been shown to be unstable over time and hard to anchor to fixed scales between sessions (Yannakakis and Hallam, 2011).

For our work presented here, we attempt to exploit SC to indicate sympathetic activation and HR to indicate para-sympathetic activation with self-report measures as a source of ground truth. In order to mediate the effect of the instability of self-reports, we

treat these as expressions of preference rather than direct indications of the subjectively experienced stressfulness.

### E.3.1 Physiology of PTSD

In mediated stimulus exposure paradigms, PTSD-patients exhibit physiological responses to stressful visual and auditory stimuli that are significantly different from the responses of non-patients (Perala, 2007). Their responses are generally characterized by high sympathetic activity as measured by SC and HR. In experimental studies, slower SC habituation, elevated resting SC, and greater SC responses to startling stimuli have been found to be robust identifying characteristics of PTSD-patients. Additionally, elevated resting HR and larger HR responses to startling stimuli and trauma cues have been identified as indicators of PTSD. Indeed, HR has been shown to prospectively predict PTSD in some studies (Pole, 2007). This indicates the higher base levels of arousal and heightened sensitivity to stress that are typical of the disorder. It has been suggested that these differences could be used to support diagnostic differentiation between PTSD patients and non-patients as well as between different degrees of PTSD symptom severity (Blechert et al., 2007) guiding treatment strategies or allowing for adaptive treatment tools (Wood et al., 2011). While prior work has related multiple modalities to PTSD, in the present study we contribute by investigating the relationship between PTSD profiles, self-reports of stress and SC, BVP and HR signal features in response to rich interactive simulations and determine that employing and combining multiple physiological modalities provides additional relevant information for characterizing patient responses, compared to using a single modality alone.

## E.4 The StartleMart Game for PTSD Treatment

Two well-known treatment approaches for PTSD — favored because of strong evidence for their therapeutic efficacy — are the *cognitive-behavioral therapy* techniques of *exposure therapy* and *stress inoculation training*. In exposure therapy, the therapist confronts the patient with anxiety provoking stimuli in a controlled setting in order to extinguish reactions to the stimuli and/or allow the patient to reprocess the memories cued by the stimuli. Three common variations are the use of real life stimuli i.e. *in vivo*, representing stimuli via media i.e. *mediated*, or having the patient imagine the stress provoking situations and thus self-generate the stimuli i.e. *imaginal* (Foa et al., 2009). In stress inoculation training, the therapist exposes the patient to stimuli and situations that are not directly linked to the original trauma of the patient, but that cause problematic anxiety responses that are difficult for the patient to cope with (Foa et al., 2009). In the

present study we utilize StartleMart as a game facilitator of exposure therapy and stress inoculation training. The game implements a simulation of a number of experiences from everyday life that are known to be stressful to PTSD patients (Kashdan et al., 2010), and additionally provides cues of traumatic experiences that war veterans may have experienced. The stimuli are designed around three typical symptoms of PTSD, namely *fear-avoidance behavior*, *hyper-arousal* (i.e. heightened startle response), and *re-experiencing* of traumatic events triggered by an outside stimulus or general stress (Foa et al., 2009). For a deeper discussion of related work in using simulations and games for mental health, and an in-depth presentation of the StartleMart game, we refer to our previous work (Holmgård et al., 2013b). Fig. E.1 gives examples of the types of stimuli delivered by the game.

## E.5 Experimental Protocol and Data Collection

In this section we provide details about the participants of our experiment and the experimental protocol followed for the clinical trials of the game. Fourteen male PTSD patients, veterans from Danish military operations in Afghanistan, are included in the study presented in this paper. The participants are in psychiatric treatment for PTSD and qualify for the PTSD diagnosis. All subjects in the sample are medicated with *Selective Serotonin Re-uptake Inhibitors (SSRI)* which is known to generally lower sympathetic activity and in particular SC (Siepmann et al., 2003), while recent research found no significant effect on HR variability (Kemp et al., 2010). This clearly adds a challenge to the detection of SC stress responses to game stimuli since patients are expected to manifest responses that are pharmacologically suppressed to an unknown degree. Each patient participates in the experiment twice, engaging in a total of 6 game play sessions, 3 per participation (11 patients have participated in both sessions, while 3 participated in the first session only). For each participation, the 3 sessions vary in terms of goal locations in the virtual environment and in terms of the specific configuration of the stressful experiences.

### E.5.1 Physiological Sensors and Setup

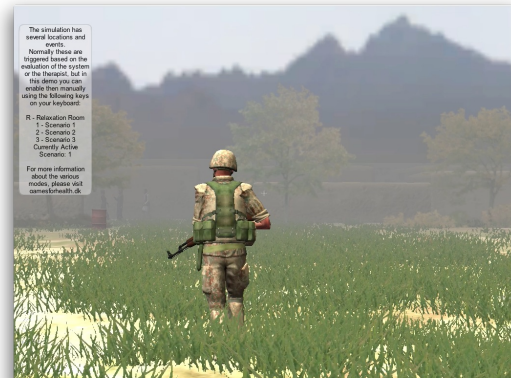
For continuous measurement of SC and BVP the IOM biofeedback device<sup>1</sup> is used. The IOM biofeedback device samples SC and BVP at a rate of 300 Hz and down-samples the signals to 30 Hz in firmware before transmitting them to the recording computer. An example of SC and BVP signals collected from a single session is illustrated in

---

<sup>1</sup><http://www.wilddivine.com/>



(A) Sound of ventilator blowing overhead.



(B) Sound of wind blowing.



(C) Man walking toward player.



(D) Man running toward player.



(E) Man staring at player.



(F) Wounded soldier staring at player.

FIGURE E.1: The three traumatic experience cues of the game (b, d, f) and the immediately preceding stressful scenes from everyday life (a, c, e). Elements of the everyday life scenes bleed into the cue scenes, referencing re-experience, a symptom typical for PTSD.

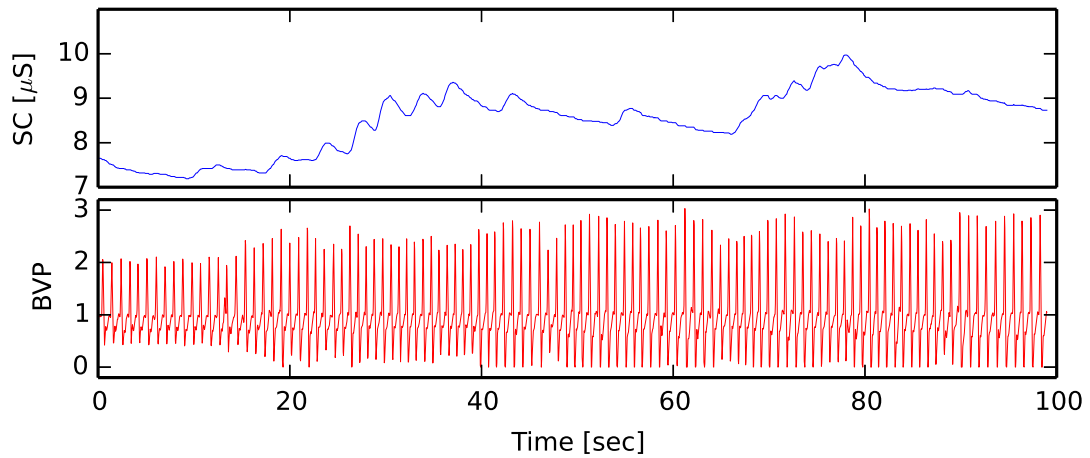


FIGURE E.2: An example of an SC and a BVP signal recorded from a single game session: Subject no. 5, game session 3.

Fig. E.2. The experimental paradigm and protocol are further detailed in our previous work (Holmgård et al., 2013b).

## E.6 User Data Features

This section details the three types of data obtained from, or extracted for, each experiment participant considered in this study. These include the PTSD profile of the patient, the subjective self-reports of stress during the experiment and the set of features extracted from the physiological signals.

### E.6.1 PTSD Profile

Each participant is subjected to the PTSD Module of the Structured Clinical Interview for the DSM (SCID) (First et al., 2002) and completes the military version of the PTSD Checklist-IV (PCL-M) (Blanchard et al., 1996), a 17-item questionnaire that yields a PTSD symptom severity score in the interval 17–85. Then all patients are profiled in terms of *age*, PTSD checklist score *PCL*, number of deployments (i.e. war missions) experienced  $N_{dep}$ , and the number of days since their return from their latest deployment  $N_{day}$ . The average, standard deviation and range values of the PTSD profile features across all 14 patients are presented in Table E.1. For the veteran PTSD patients, traumatized by experiences during deployment in this study, we assume that  $N_{day}$  may be considered an adequately precise measure of the time passed since the traumatizing experience. The deployment situation as a whole may be considered a highly stressful experience and as such part of the traumatizing situation. This means that the age of the trauma for all purposes here is assumed to be equivalent to  $N_{day}$ .

TABLE E.1: PTSD profile features

Feature	Average	Standard deviation	Range
Age	26.8	2.5	22–32
PCL	58.0	4.9	50–65
$N_{dep}$	1.77	0.67	1–3
$N_{day}$	1001.2	432.4	113–1685

### E.6.2 Self-Reports of Stress

Before, immediately after, and following a short break after each of the three sessions, the patient is asked to provide a rating of his subjectively experienced stress level on the *Subjective Units of Distress Scale* (SUDS) (Wolpe, 1973) in a range from 0 to 100 with 0 representing complete absence of stress and 100 representing the most stressful experience the patient can recall.

## E.7 Features Extracted from Physiological Signals

In the following section we present the features extracted from the two physiological signals and the motivation for including these signals. An overview of all features is presented in Table E.2.

### E.7.1 Features Extracted from Skin Conductance

SC features are extracted from complete game sessions. Session data is procedurally and visually inspected for outliers and other indications of artifacts. Session data instances containing artifacts are either reconstructed, if possible, or removed from the data set. Following this data cleaning process — that removed 7 (9%) of all possible 75 game sessions resulting in a total of 68 (91%) sessions — all signals are adjusted for baseline readings, subtracting the individual session mean baseline value from the raw signal. Prior to feature extraction all signals are normalized via *min-max normalization* within individuals and across sessions from the same day. In order to account for any day-variation effects, signals from the same patients, but taken on different days, are treated as separate individuals. In accordance with recommendations from earlier studies on SC signal processing (Picard et al., 2001; Yannakakis and Hallam, 2008; Martínez et al., 2011), a number of features that summarize the key statistical characteristics of SC signals are extracted: Mean SC value ( $SC_{\bar{x}}$ ), standard deviation of the SC signal ( $SC_{\sigma}$ ), minimum SC value ( $SC_{min}$ ), maximum SC value ( $SC_{max}$ ), the difference between the maximum and minimum SC value ( $SC_{range}$ ), the Pearson correlation between recording time ( $t$ ) and SC values ( $R_{SCt}$ ), the value of the first SC sample ( $SC_{\alpha}$ ), the value of the

final SC sample ( $SC_\omega$ ), the difference and absolute difference between final and first SC value ( $SC_{\omega-\alpha}$ ) and ( $|SC_{\omega-\alpha}|$ ), the time of the minimum SC value ( $t_{SC_{min}}$ ), the time of the maximum SC value ( $t_{SC_{max}}$ ), the absolute time ( $t$ ) difference between the minimum and maximum SC values ( $|t_{SC_{range}}|$ ), the means of the absolute values of the first and second differences of the SC signal ( $SC_{|\delta_1|}$ ) and ( $SC_{|\delta_2|}$ ). An uncommonly used feature, the mean of the absolute first difference of the absolute first difference ( $|SC_{\delta_\delta}|$ ), is added in an attempt to describe the tendency toward weak habituation in the signal.

### E.7.2 Features Extracted from Blood Volume Pulse

BVP features are also extracted from complete game sessions after inspection for artifacts. Only signals with uncorrupted corresponding SC signals are considered. None of these BVP signals were impacted by artifacts to a detrimental degree and hence BVP features are calculated for all 68 sessions remaining from the first inspection process (91%). Firstly, *heart rate* (HR) is computed using a 5-second sliding window by extrapolating the inter-beat time intervals detected in the BVP signal. The measurement unit for the resulting HR signal is beats per minute (BPM) whereas BVP is a relative measure of blood vessel pressure. Features from HR as well as BVP are chosen in order to cover the more significant BVP signal dynamics identified in previous studies in the field (Picard et al., 2001; Goldberger et al., 2001; Yannakakis and Hallam, 2008). Note that while HR and SC present similar features, BVP is a relative signal, and therefore extracted features focus only on its periodic nature. The RR features are aimed at providing an insight on the frequency domain of HR, and they have been developed over decades of research on psychophysiology (Goldberger et al., 2001).

For HR the following features are extracted: Mean ( $HR_{\bar{x}}$ ), maximum HR ( $HR_{max}$ ), minimum HR ( $HR_{min}$ ), range of HR ( $HR_{range}$ ) and standard deviation ( $HR_\sigma$ ). The Pearson correlation between measurement time and HR value ( $R_{HR_t}$ ), the HR at the start of the session ( $HR_\alpha$ ), at the end of the session ( $HR_\omega$ ), and the difference between the two ( $HR_{\omega-\alpha}$ ). The time of the maximum recorded HR value ( $t_{HR_{max}}$ ), the time of the minimum recorded HR value ( $t_{HR_{min}}$ ), and the difference in time between the two ( $t_{HR_{range}}$ ). The local variation of the HR signal as represented by the means of the absolute values of the first and second differences of the signal ( $HR_{|\delta_1|}$  and  $HR_{|\delta_2|}$ ).

For the raw BVP the following features are extracted: Mean ( $BVP_{\bar{x}}$ ), and standard deviation ( $BVP_\sigma$ ). The local variation of the BVP signal as represented by the means of the absolute values of the first and second differences of the signal ( $BVP_{|\delta_1|}$  and  $BVP_{|\delta_2|}$ ). The mean and standard deviation of the inter-beat amplitude ( $IBAm_{\bar{x}}$  and  $IBAm_\sigma$ ).



Additionally, given the inter-beat time intervals (RR intervals) of the BVP signal a number of heart rate variability extractors are proposed, concerned with the time-domain and the frequency domain, respectively:

- *HRV-time domain*: The mean and standard deviation of RR intervals ( $RR_{\bar{x}}$  and  $RR_{\sigma}$ ), the fraction of RR intervals that differ by more than 50 msec from the previous RR interval ( $pRR50$ ) and the root-mean-square of successive differences of RR intervals ( $RR_{RMS}$ ) (Goldberger et al., 2001).
- *HRV-frequency domain*: The frequency band energy values derived from power spectra obtained using the Lomb periodogram (Moody, 1993); energy values are computed as the integral of the power of each of the following two frequency bands, relevant for short experiences (Force, 1996): High Frequency ( $HF$ ) band: (0.15, 0.4] Hz and Low Frequency ( $LF$ ) band: (0.04, 0.15] Hz. In addition, the ratio  $\frac{LF}{HF}$  and the normalized values  $\frac{LF}{(LF+HF)}$  and  $\frac{HF}{(LF+HF)}$  are also included as recommended in (Force, 1996).

## E.8 Results

There exists a relation between the PTSD profile of a patient and the levels of stress that is experienced in everyday situations. Therefore we assume a relationship between the patient's PTSD profile and manifestations of stress on the physiological signals recorded across several sessions of interacting with StartleMart. First, we investigate this relation for each modality independently (see Section E.8.1 and Section E.8.3) using a correlation analysis between the PTSD profile feature set and the physiological features using Spearman's rank correlation coefficient  $\rho$  (Kendall, 1970). Secondly, we use principal component analysis (PCA) to also investigate the interdependencies between modalities by studying the relation between the principal components and the features (see Section E.8.5).

Furthermore, we investigate how physiological signals vary along different levels of stress experience during the game. On that basis, we study the correlation between self-reported stress levels and the extracted physiological features using a pair-wise correlation metric (see Section E.8.2 and Section E.8.4). As noted in Section E.3 there is reason to believe that pair-wise preference analysis is a useful approach for examining self-reports. For this purpose, we create two sets of preference pairs. The first set (denoted as *Day*) contains three preference pairs for each session by comparing the post-experience SUDS ratings given to each of the three games. The *preferred* game

TABLE E.2: Overview of features extracted from SC and BVP.

Symbol	Feature
$SC_{\bar{x}}$	Mean SC value
$SC_{max}$	Max SC value
$SC_{min}$	Min SC value
$SC_{range}$	Difference between max and min SC
$SC_{\sigma}$	Standard deviation of SC
$R_{SCt}$	Correlation, recording time and SC
$SC_{\alpha}$	Value of the first SC sample
$SC_{\omega}$	Value of the final SC sample
$SC_{\omega-\alpha}$	Difference between final and first SC
$ SC_{\omega-\alpha} $	Absolute difference between final and first SC
$t_{SC_{max}}$	Time of the max SC value
$t_{SC_{min}}$	Time of the min SC value
$ t_{SC_{range}} $	Time between the max and min recorded SC values
$SC_{ \delta_1 }$	Mean of absolute values of 1 <sup>st</sup> difference of SC
$SC_{ \delta_2 }$	Mean of absolute values of 2 <sup>nd</sup> difference of SC
$SC_{ \delta_\delta }$	Mean of absolute values of 1 <sup>st</sup> difference of 1 <sup>st</sup> difference of SC
$HR_{\bar{x}}$	Mean HR
$HR_{max}$	Max HR
$HR_{min}$	Min HR
$HR_{range}$	Range of HR
$HR_{\sigma}$	Standard deviation of HR
$R_{HRt}$	Correlation, measurement time and HR value
$HR_{\alpha}$	First HR value of the session
$HR_{\omega}$	Final HR of the session
$HR_{\omega-\alpha}$	Difference between the final and first HR values
$t_{HR_{max}}$	Time of the max recorded HR value
$t_{HR_{min}}$	Time of the min recorded HR value
$t_{HR_{range}}$	Time between max and min recorded HR values
$HR_{ \delta_1 }$	Mean of absolute values of 1 <sup>st</sup> difference of HR
$HR_{ \delta_2 }$	Mean of absolute values of 2 <sup>nd</sup> difference of HR
$BVP_{\bar{x}}$	Mean BVP
$BVP_{\sigma}$	Standard deviation of BVP
$BVP_{ \delta_1 }$	Mean of absolute values of 1 <sup>st</sup> difference of BVP
$BVP_{ \delta_2 }$	Mean of absolute values of 2 <sup>nd</sup> difference of BVP
$IBamp_{\bar{x}}$	Mean of inter-beat amplitude
$IBamp_{\sigma}$	Standard deviation of inter-beat amplitude
$RR_{\bar{x}}$	Mean of RR intervals
$RR_{\sigma}$	Standard deviation of RR intervals
$RR_{RMS}$	Root-mean-square of differences of RR intervals
$pRR50$	Fraction of RR intervals that differ by more than 50 msec from the previous RR interval
$HF$	Integral of power of Lomb periodogram High Frequency band: (0.15, 0.4] Hz
$LF$	Integral of power of Lomb periodogram Low Frequency band: (0.04, 0.15] Hz
$\frac{LF}{(LF+HF)}$	Normalized values of the Low Frequency band
$\frac{HF}{(LF+HF)}$	Normalized values of the High Frequency band
$\frac{LF}{HF}$	Ratio of Low Frequency over High Frequency band

on each pair corresponds to the highest rating (i.e. preference in this context denotes higher stress levels). In cases where the SUDS ratings are equal the stress preference pair is considered ambiguous and discarded. In the second set (denoted as *Adjacent*), we only extract two pairs from each session following the same procedure. We omit the comparison between the first and third game to minimize noise introduced by the variation on the rating scale due to memory decay. Note that the relations between the self-reported SUDS ratings collected from the patients are expected to become increasingly vague over time. This, in turn, affects the quality of self-reported ratings. Episodic memory traces that form the basis of self-reports fade over time, but the precise rate at which this memory decay occurs is unknown in this case and most likely individual (Robinson and Clore, 2002). Ideally, memory decay is so slow that the patient will have a clear feeling of the first session when rating the final session, but it is possible that only comparisons between immediately adjacent sessions are valid. To account for this uncertainty, we analyze the correlations for the *Day* and *Adjacent* sets independently. Correlation values are calculated for each physiological feature via the following test statistic (Yannakakis and Hallam, 2008)

$$c(z) = \sum_{i=1}^{N_s} \{z_i/N_s\} \quad (\text{E.1})$$

where for each pair  $i$ ,  $z_i = 1$  if the preferred game (i.e. higher stress report) presents a higher feature value, and  $z_i = -1$  otherwise;  $N_s$  represents the total number of pairs.

All the correlation coefficients discussed in the following sections are included in Table E.3.

### E.8.1 Correlations Between PTSD Profile and SC Features

Results suggest that patients suffering from more severe degrees of PTSD (higher PCL values) respond with higher  $SC_{max}$  and a higher increase across the sessions as indicated by  $SC_{\omega-\alpha}$ . This corresponds to findings that PTSD patients are more responsive to stressful stimuli. They also complete the session with a higher  $SC_{\omega}$  which corresponds to findings that PTSD patients are more responsive and habituate slower than non-patients. Patients with more severe PTSD exhibit higher values of all typical measures of local variation. The correlations between PCL and  $SC_{|\delta_1|}$ ,  $SC_{|\delta_2|}$  indicate that patients with more severe PTSD exhibit more variation. We hypothesize this is due to the relation between the severity of the syndrome and the hyper-responsiveness and hyper-arousal of the patient, meaning the patient responds more often to stimuli in the game.  $SC_{|\delta_\delta|}$  also correlates with symptom severity suggesting PTSD patients' slower

TABLE E.3: Correlations  $\rho$  between physiological signal features and PTSD profile features are in the left section of the table. Correlations  $c(z)$  between physiological signal features and self-reported stress are in the right section of the table. Statistically significant correlations appear in bold ( $p < 0.05$ ) and italics ( $p < 0.10$ ).

	Age	PCL	$N_{dep}$	$N_{day}$	Day	Adjac.
$SC_{\bar{x}}$	0.10	0.10	0.01	0.08	<i>0.15</i>	<i>0.14</i>
$SC_{max}$	0.22	<b>0.29</b>	0.05	<i>-0.25</i>	<i>-0.15</i>	0.00
$SC_{min}$	-0.16	0.03	<b>-0.31</b>	0.05	<i>-0.15</i>	0.00
$SC_{range}$	0.23	<i>0.24</i>	0.13	<i>-0.26</i>	<b>-0.25</b>	<b>-0.19</b>
$SC_{\sigma}$	0.26	0.17	0.13	<i>-0.23</i>	<i>-0.15</i>	0.00
$R_{SCt}$	0.10	0.02	0.15	-0.06	<i>0.15</i>	<i>0.14</i>
$SC_{\alpha}$	0.11	0.08	-0.13	0.10	<b>0.25</b>	0.10
$SC_{\omega}$	0.08	<b>0.35</b>	-0.17	<b>-0.30</b>	-0.02	-0.05
$SC_{\omega-\alpha}$	-0.08	<b>0.31</b>	-0.03	<i>-0.25</i>	-0.02	0.00
$ SC_{\omega-\alpha} $	0.09	<b>0.32</b>	-0.01	<b>-0.35</b>	-0.02	0.05
$tSC_{max}$	-0.17	0.06	0.10	-0.12	-0.02	0.00
$tSC_{min}$	0.06	0.02	-0.13	-0.02	<i>-0.12</i>	-0.10
$ tSC_{range} $	-0.04	-0.07	0.11	-0.15	0.08	0.05
$SC_{ \delta_1 }$	0.15	<b>0.29</b>	0.13	<i>-0.26</i>	<i>-0.12</i>	<i>-0.14</i>
$SC_{ \delta_2 }$	0.15	<b>0.28</b>	0.14	<i>-0.25</i>	<i>-0.12</i>	<i>-0.14</i>
$SC_{ \delta_{\bar{x}} }$	0.15	<b>0.28</b>	0.14	<i>-0.25</i>	-0.02	0.00
$HR_{\bar{x}}$	<b>-0.53</b>	0.18	-0.01	0.21	-0.08	-0.10
$HR_{max}$	-0.03	0.05	0.04	0.17	<b>0.25</b>	<i>0.19</i>
$HR_{min}$	-0.19	-0.04	-0.16	<i>0.23</i>	-0.46	-0.38
$HR_{range}$	0.09	0.07	0.21	0.00	<b>0.42</b>	<b>0.29</b>
$HR_{\sigma}$	0.09	0.18	<i>0.25</i>	-0.08	<b>0.32</b>	<b>0.29</b>
$R_{HRt}$	0.19	0.02	0.05	<b>-0.33</b>	-0.05	-0.05
$HR_{\alpha}$	-0.07	-0.05	0.07	<b>0.31</b>	-0.05	0.05
$HR_{\omega}$	<b>-0.44</b>	0.21	0.01	-0.07	<b>0.22</b>	0.14
$HR_{\omega-\alpha}$	-0.21	0.15	-0.07	<i>-0.25</i>	<i>0.12</i>	0.00
$tHR_{max}$	-0.08	-0.07	0.22	-0.07	-0.08	-0.10
$tHR_{min}$	<b>0.28</b>	<b>-0.36</b>	0.13	<i>0.23</i>	-0.25	-0.24
$tHR_{range}$	<i>-0.26</i>	0.22	0.07	-0.15	<b>0.19</b>	<i>0.24</i>
$HR_{ \delta_1 }$	-0.08	<b>0.39</b>	0.17	<i>-0.25</i>	<b>0.32</b>	<b>0.38</b>
$HR_{ \delta_2 }$	-0.11	<b>0.40</b>	0.19	<i>-0.24</i>	<b>0.32</b>	<b>0.33</b>
$BVP_{\bar{x}}$	-0.09	<b>0.28</b>	0.02	-0.16	<i>0.05</i>	0.00
$BVP_{\sigma}$	-0.04	<b>-0.29</b>	-0.02	<b>0.30</b>	-0.02	0.00
$BVP_{ \delta_1 }$	-0.12	<i>-0.24</i>	-0.06	<b>0.30</b>	-0.08	-0.05
$BVP_{ \delta_2 }$	-0.12	<i>-0.24</i>	-0.06	<b>0.30</b>	-0.08	-0.05
$IBAmp_{\bar{x}}$	-0.02	<b>-0.30</b>	-0.09	<b>0.30</b>	-0.19	-0.14
$IBAmp_{\sigma}$	0.19	-0.22	<i>0.25</i>	0.03	<b>0.25</b>	<b>0.29</b>
$RR_{\bar{x}}$	<b>0.51</b>	-0.16	0.02	-0.19	0.02	-0.05
$RR_{\sigma}$	0.02	<b>0.28</b>	0.21	<i>-0.25</i>	<b>0.22</b>	<b>0.29</b>
$RR_{RMS}$	-0.03	<b>0.31</b>	0.19	<b>-0.33</b>	<b>0.19</b>	<i>0.24</i>
$pRR50$	0.20	0.16	0.05	-0.18	<b>0.25</b>	<i>0.24</i>
$HF$	-0.04	0.13	0.22	<b>-0.42</b>	<i>0.12</i>	0.05
$LF$	0.17	<i>-0.23</i>	<i>-0.25</i>	<b>0.31</b>	-0.46	-0.38
$\frac{LF}{(LF+HF)}$	0.15	<i>-0.26</i>	<i>-0.24</i>	<b>0.45</b>	-0.22	-0.14
$\frac{HF}{(LF+HF)}$	-0.15	<i>0.26</i>	<i>0.24</i>	<b>-0.45</b>	<b>0.22</b>	0.14
$\frac{LF}{HF}$	0.15	<i>-0.26</i>	<i>-0.24</i>	<b>0.45</b>	-0.22	-0.14

habituation compared to non-patients (Pole, 2007). Significant positive correlation is observed between  $N_{dep}$  and  $SC_{min}$ . No clear explanation can be given for this, since more deployments should mean a higher degree of exposure to potentially highly stressful situations, but it should be noted that the range of the number of deployments in the sample is limited to 1 to 3. One could speculate that individuals who were only diagnosed with PTSD after several deployments were less susceptible to contracting the hyper-aroused state of PTSD. It would follow that they would exhibit lower SC bounds than their more susceptible colleagues, but the explanation remains speculation. A negative correlation is observed between  $N_{day}$  and the last SC value recorded in session; PTSD symptoms typically abate as a function of time (Foa et al., 2009), so this relation matches the literature on PTSD. The literature also matches the relation between  $N_{day}$  and PCL: PCL and  $N_{day}$  correlate negatively ( $\rho = -0.51, p < 0.01$ ) indicating the symptom severity decreases over time. It seems plausible that  $N_{day}$  is an inverse indicator of symptom severity and that less severe cases of PTSD exhibit lower bounds of SC, most likely due to a less elevated mean SC level and faster (closer to normal) habituation. Altogether, we argue the results indicate a positive relationship between symptom severity and features of SC responses to StartleMart.

### E.8.2 Correlations Between Self-Reports and SC Features

Two significant effects are identified across the two approaches to generating preferences pairs: A negative correlation between self-reports of stress and the range of the SC signal ( $SC_{range}$ ) and a positive correlation between reported stress and initial SC values. Both effects are consistent with the fact that patients with severe PTSD symptoms exhibit high SC values and weaker habituation. This means their SC values stay higher and their signals are subject to quick stabilization at the individually higher baseline. The correlations indicate that patients feeling stressed by interacting with StartleMart exhibit matching SC responses and supports the relevance of the game to the target group.

### E.8.3 Correlations between PTSD Profile and HR/BVP Features

A number of correlations are observed between the patients' PTSD profiles, and the BVP/HR features. Both average ( $HR_{\bar{x}}$ ) and last HR ( $HR_{\omega}$ ) are negatively correlated with age while no significant correlation is observed with respect to PCL, days from last deployment ( $N_{day}$ ) or days deployed ( $N_{dep}$ ). Age and PCL present an equivalent negative correlation ( $\rho = -0.52, p < 0.01$ ). This could indicate that in this sample older patients exhibit greater resilience toward PTSD as seen by lower PCL scores and lower

HR values; an interpretation which is consistent with findings in the literature on PTSD in veterans (Magruder et al., 2004).

More severe PTSD appears to result in a higher reactivity to the stressors as suggested by the positive correlation between PCL and a number of features that measure the local variability of the HR signal ( $HR_{|\delta_1|}$ ,  $HR_{|\delta_2|}$ ,  $RR_\sigma$  and  $RR_{RMS}$ ). Note that a higher value of these features is typically related to a larger number of *peaks* in the signal (quick increments on HR) that increase local variability while not necessarily affecting global variability (as measured by  $HR_\sigma$ ). This appears to be a strong relation as it has also been observed in the SC features. Due to the periodicity of BVP, its standard deviation ( $BVP_\sigma$ ) captures information of different nature, related more closely to the average inter-beat amplitude ( $IBAmp_{\bar{x}}$ ) than to the local variability of HR. PCL is negatively correlated to both  $BVP_\sigma$  and  $IBAmp_{\bar{x}}$  which suggests that more severe PTSD would be related to higher sympathetic arousal.

PCL is also correlated (negatively) to the time to the lowest recorded HR ( $t_{HR_{min}}$ ) suggesting that patients with more severe symptoms of PTSD respond earlier to the stressful stimuli and do not revert to a less stressed state during a session, though this feature, as noted, correlates with age as well.

The number of days from deployment ( $N_{days}$ ) appears to be positively correlated with a higher activity of the sympathetic nervous system (captured by  $LF$  and  $\frac{LF}{LF+HF}$  (Acharya et al., 2006), also  $IBAmp_{\bar{x}}$  and  $BVP_\sigma$ ), negatively correlated to higher activity of the para-sympathetic nervous system (captured by  $HF$  and  $\frac{HF}{LF+HF}$ ) and positively correlated with a dominance of sympathetic over para-sympathetic ( $LF/HF$ ). Given the connection between sympathetic activity and stress, these results show that the participants with older traumas appear to be more stressed during the therapy than patients with more recent traumas. These correlations to a certain extent run counter to the idea of spontaneous PTSD recovery over time, though one possible explanation could be that patients who are further into treatment respond with stronger manifestations of sympathetic dominance when subjected to novel therapeutic methods. On the other hand, a correlation is observed between  $RR_{RMS}$  and  $N_{day}$ . It would seem that this correlation matches the assumption that  $N_{day}$ , representing the age of the trauma, is a rough measure of spontaneous recovery leading to lower manifestation from patients with older traumas. Finally, mean inter-beat intervals ( $RR_{\bar{x}}$ ) correlate positively with age, mirroring the relation found between  $HR_{\bar{x}}$  and age.

#### E.8.4 Correlations between Self-Reports and HR/BVP Features

Similar patterns of significant effects are identified across the two approaches to generating preferences pairs for the HR/BVP signals. For HR features, measures connected with stress and sensitivity to stress exhibit positive correlations to the ranked subjective evaluation of session stressfulness. The same patterns are observed for the features extracted directly from the BVP signal. Again, as was the case for the SC signal, these correlations indicate both that patients feel stressed from interacting with StartleMart and that this experience scales with symptom severity. The stronger effect between the features derived from HR and self-reports than between the SC features and self-reports matches findings in the literature suggesting that HR features provide a robust physiological indicator of PTSD symptom severity (Pole, 2007).

#### E.8.5 Principal Component Analysis of Physiological Features and their Relations to PTSD symptom severity

While the results suggest that the applied modalities are useful in characterizing player stress responses in relation to interacting with StartleMart, the high number of features makes the identification of the underlying causes difficult. In order to investigate whether any unifying components exist which underlie the correlations in the large feature set, a principal component analysis with no rotation is conducted. All extracted features are subjected to the analysis producing components that combine information across modalities, producing an initial set of 45 principal components.

Initially, 45 principal components are generated. The first five components account for approximately 43%, 30%, 12%, 7%, and 4% of the variance, respectively, after which the proportion of explained variance for each component reduces rapidly. The components are depicted in a scree plot in Fig. E.3. In order to retain a low number of components, we coerce the model to produce two components, though three components could have been considered as well. However, choosing two components yields a balance between the variance explained by each of the two resulting components as each accounts for approximately half of the variance as indicated in Table E.4 ( $df = 901$ ,  $\chi^2 = 16166.24$ ,  $p < 0.01$ ).

TABLE E.4: Standard deviation and explained variance for each of the two principal components extracted from the feature set.

	Component 1	Component 2
Standard deviation	3.1841	2.8916
Proportion of Variance	0.5480	0.4520
Cumulative Proportion	0.5480	1.0000

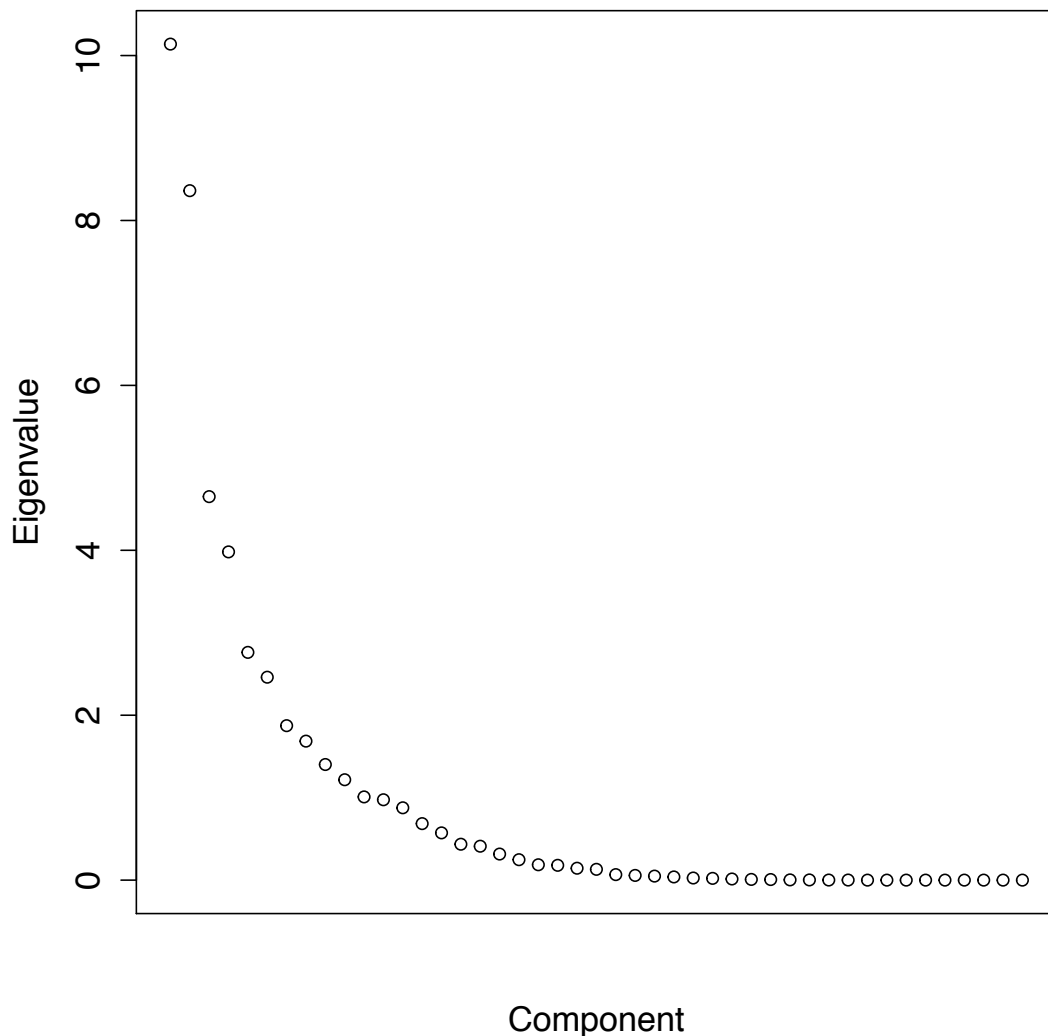


FIGURE E.3: Scree plot of all 45 principal components derived from the original feature set. The scree plot shows two components explaining most of the variance in the data with the remaining components explain relatively less.

Though the analysis does not provide any inherent labeling of the resulting components it could be hypothesized that the two components are related to para-sympathetic and sympathetic activation in response to the simulation, respectively, or put differently, the player's ability to habituate to the stimuli or respond to the stimuli with manifestations of stress.

To provide an insight into the most important features for each component, Table E.5 presents the Pearson correlations between the individual features and the principal components, ordered by the magnitude of the correlations. Component 1 is characterized primarily by negative correlations to measures of sympathetic activity captured via BVP



expressed in HR. As described in Section E.3.1, PTSD symptom severity is characterized by elevated resting HR and larger HR responses to stimuli and the component appears to capture this phenomenon. Component 2 is characterized primarily by positive correlations to features extracted from SC, indicating sympathetic activity, also in accordance with expectations from the literature. From the correlations between the individual features and the two components, it seems that the first component primarily represents responses captured via BVP, while the second component primarily represents responses captured via SC. However, the fact that both components do exhibit correlations to features from both modalities suggests that the two modalities together enable the capturing of both resilience and sensitivity to manifesting stress in response to the simulation. The principal components are subsequently correlated to Age, PCL,  $N_{day}$ , and  $N_{dep}$ , and self-reports as the external measures of symptom severity and tendency to manifest stress in response to the simulation.

To test the relation between these two principal components and the measures of PTSD symptom severity, the same correlation analyses applied to the individual features are applied to the components. The results are reported in Table E.6. A negative correlation between the first component and PCL and a borderline significant positive correlation between the second component and PCL are observed. Additionally, a positive correlation between the first component  $N_{day}$  is observed, while a negative correlation between the second component and  $N_{day}$  is evident. The two pairs of correlations conform to expectations from the literature on PTSD symptoms, as described earlier, and could be seen as further indication that the two components represent patients' tendencies toward para-sympathetic and sympathetic activation in response to the simulation. Notably, neither of the components correlate with Age or  $N_{dep}$ . Pairwise correlations between the extracted components and self-reports of experienced stress also exhibit to the same pattern, with component 1 correlating negatively with reports of stress and component 2 correlating positively with reports of stressful experience. Taken together, this could indicate a more robust relation between the multimodal components, symptom severity, and self-reports than between individual features and external measures of ground truth.

## E.9 Discussion

The PCL score of the patients served as the first measure of ground truth describing symptom severity in this study. The PCL instrument is well-validated and the de facto standard for PTSD severity screening (Foa et al., 2009), but is nonetheless based on self-reports of personal experience retrieved from memory. This is an inherent weakness of

TABLE E.5: Correlations between individual features and the two extracted principal components. For each component, the features listed are sorted according to the absolute value of the correlation coefficient in order to allow for a straightforward identification of the features with the strongest relation to the component.

Component 1		Component 2	
Feature	$R$	Feature	$R$
$RR_\sigma$	-0.83	$SC_\sigma$	0.78
$RR_{RMS}$	-0.83	$SC_{range}$	0.78
$HR_{ \delta_1 }$	-0.77	$ SC_{\omega-\alpha} $	0.78
$HR_{ \delta_2 }$	-0.76	$SC_{ \delta_1 }$	0.77
$LF$	0.74	$SC_{ \delta_2 }$	0.77
$\frac{HF}{(LF+HF)}$	-0.69	$SC_{ \delta_\delta }$	0.77
$\frac{LF}{(LF+HF)}$	0.69	$SC_{max}$	0.73
$HR_{range}$	-0.66	$SC_{min}$	-0.66
$\frac{LF}{HF}$	0.66	$HR_{min}$	-0.62
$HR_\sigma$	-0.63	$RR_{\bar{x}}$	0.52
$IBAmp_{\bar{x}}$	0.60	$HR_{\bar{x}}$	-0.50
$pRR50$	-0.53	$SC_{\omega-\alpha}$	0.48
$SC_{min}$	-0.53	$pRR50$	0.46
$SC_{range}$	0.52	$SC_\omega$	0.43
$HR_{min}$	0.50	$RR_\sigma$	0.42
$SC_{\bar{x}}$	-0.49	$\frac{LF}{(LF+HF)}$	-0.42
$BVP_{ \delta_1 }$	0.49	$\frac{HF}{(LF+HF)}$	0.42
$BVP_{ \delta_2 }$	0.49	$RR_{RMS}$	0.42
$ SC_{\omega-\alpha} $	0.48	$HF$	0.39
$HR_{max}$	-0.48	$HR_\omega$	-0.38
$SC_\sigma$	0.43	$HR_\alpha$	-0.37
$BVP_\sigma$	0.41	$\frac{LF}{HF}$	-0.37
$SC_{max}$	0.41	$SC_\alpha$	-0.35
$SC_{\omega-\alpha}$	0.39	$BVP_{ \delta_2 }$	-0.33
$SC_\alpha$	-0.39	$BVP_{ \delta_1 }$	-0.33
$ t_{SC_{range}} $	-0.35	$SC_{\bar{x}}$	-0.32
$t_{HR_{max}}$	0.35	$LF$	-0.32
$t_{HR_{range}}$	0.33	$IBAmp_{\bar{x}}$	-0.28
$R_{SCt}$	-0.33	$IBAmp_\sigma$	-0.26
$SC_\omega$	0.32	$HR_{range}$	0.25
$SC_{ \delta_1 }$	0.32	$HR_{ \delta_1 }$	0.23
$SC_{ \delta_2 }$	0.32	$HR_{ \delta_2 }$	0.22
$SC_{ \delta_\delta }$	0.32	$t_{SC_{max}}$	0.19
$HF$	-0.31	$t_{SC_{min}}$	0.16
$BVP_{\bar{x}}$	-0.30	$R_{SCt}$	0.14
$HR_{\bar{x}}$	-0.29	$HR_\sigma$	0.13
$RR_{\bar{x}}$	0.25	$t_{HR_{max}}$	0.12
$IBAmp_\sigma$	-0.24	$IBAmp_\sigma$	0.11
$HR_\alpha$	-0.23	$t_{HR_{range}}$	0.10
$t_{SC_{min}}$	0.15	$HR_{max}$	-0.10
$R_{HRt}$	0.14	$ t_{SC_{range}} $	0.09
$HR_{\omega-\alpha}$	0.14	$BVP_{\bar{x}}$	0.06
$t_{HR_{min}}$	-0.11	$HR_{\omega-\alpha}$	0.06
$HR_\omega$	-0.08	$R_{SCt}$	-0.05
$t_{SC_{max}}$	0.02	$t_{HR_{min}}$	-0.02

the presented study, but one we suspect is innate and difficult to overcome in any study involving a syndrome defined partially by personal experience. The negative correlation between PCL values and  $N_{day}$ , which matches expectations according to the literature, strengthens the validity of the measure. The second measure of ground truth is the SUDS values collected during the game-play sessions. These are subject to the concerns

TABLE E.6: Correlations  $\rho$  and  $c(z)$  between the two principal components and Age, PCL,  $N_{day}$ , and  $N_{dep}$ , and self-reports. Statistically significant correlations appear in bold ( $p < 0.05$ ) and italics ( $p < 0.10$ ).

	Component 1	Component 2
Age	0.10	0.18
p	0.47	0.39
<i>PCL</i>	<b>-0.29</b>	<i>0.26</i>
p	0.04	0.06
<i>N<sub>day</sub></i>	<b>0.30</b>	<b>-0.35</b>
p	0.03	0.02
<i>N<sub>dep</sub></i>	-0.22	0.21
p	0.22	0.22
Self-reports, day	<b>-0.25</b>	<b>0.22</b>
p	0.02	0.03
Self-reports, adjac.	<i>-0.14</i>	<i>0.19</i>
p	0.08	0.06

related to ratings (as described in Sections E.3 and E.8), but these concerns are sought mediated by the use of pair-wise preferences as the basis for the correlation analysis; this analysis ignores the exact value of the ratings and considers only the ordinal relation between ratings given on the same day or adjacent sessions. In Table E.3 negative correlations are present between self-reports and  $SC_{max}$  and  $SC_{min}$  when pairs are constructed across all sessions in a day. Based on findings in the literature, we would expect these to be positive. However, when pairs are limited to adjacent sessions these effects disappear and only effects matching expectations from theory remain. We consider this a confirmation that the absolute value of self-reported ratings of stress becomes increasingly unreliable over time as memories decay. Future work using StartleMart might benefit from including stress evaluations as preferences at the report level.

Some features extracted from BVP indicate dominance of sympathetic activation over para-sympathetic that scales with the age of the trauma, contrary to our expectation of spontaneous recovery. Though one explanation could be that veteran patients respond stronger to novel treatment methods, further investigation is necessary to fully understand these relations.

The feature combination through principal component analysis suggests that it may be feasible to reduce the physiological stress manifestations in response to the simulation to two underlying components which could be interpreted as resilience and sensitivity towards the stressful stimuli. These two components correlate with measures of symptom severity and self-reports as expected from the literature. However, as a cross-modality feature combination technique the principal component analysis seems to fall short, as one component is dominated by BVP/HR features, and the other component is dominated by SC features.

In general, the analyses presented in this paper are limited to correlating features and applying linear methods of feature combination through principal component analysis. Recent work in the literature (Zhai et al., 2005; Hernandez et al., 2011) describes how applications of non-linear techniques of analysis and machine learning can support stress detection and the data set described here could advantageously be analyzed by these methods in the future. Additionally, the application of SC signal deconvolution could allow us to separate tonic and phasic components of the SC signal, identifying phasic drivers underlying responses to in-game events (Benedek and Kaernbach, 2010b). This could allow us to develop personalized, event-based PTSD profiles that integrate information from the simulation context into the stress detection process. Finally, more advanced methods of multimodal signal fusion could enable a better characterization of the stress responses through the combination of the SC and BVP/HR signals, possibly yielding a more satisfactory cross-modality combination and a more accurate model of the patients' stress responses (Martinez et al., 2013).

## **E.10 Conclusion**

In this study we used StartleMart, a game-based PTSD exposure therapy and stress inoculation therapy tool, to elicit stress responses from 14 male PTSD patients. We collected physiological indications of stress responses from skin conductance and blood volume pulse, along with external PTSD profile information indicating PTSD symptom severity as well as self-reports of experienced stress as sources of ground truth. From the physiological signals, 45 individual features were extracted and correlated to the sources of ground truth. The results of the analyses in this paper indicate that physiological responses to StartleMart are highly correlated with PTSD symptom severity and subjective experience expressed through self-reports of stress. Additionally, an application of principal component analysis to reduce the number of features into two distinct components suggests that two response patterns are manifested in relation to the content presented in the simulation: One which is primarily related to stress resilience/parasympathetic activity and exhibits a negative correlation to external measures of PTSD symptom severity and one which is primarily related to stress sensitivity/sympathetic activity and exhibits a positive correlation to external measures of PTSD symptom severity. This underlines the complex nature of user responses to rich stimulus presenting simulations and motivates the further use and study of multiple modalities for capturing stress responses. Further, the fact that StartleMart elicits stress responses with PTSD patients lends credence to the general idea of using game-based stimuli of every-day life situations for stress inoculation training for PTSD patients. However, any treatment efficacy is unknown at this point and would require a randomized study.

Nonetheless, the fact that physiological responses seem to scale with measures of symptom severity, self-reports and an indicator of recovery over time, indicates a promise to using stress eliciting game-based solutions like StartleMart for diagnosis and treatment of PTSD. Future work will focus on leveraging these findings to refine profiling and adaptive game-based solutions supporting diagnosis and treatment in psychiatric work.

## **Acknowledgments**

This research was supported by the Danish Council for Technology and Innovation under the *Games for Health* project and by the FP7 ICT project *ILearnRW* (project no: 318803).

## Appendix F

# To Rank or to Classify? Annotating Stress for Reliable PTSD Profiling

### Reference:

Holmgård, Christoffer, Georgios N. Yannakakis, Héctor P. Martínez, and Karen-Inge Karstoft (2015). “To Rank or to Classify? Annotating Stress for Reliable PTSD Profiling”. In: *Affective Computing and Intelligent Interaction (ACII), 2015 Humaine Association Conference on*. IEEE.

### F.1 Abstract

In this paper we profile the stress responses of patients diagnosed with post-traumatic stress disorder (PTSD) to individual events in the game-based PTSD stress inoculation and exposure virtual environment *StartleMart*. Thirteen veterans suffering from PTSD play the game while we record their skin conductance. Game logs are used to identify individual events, and continuous decomposition analysis is applied to the skin conductance signals to derive event-related stress responses. The extracted skin conductance features from this analysis are used to profile each individual player in terms of stress response. We observe a large degree of variation across the 13 veterans which further validates the idiosyncratic nature of PTSD physiological manifestations. Further to game data and skin conductance signals we ask PTSD patients to indicate the most stressful event experienced (class-based annotation) and also compare the stress level of all events in a pairwise preference manner (rank-based annotation). We compare the

two annotation stress schemes by correlating the self-reports to individual event-based stress manifestations. The self-reports collected through class-based annotation exhibit no correlation to physiological responses, whereas, the pairwise preferences yield significant correlations to all skin conductance features extracted via continuous decomposition analysis. The core findings of the paper suggest that reporting of stress preferences across events yields more reliable data that capture aspects of the stress experienced and that features extracted from skin conductance via continuous decomposition analysis offer appropriate predictors of stress manifestation across PTSD patients.

## F.2 Introduction

This paper describes a game-based method for profiling *post traumatic stress disorder* (PTSD) using affective responses to events in a virtual environment incorporating principles of game design, called the *StartleMart* game (Holmgård et al., 2013b; Holmgård et al., 2014c). We describe how we construct a specific paradigm for eliciting and capturing affective responses to particular in-game events of 13 PTSD patients through the *StartleMart* game. We then use these responses to enable individual PTSD profiling and stress detection through continuous decomposition analysis (Benedek and Kaernbach, 2010a) of *skin conductance* (SC) manifestations to in-game stressor events. Our results indicate that participants' memories of the most stressful events correspond poorly to SC responses, but their ordering of events in terms of stressfulness corresponds strongly to these same responses.

The work presented here is inspired by previous pioneering work on the use of virtual environments and games as tools supporting affective learning in PTSD patients (Parsons and Rizzo, 2008; Wiederhold and Wiederhold, 2008) and it builds upon previous work (Holmgård et al., 2013b; Holmgård et al., 2014c) in which it is demonstrated that features from affective responses to game play sessions in the *StartleMart* game contain useful indications of the symptom severity of PTSD patients, pointing to the relevance of simulations and games for affective learning and profiling. Here, we move from correspondence analysis at the game session level to analyzing *event*-based stress responses in order to obtain not only general indications of the patient's affective response but also to identify responses to individual events. By identifying and profiling responses to stressful situations in war veterans suffering from PTSD we create a novel and efficient method for understanding the syndrome configuration of the individual patient.

The paper is novel in that continuous decomposition analysis is applied for extracting appropriate indicators of sympathetic arousal from skin conductance in PTSD patients. Such an approach allows us to derive stress detectors such as tonic and phasic drivers

of skin conductance soon after a stressful event (elicitor) is presented to users (Benedek and Kaernbach, 2010a; Bach and Friston, 2013). Additionally we, for the first time, compare two different stress self-annotation schemes for their consistency to manifested stress via skin conductance. Our results further validate evidence and observations in the literature (Metallinou and Narayanan, 2013; Yannakakis and Hallam, 2011; Martinez et al., 2014; Yannakakis and Martinez, 2015) suggesting that rank-based (compared to class-based or rating-based) annotation yields better approximators of the ground truth of experienced emotion.

### F.3 Background: PTSD, Games, and Affect

In this section we describe the PTSD syndrome, the relationship between games and PTSD treatment, PTSD’s links to human physiology and its affective manifestations.

#### F.3.1 Post Traumatic Stress Disorder

Post Traumatic Stress Disorder (PTSD) is a psychiatric diagnosis describing an often severely disabling syndrome that is sometimes developed after being exposed to highly stressful situations. Veterans from military operations are a high-risk group for developing this syndrome (Hoge et al., 2004).

Two well-known treatment approaches for PTSD — favored because of strong evidence for their therapeutic efficacy — are the *cognitive-behavioral therapy* techniques of exposure therapy and stress inoculation training. In exposure therapy, the therapist confronts the patient with anxiety provoking stimuli in a controlled setting in order to extinguish reactions to the stimuli and/or allow the patient to reprocess the memories cued by the stimuli. Three common variations are the use of real life stimuli i.e. *in vivo*, representing stimuli via media i.e. *mediated*, or having the patient imagine the stress provoking situations and thus self-generate the stimuli i.e. *imaginal* (Foa et al., 2009). In stress inoculation training, the therapist exposes the patient to stimuli and situations that are not directly linked to the original trauma of the patient, but that cause problematic anxiety responses that are difficult for the patient to cope with (Foa et al., 2009).

#### F.3.2 Games for PTSD treatment

Games and game-like worlds have successfully been used as mental health interventions by appropriating commercial games (Holmes et al., 2009) and by developing specialized solutions (Hoque et al., 2009). Among the possible ways of treating PTSD, computer



games and virtual environments have a particular potential for eliciting stress in a controlled, graded fashion and can provide an immersive and rich medium for PTSD treatment (Wiederhold and Wiederhold, 1998; Wiederhold and Wiederhold, 2008; Rizzo et al., 2009a; Rizzo et al., 2009a). Earlier research has demonstrated the usefulness of virtual environments for treating veterans' PTSD with *virtual reality therapy*, an extension of exposure therapy (Parsons and Rizzo, 2008; Wood et al., 2011). Some implementations of virtual reality therapy have focused on exposing the patient to the original stressful, traumatizing situation, in the vain of classic exposure therapy. Notable examples are the *Virtual Iraq* and *Virtual Afghanistan* applications that show promising results in clinical testing (Reger et al., 2011; Rizzo et al., 2009a). Other implementations have focused on appropriating principles from stress inoculation training (Wiederhold and Wiederhold, 2008). StartleMart implements a hybrid of exposure therapy and stress inoculation training.

The conversation between the user and the therapist is central for treatment efficacy (Foa et al., 2009). The user's perception of which events are stressful, and the user's actual physiological stress responses to the same events, are important touchstones in this conversation. Therefore, we see a need for self-report schemes that allow the user to accurately report which events were perceived as stressful. We address this by comparing two schemes for self-reporting experiences of stressful events, classification and ranking, with physiological responses as the ground truth.

### **F.3.3 Physiology of PTSD**

When placed in mediated stimulus exposure paradigms, PTSD-patients exhibit physiological responses to stressful visual and auditive stimuli that are significantly different from the responses of non-patients (Perala, 2007). Their responses are generally characterized by high sympathetic activity as measured by SC. Responses that have been found to be robust indicators of PTSD conditions via experimental studies include slower SC habituation, elevated resting SC, and greater SC responses to startling stimuli (Pole, 2007). In general, higher base levels of arousal and heightened sensitivity to stress seem to characterize the physiological manifestations of the disorder. It has been suggested that these differences could be used to support diagnostic differentiation between PTSD patients and non-patients as well as between different degrees of PTSD symptom severity (Blechert et al., 2007) guiding treatment strategies or allowing for adaptive treatment tools (Wood et al., 2011).

Although the general symptomatology of PTSD is consistent across sufferers, every instance of the syndrome includes idiosyncratic aspects related to the particular individual and the instigating stressful experience. Which particular events elicit the strongest stress-response or trigger flashbacks vary across individuals and research has shown the strength of the response to be contingent upon the stimulus relation to the original trauma (Liberzon et al., 1999). This characteristic of the PTSD syndrome proves a challenge when developing any treatment or diagnostic tool as we assume that greater efficacy and precision comes at a cost of lower generalizability. For virtual environments, this is particularly challenging as convincing environments are time-consuming and expensive to develop. In order for a virtual environment for PTSD to be as generally useful and cost-effective as possible, it should apply to as wide a range of PTSD symptomatology as possible.

A design solution to this challenge is presented in the stress-inoculation training approach taken in this study. By choosing a stimulus delivery environment that is predominantly related to the everyday strivings of PTSD patients (Kashdan et al., 2010), rather than the original trauma, some specificity is sacrificed, but a greater relevance across patients is attempted. The general relevance of the approach has been demonstrated in (Holmgård et al., 2013b) and (Holmgård et al., 2014c) which focus on feature extraction from full length sessions, while here we move to an event-level analysis. Simultaneously, the everyday environment supports a patient/therapist conversation about the stimuli present in everyday life that are most stressful to the particular patient, supporting the stress inoculation process.

In the following section we outline the key features of the StartleMart game; for the interested reader the game is described in greater detail in (Holmgård et al., 2013b; Holmgård et al., 2014c).

## F.4 The StartleMart Game for PTSD Treatment

The task of shopping in a supermarket is a common situation that is severely challenging to many patients suffering from PTSD (Kashdan et al., 2010). Supermarkets are highly stimulating environments with social interactions and unpredictable auditory and visual experiences which PTSD patients find stressful; some to the extent that they avoid going shopping or only do so with a helper present for emotional support. Consequently, the game is built to primarily take place in a virtual supermarket (see Fig. F.1). The supermarket environment includes a number of stressors that aim at eliciting stress in the player. These are designed around three typical symptoms of PTSD, namely *fear-avoidance behavior*, *hyper-arousal* (i.e. heightened startle response), and



(A) Sound of ventilator blowing overhead.



(B) Sound of wind blowing.



(c) Man walking angrily toward player.



(d) Man running toward player.



(E) Man staring at player.



(F) Wounded soldier staring at player.

FIGURE F.1: The three flashbacks of the game (b, d, f) and the immediately preceding supermarket scenes (a, c, e). Elements of the supermarket bleed into the flashbacks, simulating re-experience.

*re-experiencing* of traumatic events triggered by an outside stimulus or general stress (Foa et al., 2009). Stressors targeting *fear-avoidance behavior* include the layout of the supermarket which is designed to include hidden angles and preventing the player from attaining a full overview of the location and Non-Player-Characters (NPCs) that provide

socially stressful experiences. Stressors targeting *hyper-arousal* include a dog barking at the entrance to the supermarket and the sound of crashes and glass breaking suddenly playing at random locations in the supermarket. Stressors targeting *re-experiencing* are included in the form of three different flashbacks.

## F.5 Experimental Protocol and Data Collection

In this section we provide details about the participants of our experiment and the experimental protocol followed. Thirteen male PTSD patients, veterans from Danish military operations in Afghanistan, are included in the study presented in this paper. The participants are in psychiatric treatment for PTSD and qualify for the PTSD diagnosis. All subjects are medicated with *Selective Serotonin Re-uptake Inhibitors (SSRI)* which is known to generally lower sympathetic activity and in particular SC (Siepmann et al., 2003). This clearly adds a challenge to the detection of SC stress responses to game stimuli since patients are expected to manifest responses that are pharmacologically suppressed to an unknown degree. Each patient participates in the experiment twice, engaging in a total of 6 game play sessions, 3 per participation (11 patients have participated in both sessions, while 3 participated in the first session only), progressing through low stress intensity, medium stress intensity, and high stress intensity. The experimenters, trained psychologists, welcome the participant, complete a diagnostic interview with the patient and collect various instances of demographic and background data from either the patient himself or the patient's medical records. The participant is introduced to the experimental setup and seated in front of the controls and monitor. The biofeedback device is attached to the participant's fingertips (see more details in Section F.5.1), and a brief introduction to the game rules and how to control the game is given. Following a short waiting period, collecting baseline SC data, the participant is asked to play three sessions of the game. Subjective data (self-reports) is collected over the course of the experiment. Finally, the experimenter debriefs the participant, responding to any concerns or issues the patient might have.

### F.5.1 Physiological Sensors and Setup

For continuous measurement of SC the IOM biofeedback device<sup>1</sup> is used. The IOM biofeedback device samples SC at a rate of 300 Hz and down-samples them to 30 Hz in firmware before transmitting them to the recording computer. The device's measuring electrodes are attached dryly to the distal phalanges of the little and middle fingers of the

---

<sup>1</sup><http://www.wilddivine.com/>

patient's non-dominant hand. A sensor measuring blood volume pulse is attached to the ring finger, but is not used for the study and analysis presented here. Since frustration with the control scheme of the game might introduce unwanted variation and artifacts to the results of the experiment (Yannakakis et al., 2010) the game is configured to use standard controls for first-person-perspective computer games which should be familiar to most patients. The mouse, operated with the patient's dominant hand, controls the perspective and the keyboard controls movement. To minimize the risk of movement artifacts in the physiological readings, patients operate the keyboard (W, A, S, D or arrow keys) with only the index finger of their non-dominant hand, keeping the other fingers still.

### F.5.2 Game Logging

During game play, a number of features are logged constantly. A screen-shot from the player's perspective is logged every second to allow for reconstruction of the events of the game play session and for subsequent identification of the most stressful experiences. Stressor stimulus presentations are logged as *game events* whenever they occur. The four types of events are labeled as: 1) *sound* events, when sudden sounds of crashes and glass breaking are played, 2) *pickup* events, when the player obtains one of the items on the shopping list, 3) *social* events, when the player is close to one or more of the NPCs in the supermarket and 4) *flashback* events, when the flashback of the session is presented.

## F.6 Self-reports and SC Feature Extraction

In this section we describe the data collected from the two stress self-report schemes (Section F.6.1) and the SC features extracted via continuous composition analysis (Section F.6.2).

### F.6.1 Stress Self-reports

At the end of each session, the patient is asked to indicate which singular event during the session was considered the most stressful, if any. This self-report is subsequently reduced to one of the four event categories: social, sound, pickup, and flashback. The player is then presented with a *4-alternative-forced-choice* (4AFC) survey comprised of series of pairs constructed from all events that happened during the game. The two screen-shots that were taken closest in time to each event are presented side by side as a representation of each event. The player is asked to express a preference for which of

the two events that was most stressful, if both were equally stressful, or if neither of the events were stressful. The player completes this process for pairings of all logged events producing a global ordering of the events in terms of stressfulness.

### F.6.2 Skin Conductance Event Response Features

The trough-to-peak analysis of skin conductivity response (SCR) amplitude, area or similar measures, can be subject to super-positioning of phasic and tonic activity. This may necessitate the subtraction of baseline measures or other forms of signal correction (Boucsein, 2011). It has been suggested that even with such corrections one may still confound phasic and tonic SC which is undesirable in a study focusing predominantly on event-related activation (Benedek and Kaernbach, 2010a).

To address this potential issue, features of the player's SC at the time of the event are extracted using Continuous Decomposition Analysis (CDA) as described in (Benedek and Kaernbach, 2010a). The method allows for the decomposition of phasic and tonic electro-dermal activity. It initially separates super-positioned phasic and tonic components of the raw SC signal. Subsequently it adapts a general model of the human skin's impulse response function (representing the basic SCR shape that would result from a unit impulse) to the phasic activity by sampling the tonic component around the event response to establish a local baseline and fitting the general impulse response function to the shape of the phasic component. The result is expressed in a *phasic driver* measured in  $\mu S$  that approximates the phasic response affecting the signal within the event window. As such, the phasic driver across the event window can be interpreted as a locally baseline-corrected measure of the patient's SC response to the event. As a result of the decomposition procedure the phasic driver value can take on negative values. A detailed example from the CDA process is provided in Fig. F.2. More details about the CDA method can be found in (Benedek and Kaernbach, 2010a). A 1 – 4 s after-event response window is applied, meaning that only activation occurring within this window is considered relevant to the event (see Fig. F.2). A minimum phasic driver threshold value of 0.05  $\mu S$  is used, meaning that only events with a phasic driver value exceeding this threshold are considered significant and counted as SCRs.

From the CDA result, four skin conductance response features are extracted for each event in the game: the mean phasic driver within the skin conductance response window ( $S_p$ ), the integral of the phasic driver within the response window ( $i_p$ ), the mean tonic SC ( $S_t$ ) within the window, and the global mean within the window ( $S_g$ ). The literature suggests that features based on the phasic driver of skin conductance are supreme detectors of heightened sympathetic arousal (Benedek and Kaernbach, 2010a). We therefore

trust that the four features extracted are appropriate indicators of stress elicited around the stressful events provided by the game (Bach and Friston, 2013; Boucsein, 2011).

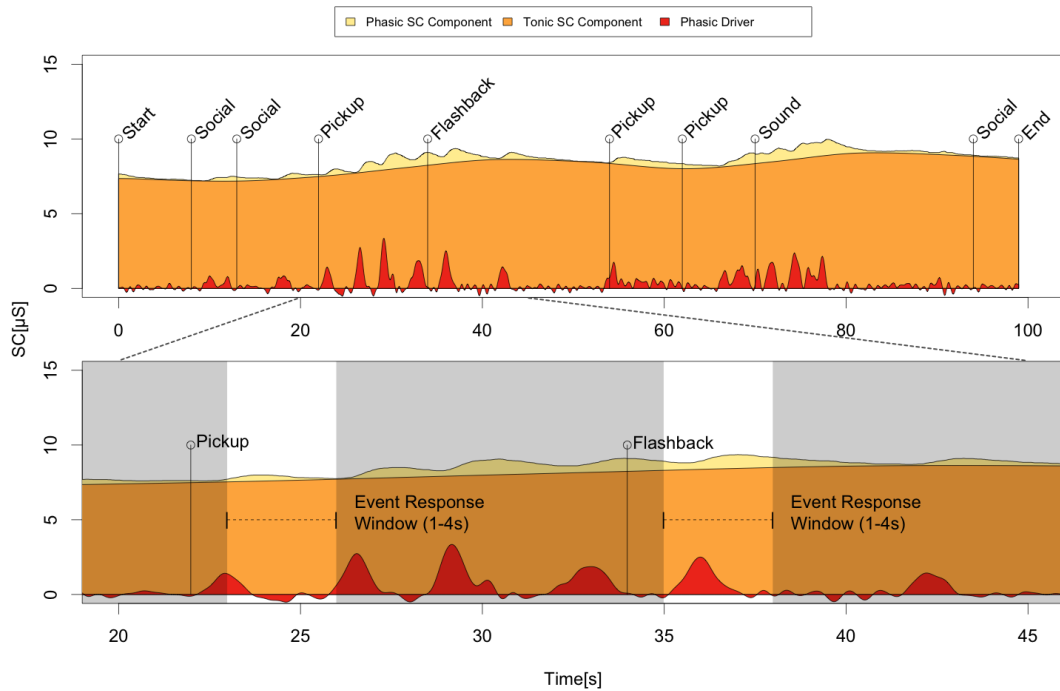


FIGURE F.2: Continuous Decomposition Analysis of SC of Player 5, Session 3. The top graph shows the full game session whereas the bottom graph shows a detailed view of an excerpt from that session. Both graphs depict three components extracted from the raw SC signal: Phasic activity (yellow), tonic activity (orange), and the phasic driver (red) of SC. SC features are extracted within the event response windows as illustrated in the bottom graph.

## F.7 Results

In this section we present a descriptive overview of the SC responses of the patients of the study to the included events demonstrating the individual differences in responses to the events in the StartleMart and the complex relationship between physiological responses and subjective experience. Moreover, we investigate the relation between the two self-report schemes and the extracted SC features. First, in Section F.7.1 we correlate patient classifications of the most stressful event with the extracted skin conductance features. Then, in Section F.7.2 we correlate preferences of the stressfulness of events with the same features.

58 sessions were deemed free of protocol and uncorrectable sensor artifacts and thus usable for analysis. Sessions ranged in length from 66 to 202 seconds, with an average length of 144.8s and a standard deviation of 36.3s. Together, these sessions provided 479 events over the defined phasic driver threshold value, an average of 8.3 events per

session, with a range from 2 to 11 above threshold events per session and a standard deviation of 2.2 events.

### F.7.1 Classifying Stress

Our first approach to obtaining self-reports from patients was to ask them to pick of the most stressful event during the game session. All events in each session of the dataset are annotated with this classification: labeling an event as either the most stressful or not. A descriptive overview of chosen events and  $S_p$  values for each patient and each session is provided in Fig. F.3. The figure shows the differences between individuals in physiological responses to events, differences in self-reports of experienced stress, and discrepancies between the physiological responses to events and self-reports of which events were experienced as most stressful. Most patients exhibit a tendency to report the flashbacks of StartleMart as the most stressful type of experience; however, their physiological responses across events indicate that responses to other types of events, on average, are stronger. Figure F.3 indicates the need of more complex, non-linear, models for mapping the physiological responses of patients to the perceived experience of stress from interacting with the StartleMart game. Further it demonstrates illustratively the inconsistency between the most stressful events reported by the patients and heightened sympathetic arousal (via the mean phasic driver feature).

To investigate the relationship between the events classified as most stressful and their corresponding SC features values, the annotated events are correlated to the four extracted SC features using the binomially-distributed pairwise correlation described in (Yannakakis and Hallam, 2011) which is calculated as:  $c(z) = \sum_i \{z_i\} / N$ ; where  $N$  is the sample size, and  $z_i$  is 1 if there is an agreement between the annotation (e.g. most stressful) and the corresponding SC feature and  $-1$  otherwise. The  $c(z)$  values are calculated by considering annotation agreements with the most stressful event as determined by physiology (i.e. the corresponding SC feature). To further explore potential effects between reported and manifested stress we also calculate potential agreements between the reported event and the three, five and ten most stressful events as manifested by SC features. The results reported in Table F.1 showcase significant negative correlations when the single, three and five most stressful events are considered. This already indicates a poor consistency between class-based self-reports and physiological indications of stress. Only when the ten most stressful events are considered we start observing a minor positive correlation which demonstrates that some of the largest physiological responses are captured in the class (most stressful event reports) only when 10 of those events are considered.





FIGURE F.3: Individual event response patterns: Patients and event types are cross tabulated. The color of each cell represents the patient’s mean normalized  $S_p$  response to the event type, across all sessions, extracted using Continuous Decomposition Analysis (Benedek and Kaernbach, 2010a). The size of each green dot indicates the number of times the patient picks the corresponding type of event as the most stressful. For fair comparison purposes, both the event response levels and the event preference frequencies are normalized into  $[0, 1]$  within each individual, across all sessions.

### F.7.2 Ranking Stress

Our second approach to obtaining reliable stress self-reports from PTSD patients uses the fully ordered preference-pairs constructed from the 4AFC selections. These are pairwise-correlated to the extracted SC also using the  $c(z)$  test statistic. The p-value, in this case, is obtained from the normal distribution as the binomially-distributed  $c(z)$  approximates the normal distribution when large samples are considered. As shown in Table F.1 significant positive correlations are found between reported ranks of stress and all four extracted SC features. This suggests that reported stress preferences expressed through 4AFC yield event orderings that are highly consistent with the orderings of the physiological responses as measured via any of the four features. In particular,  $S_p$  and  $i_p$  demonstrate the highest correlation values with reported stress preferences. Based on the obtained results it appears that preference-based annotation of stress is a reliable

TABLE F.1: Pairwise rank-correlations ( $c(z)$ ) and corresponding p-values ( $p$ ) between the most stressful event reported (class) or the reported rankings of stressful events (preference) and the four skin conductance features.

		$S_p$	$i_p$	$S_t$	$S_g$
Class (Most stressful event)	$c(z)$	-0.86	-0.86	-0.76	-0.76
	$p$	< 0.01			
Class (3 most stressful events)	$c(z)$	-0.45	-0.45	-0.24	-0.31
	$p$	< 0.01	< 0.01	0.02	< 0.01
Class (5 most stressful events)	$c(z)$	-0.17	-0.17	-0.17	-0.17
	$p$	0.04			
Class (10 most stressful events)	$c(z)$	0.14	0.14	0.14	0.14
	$p$	0.06			
Preference	$c(z)$	0.43	0.43	0.33	0.37
	$p$	< 0.01			

self-report measure of stress elicitation from events in PTSD patients. It is also obvious that classifying events as most stressful is an annotation practice that fails to capture the idiosyncratic nature of both PTSD and its physiological manifestations.

## F.8 Discussion and Conclusions

In this paper, we have presented a set of key findings on PTSD profiling and stress detection via games from a sample of 13 clinical PTSD patients interacting with the StartleMart game. Earlier work has demonstrated that this game has an ability to instigate and detect stress in patients to a degree that scales with PTSD symptom severity (Holmgård et al., 2013b). The work presented here extends that study by characterizing the individual patient symptoms from subjective experience, event-based skin conductance responses, and game logging data. Further, we introduce continuous decomposition analysis for extracting features of skin conductivity near the stressful in-game events that are appropriate indicators of stress responses.

The study demonstrates the challenges of fusing interactive environments with a stimulus exposure approach, since the agency afforded to the player significantly impacts the experimenter’s or therapist’s control over the flow of events. Our analysis shows a high degree of inter-subject variability in terms of which events the patients responded strongest to as measured via physiology, and which events they reported as being subjectively most stressful. This finding is not surprising given the general literature on PTSD, but underlines that PTSD patients’ responses to events in virtual environments exhibit the same variation across subjects as responses in more controlled stimulus-exposure paradigms or in everyday life.

When asked patients to recall the most stressful event in a session and comparing this to physiological responses around the corresponding events, no significant effect is observed. Speculatively, this may be due to memory effects where patients attribute the experienced stress to the most salient and heterogeneous event in the session. However, when patients are presented with a memory cue of each event individually and rank all of them in relation to each other, a strong correlation emerges between self-reports of stress and physiological responses. This suggests that using preference-based ranking paradigms, such as the 4AFC, provides better support for recalling the experienced stress in response to individual events even though it comes with an additional effort of comparing across all possible combinations of experienced events. It is important to note that we did not consider the comparison between rating-based annotation against the rank-based and the class-based approaches in this study. Ratings are ordinal values which are already obtained via the reported preferences of the PTSD patients (Martinez et al., 2014). Further, this study is based on a relatively narrow, homogeneous sample of patients and generalizability to other conditions or groups is an open question.

The core findings of the paper indicate that rank-based stress annotation to a series of events (i.e. rank two or more stressful events) is a beneficial method for detecting stress compared to class-based (i.e. what is the most stressful event) annotation which further validates the observations of earlier studies in affect annotation and modeling (Metallinou and Narayanan, 2013; Yannakakis and Hallam, 2011; Martinez et al., 2014; Yannakakis and Martinez, 2015). Results also suggest that the phasic driver (as obtained from continuous decomposition analysis) is a highly reliable predictor of PTSD severity manifested via skin conductance responses to in-game events. It is possible that these findings may extend to other affective disorders with arousal components, such as e.g. anxiety, though this remains an open question for future work.

## Acknowledgments

This research was supported by the Danish Council for Technology and Innovation under the *Games for Health* project and by the FP7 ICT project SIREN (project no: 258453). We would also like to thank the consortium members of the *Games for Health* project and all the PTSD patients who chose to support our research with their participation.

# Bibliography

- Aarseth, Espen J (1997). *Cybertext: Perspectives on Ergodic Literature*. JHU Press.
- Acharya, U Rajendra, K Paul Joseph, N Kannathal, Choo Min Lim, and Jasjit S Suri (2006). “Heart Rate Variability: A Review”. In: *Medical and Biological Engineering and Computing* 44.12, pp. 1031–1051.
- Ahn, Hyung-il (2010). “Modeling and Analysis of Affective Influences on Human Experience, Prediction, Decision Making, and Behavior”. PhD thesis. Massachusetts Institute of Technology.
- Ahn, Hyungil and Rosalind W Picard (2006). “Affective cognitive learning and decision making: The role of emotions”. In: *The 18th European Meeting on Cybernetics and Systems Research (EMCSR 2006)*, pp. 1–6.
- Andreassi, J.L. (2000). “Psychophysiology: Human Behavior and Physiological Response”. In:
- Antunes, Rui Filipe and Frederic Fol Leymarie (2010). “Virtual Worlds as Art Practice: EvoArt Methodologies”. In: *Proceedings of the 2010 International Conference on Electronic Visualisation and the Arts*. British Computer Society, pp. 271–277.
- Asteriadis, Stylianos, Kostas Karpouzis, Noor Shaker, and Georgios N Yannakakis (2012). “Towards Detecting Clusters of Players using Visual and Gameplay Behavioral Cues”. In: *Procedia Computer Science* 15, pp. 140–147.
- Bach, Dominik R and Karl J Friston (2013). “Model-Based Analysis of Skin Conductance Responses: Towards Causal Models in Psychophysiology”. In: *Psychophysiology* 50.1, pp. 15–22.
- Bartle, Richard (1996). “Hearts, Clubs, Diamonds, Spades: Players who Suit MUDs”. In: *Journal of MUD research* 1.1, p. 19.
- Bauer, Aaron William, Seth Cooper, and Zoran Popovic (2013). “Automated Redesign of Local Playspace Properties”. In: *Foundations of Digital Games*. Citeseer, pp. 190–197.
- Baumgarten, Robin (2013). *Robins Pages! :: projects:marioai*. URL: <http://www.doc.ic.ac.uk/~rb1006/projects:marioai>.

- Bechara, A., H. Damasio, D. Tranel, and A.R. Damasio (1997). “Deciding Advantageously before Knowing the Advantageous Strategy”. In: *Science* 275.5304, pp. 1293–1295.
- Benedek, Mathias and Christian Kaernbach (2010a). “A Continuous Measure of Phasic Electrodermal Activity”. In: *Journal of neuroscience methods* 190.1, pp. 80–91.
- (2010b). “Decomposition of Skin Conductance Data by Means of Nonnegative Deconvolution”. In: *Psychophysiology* 47.4, pp. 647–658.
- Berdichevsky, D. and E. Neuenschwander (1999). “Toward an Ethics of Persuasive Technology”. In: *Communications of the ACM* 42.5, pp. 51–58.
- Bethesda Softworks (1996). *The Elder Scrolls II: Daggerfall*. Bethesda Softworks.
- Björk, Staffan and Jussi Holopainen (2004). *Patterns in Game Design*. Charles River Media.
- Blanchard, Edward B, Jacqueline Jones-Alexander, Todd C Buckley, and Catherine A Forneris (1996). “Psychometric Properties of the PTSD Checklist (PCL)”. In: *Behaviour research and therapy* 34.8, pp. 669–673.
- Blechert, Jens, Tanja Michael, Paul Grossman, Marta Lajtman, and Frank H. Wilhelm (2007). “Autonomic and Respiratory Characteristics of Posttraumatic Stress Disorder and Panic Disorder”. In: *Psychosomatic Medicine* 69.9, pp. 935–943. DOI: 10.1097/PSY.0b013e31815a8f6b.
- Blizzard Entertainment (2014). *World of Warcraft*. Blizzard Entertainment.
- Blizzard North (1996). *Diablo*. Blizzard Entertainment.
- Borsboom, Denny (2005). *Measuring the Mind: Conceptual Issues in Contemporary Psychometrics*. Cambridge University Press.
- Borsboom, Denny, Gideon J Mellenbergh, and Jaap van Heerden (2004). “The Concept of Validity”. In: *Psychological Review* 111.4, p. 1061.
- Boucsein, W. (2011). *Electrodermal Activity*. Springer Verlag.
- Bourki, Amine, Guillaume Chaslot, Matthieu Coulm, Vincent Danjean, Hassen Doghmen, Jean-Baptiste Hooek, Thomas Héroult, Arpad Rimmel, Fabien Teytaud, Olivier Teytaud, et al. (2010). “Scalability and Parallelization of Monte-Carlo Tree Search”. In: *Computers and Games*. Springer, pp. 48–58.
- Braben, David and Ian Bell (1984). *Elite*. Acornsoft.
- Brough, Michael (2013). *868-HACK*. Smestorp Limited.
- Brown, Joseph Alexander (2015). “Towards Better Personas in Gaming: Contract Based Expert Systems”. In:
- Browne, Cameron B, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton (2012). “A Survey of Monte Carlo Tree Search Methods”. In: *IEEE Transactions on Computational Intelligence and AI in Games* 4.1, pp. 1–43.

- Bruin, Wändi Bruine de, Andrew M Parker, and Baruch Fischhoff (2007). “Individual Differences in Adult Decision-Making Competence.” In: *Journal of personality and social psychology* 92.5, p. 938.
- Caillois, Roger (2001). *Man, Play, and Games*. University of Illinois Press.
- Calvo, Rafael A and Sidney D’Mello (2010). “Affect Detection: An Interdisciplinary Review of Models, Methods, and their Applications”. In: *Affective Computing, IEEE Transactions on* 1.1, pp. 18–37.
- Canossa, Alessandro (2012). “Give Me a Reason to Dig: Qualitative Associations between Player Behavior in Minecraft and Life Motives”. In: *Proceedings of the International Conference on the Foundations of Digital Games*. ACM, pp. 282–283.
- Canossa, Alessandro and Yun-Gyung Cheong (2011). “Between Intention and Improvisation: Limits of Gameplay Metrics Analysis and Phenomenological Debugging”. In: *DiGRA Think, Design, Play*.
- Canossa, Alessandro and Anders Drachen (2009). “Patterns of Play: Play-Personas in User-Centred Game Development”. In: *Proceedings of the 2009 DiGRA International Conference*.
- Canossa, Alessandro, Jeremy B. Badler, Magy Seif El-Nasr, Stefanie Tignor, and C. Randall Colvin (2015). “In Your Face(t). Impact of Personality and Context on Gameplay Behavior”. In: *Foundations of Digital Games*.
- Caspersen, Mads, Kristian Helle Jespersen, and Christoffer Holmgård Pedersen (2006). “Constructive Play”. MA thesis. IT University of Copenhagen. URL: <http://heureka.dk/wp-content/uploads/2008/04/Player-Behavior.pdf> (visited on 04/14/2013).
- Champanard, Alex J. *Monte-Carlo Tree Search in TOTAL WAR: ROME II’s Campaign AI*. URL: <http://aigamedev.com/open/coverage/mcts-rome-ii/> (visited on 08/12/2014).
- (2009). “Infinite Mario AI using A\* Search: The Definitive Interview with Robin Baumgarten”. In: URL: <http://aigamedev.com/open/interviews/mario-ai/> (visited on 04/14/2013).
- Cook, Michael, Simon Colton, and Jeremy Gow (2014). “Automating Game Design in Three Dimensions”. In: *Proceedings of the AISB Symposium on AI and Games*, pp. 20–24.
- Cowley, Bill, Darryl Charles, Michaela Black, and Ray Hickey (2009). “Analyzing Player Behavior in Pacman using Feature-Driven Decision Theoretic Predictive Modeling”. In: *Proceedings of the IEEE Conference on Computational Intelligence and Games, 2009. CIG 2009*. IEEE, pp. 170–177.
- Cowley, Douglas, Gurkan Te, Matt Mcfarland, and Andrew Tyler (2013). *Hoplite*. Magma Fortress.
- Damasio, A. (2000). *The Feeling of What Happens: Body and Emotion in the Making of Consciousness*. Harvest Books.

- Damasio, A. (2008). *Descartes' error: Emotion, Reason and the Human Brain*. Vintage Digital.
- Dennett, Daniel C. (1987). *The Intentional Stance*. The MIT Press, Cambridge, MA.
- Dormans, J. (2010). "Adventures in Level Design: Generating Missions and Spaces for Action Adventure Games". In: *Workshop on Procedural Content Generation in Games*.
- Drachen, A., A. Canossa, and G.N. Yannakakis (2009a). "Player Modeling using Self-Organization in Tomb Raider: Underworld". In: *Proceedings of the IEEE Conference on Computational Intelligence and Games*. IEEE, pp. 1–8.
- Drachen, Anders, Alessandro Canossa, and Georgios N Yannakakis (2009b). "Player Modeling using Self-Organization in Tomb Raider: Underworld". In: *Proceedings of the IEEE Conference on Computational Intelligence and Games*. IEEE, pp. 1–8.
- El-Nasr, Magy Seif, Anders Drachen, and Alessandro Canossa (2013). *Game Analytics: Maximizing the Value of Player Data*. Springer Science & Business Media.
- Elias, George Skaff, Richard Garfield, and K. Robert Gutschera (2012). *Characteristics of Games*. MIT Press.
- Finnsson, Hilmar and Yngvi Björnsson (2008). "Simulation-Based Approach to General Game Playing". In: *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*. Vol. 8, pp. 259–264.
- Firaxis Games (2010). *Sid Meier's Civilization V*. 2K Games.
- First, Michael B., Robert L Spitzer, Miriam Gibbon, and Janet B.W. Williams (2002). *Structured Clinical Interview for DSM-IV-TR Axis I Disorders, Research Version, Patient Edition. (SCID-I/P)*. New York: Biometrics Research, New York State Psychiatric Institute.
- Flin, Rhona H (1997). *Decision Making under Stress: Emerging Themes and Applications*. Gower Technical.
- Foa, Edna B., Terence M. Keane, Matthew J. Friedman, and Judith A. Cohen (2009). *Effective Treatments for PTSD, Practice Guidelines from the International Society for Traumatic Stress Studies*. Second. New York: Guilford Press.
- Fogg, B.J. (2003). *Persuasive Technology Persuasive Technology. Using Computers to Change What We Think and Do*. San Francisco: Morgan Kaufmann Publishers.
- Force, T. (1996). "Heart Rate Variability: Standards of Measurement, Physiological Interpretation and Clinical use. Task Force of the European Society of Cardiology and the North American Society of Pacing and Electrophysiology". In: *Circulation* 93.5, pp. 1043–65.
- Friedman, B. and P.H. Kahn Jr (2002). "Human Values, Ethics, and Design". In: *The Human-Computer Interaction Handbook*. L. Erlbaum Associates Inc., pp. 1177–1201.
- Fullerton, T., C. Swain, and S. Hoffman (2004). *Game Design Workshop: Designing, Prototyping, and Playtesting Games*. Focal Press.

- Fürnkranz, Johannes and Eyke Hüllermeier (2005). "Preference Learning". In: *Künstliche Intelligenz* 19.1, pp. 60–61.
- Gibson, James J. (1977). "The Concept of Affordances". In: *Perceiving, Acting, and Knowing*, pp. 67–82.
- Gigerenzer, Gerd and Wolfgang Gaissmaier (2011). "Heuristic Decision Making". In: *Annual Review of Psychology* 62, pp. 451–482.
- Gigerenzer, Gerd and Reinhard Selten (2002). *Bounded Rationality: The Adaptive Toolbox*. MIT Press.
- Goldberger, J.J., S. Challapalli, R. Tung, M.A. Parker, and A.H. Kadish (2001). "Relationship of Heart Rate Variability to Parasympathetic Effect". In: *Circulation* 103.15, p. 1977.
- Gosling, James, Sun Microsystems, and Oracle Coporation (2011). *Java 7*. Oracle Corporation.
- Gregory, R.J. (2004). *Psychological Testing: History, Principles, and Applications*. Allyn & Bacon.
- Grigorenko, E.L. and R.J. Sternberg (1995). "Thinking Styles". In:
- Haag, A., S. Goronzy, P. Schaich, and J. Williams (2004). "Emotion Recognition using Bio-Sensors: First Steps towards an Automatic System". In: *Affective Dialogue Systems*, pp. 36–48.
- Hartsook, K., A. Zook, S. Das, and M. Riedl (2011). "Toward Supporting Stories with Procedurally Generated Game Worlds". In: *Proceedings of the IEEE Conference on Computational Intelligence and Games*, pp. 297–304.
- Healey, Jennifer A and Rosalind W Picard (2005). "Detecting Stress during Real-World Driving Tasks using Physiological Sensors". In: *Intelligent Transportation Systems, IEEE Transactions on* 6.2, pp. 156–166.
- Hernandez, Javier, Rob R Morris, and Rosalind W Picard (2011). "Call Center Stress Recognition with Person-Specific Models". In: *Affective Computing and Intelligent Interaction*. Springer, pp. 125–134.
- Hidden Path Entertainment and Valve Corporation (2012). *Counter Strike: Global Offensive*. Valve Corporation.
- Hingston, Philip (2010). "A New Design for a Turing Test for Bots". In: *Proceedings of the IEEE Conference on Computational Intelligence and Games*, pp. 345–350.
- ed. (2012). *Believable Bots*. Springer.
- (2013). *The 2K BotPrize. Can Computers Play like People?* URL: <http://botprize.org> (visited on 08/01/2013).
- Hoge, Charles W, Carl A Castro, Stephen C Messer, Dennis McGurk, Dave I Cotting, and Robert L Koffman (2004). "Combat Duty in Iraq and Afghanistan, Mental Health Problems, and Barriers to Care". In: *New England Journal of Medicine* 351.1, pp. 13–22.



- Holmes, Emily A, Ella L James, Thomas Coode-Bate, and Catherine Deeptose (2009). “Can Playing the Computer Game “Tetris” Reduce the Build-Up of Flashbacks for Trauma? A Proposal from Cognitive Science”. In: *PLoS One* 4.1, e4153.
- Holmgård, Christoffer, Julian Togelius, and Georgios N. Yannakakis (2013a). “Decision Making Styles as Deviation from Rational Action. A Super Mario Case Study.” In: *Ninth Annual AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.
- Holmgård, Christoffer, Georgios N. Yannakakis, Karen-Inge Karstoft, and Henrik Steen Andersen (2013b). “Stress Detection for PTSD via the StartleMart Game”. In: *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*. IEEE, pp. 523–528.
- Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis (2014a). “Evolving Personas for Player Decision Modeling”. In: *Proceedings of the IEEE Conference on Computational Intelligence and Games*.
- (2014b). “Generative Agents for Player Decision Modeling in Games”. In: *Proceedings of the 9th Conference on the Foundations of Digital Games*.
- Holmgård, Christoffer, Georgios N. Yannakakis, Héctor P. Martínez, Karen-Inge Karstoft, and Henrik Steen Andersen (2014c). “Multimodal PTSD characterization via the startlemart game”. In: *Journal on Multimodal User Interfaces*, pp. 1–13.
- Holmgård, Christoffer, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis (2014d). “Personas versus Clones for Player Decision Modeling”. In: *Proceedings of the International Conference on Entertainment Computing*.
- Holmgård, Christoffer, Antonios Liapis, Togelius Julian, and Georgios N. Yannakakis (2015). “MiniDungeons 2: An Experimental Game for Capturing and Modeling Player Decisions”. In: *Proceedings of the 10th Conference on the Foundations of Digital Games*.
- Hoque, Mohammed E, Joseph K Lane, Rana El Kaliouby, Matthew Goodwin, and Rosalind W Picard (2009). “Exploring Speech Therapy Games with Children on the Autism Spectrum”. In:
- Horswill, Ian and Leif Foged (2012). “Fast Procedural Level Population with Playability Constraints”. In: *Proceedings of the Artificial Intelligence and Interactive Digital Entertainment Conference*.
- Hullett, Kenneth and Jim Whitehead (2010). “Design Patterns in FPS levels”. In: *Proceedings of the Fifth International Conference on the Foundations of Digital Games*. ACM, pp. 78–85.
- id Software (2010). *Quake Live*. id Software.
- Ijsselstein, W., K. Poels, and YAW de Kort (2008). “The Game Experience Questionnaire: Development of a Self-Report Measure to Assess Player Experiences of Digital Games”. In: *TU Eindhoven, Eindhoven, The Netherlands*.

- Isaksen, Aaron, Dan Gopstein, and Andy Nealen (2015). “Exploring Game Space using Survival Analysis”. In: *Foundations of Digital Games*.
- Jacobsen, Emil Juul, Rasmus Greve, and Julian Togelius (2014). “Monte Mario: Platforming with MCTS”. In: *Proceedings of the 2014 Conference on Genetic and Evolutionary Computation*. ACM, pp. 293–300.
- Juul, Jesper (2011). *Half-Real: Video Games between Real Rules and Fictional Worlds*. MIT Press.
- Kahneman, Daniel (2003). “Maps of Bounded Rationality: Psychology for Behavioral Economics”. In: *American Economic Review*, pp. 1449–1475.
- (2011). *Thinking, Fast and Slow*. Farrar, Straus and Giroux.
- Kahneman, Daniel and Gary Klein (2009). “Conditions for Intuitive Expertise: A Failure to Disagree.” In: *American Psychologist* 64.6, p. 515.
- Kahneman, Daniel and Amos Tversky (1979). “Prospect Theory: An Analysis of Decision Under Risk”. In: *Econometrica: Journal of the Econometric Society* 47 (2), pp. 263–291.
- Kaptein, M. and D. Eckles (2010). “Selecting Effective Means to Any End: Futures and Ethics of Persuasion Profiling”. In: *Persuasive technology*, pp. 82–93.
- Kaptein, M., P. Markopoulos, B. de Ruyter, and E. Aarts (2009). “Can You be Persuaded? Individual Differences in Susceptibility to Persuasion”. In: *Human-Computer Interaction–INTERACT 2009*, pp. 115–118.
- Karakovskiy, Sergey and Julian Togelius (2012). “The Mario AI Benchmark and Competitions”. In: *IEEE Transactions on Computational Intelligence and AI in Games* 4.1, pp. 55–67.
- Kashdan, Todd B, William E Breen, and Terri Julian (2010). “Everyday Strivings in War Veterans with Posttraumatic Stress Disorder: Suffering from a Hyper-Focus on Avoidance and Emotion Regulation”. In: *Behavior therapy* 41.3, pp. 350–363.
- Kaufman, Leonard and Peter J. Rousseeuw (2005). *Finding Groups in Data: An Introduction to Cluster Analysis*. Second Edition. New Jersey: Wiley.
- Kemp, Andrew H, Daniel S Quintana, Marcus A Gray, Kim L Felmingham, Kerri Brown, and Justine M Gatt (2010). “Impact of Depression and Antidepressant Treatment on Heart Rate Variability: A Review and Meta-Analysis”. In: *Biological Psychiatry* 67.11, pp. 1067–1074.
- Kendall, MG (1970). *Rank Correlation Methods*. 4th edition. London: Griffin.
- Kimbrough, Steven Orla, Gary J. Koehler, Ming Lu, and David Harlan Wood (2008). “On a Feasible-Infeasible Two-Population (FI-2Pop) Genetic Algorithm for Constrained Optimization: Distance Tracing and No Free Lunch”. In: *European Journal of Operational Research* 190.2, pp. 310–327.
- Klein, G.A. (1993). “A Recognition-Primed Decision (RPD) Model of Rapid Decision Making”. In: *Decision Making in Action: Models and Methods* 5.4, pp. 138–147.

- Krishnan, R, G Sivakumar, and P Bhattacharya (1999a). “A Search Technique for Rule Extraction from Trained Neural Networks”. In: *Pattern Recognition Letters* 20.3, pp. 273–280.
- (1999b). “Extracting Decision Trees from Trained Neural Networks”. In: *Pattern Recognition* 32.12.
- Lehmann, E and H. J. M. D’Abrera (1975). *Nonparametrics: Statistical Methods Based on Ranks*. San Francisco: Holden-Day.
- Liapis, Antonios (2014). “Searching for Sentient Design Tools for Game Development”. PhD thesis. Copenhagen: IT University of Copenhagen.
- Liapis, Antonios, Georgios N. Yannakakis, and Julian Togelius (2013a). “Designer Modeling for Personalized Game Content Creation Tools”. In: *Proceedings of the AIIDE Workshop on Artificial Intelligence & Game Aesthetics*.
- (2013b). “Generating Map Sketches for Strategy Games”. In: *Proceedings of Applications of Evolutionary Computation*. Vol. 7835, LNCS. Springer, pp. 264–273.
- (2013c). “Sentient Sketchbook: Computer-Aided Game Level Authoring”. In: *Proceedings of the 8th Conference on the Foundations of Digital Games*, pp. 213–220.
- (2013d). “Towards a Generic Method of Evaluating Game Levels”. In: *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.
- Liapis, Antonios, Christoffer Holmgård, Georgios N. Yannakakis, and Julian Togelius (2015). “Procedural Personas as Critics for Dungeon Generation”. In: *Applications of Evolutionary Computation*. Vol. 9028, Lecture Notes in Computer Science. Springer.
- Liberzon, Israel, Stephan F Taylor, Richard Amdur, Tara D Jung, Kenneth R Chamberlain, Satoshi Minoshima, Robert A Koeppe, and Lorraine M Fig (1999). “Brain Activation in PTSD in Response to Trauma-Related Stimuli”. In: *Biological psychiatry* 45.7, pp. 817–826.
- Linden, R. van der, R. Lopes, and R. Bidarra (2013). “Procedural Generation of Dungeons”. In: *IEEE Transactions on Computational Intelligence and AI in Games* 6.1, pp. 78–89.
- Lipshitz, Raanan, Gary Klein, Judith Orasanu, and Eduardo Salas (2001). “Taking Stock of Naturalistic Decision Making”. In: *Journal of Behavioral Decision Making* 14.5, pp. 331–352.
- Loo, R. (2000). “A Psychometric Evaluation of the General Decision-Making Style Inventory”. In: *Personality and Individual Differences* 29.5, pp. 895–905.
- Lorentz, Richard J (2011). “Improving Monte-Carlo Tree Search in Havannah”. In: *Computers and Games*. Springer, pp. 105–115.
- Magruder, Kathryn M, B Christopher Frueh, Rebecca G Knapp, Michael R Johnson, James A Vaughan, Toni Coleman Carson, Donald A Powell, and Renée Hebert (2004).

- “PTSD Symptoms, Demographic Characteristics, and Functional Status among Veterans Treated in VA Primary Care Clinics”. In: *Journal of traumatic stress* 17.4, pp. 293–301.
- Malaby, Thomas M. (2007). “Beyond Play. A New Approach to Games”. In: *Games and Culture* 2.2, pp. 95–113.
- Mark, D. (2009). *Behavioral Mathematics for Game AI*. Course Technology Cengage Learning.
- Marler, R Timothy and Jasbir S Arora (2004). “Survey of Multi-objective Optimization Methods for Engineering”. In: *Structural and multidisciplinary optimization* 26.6, pp. 369–395.
- Martinez, Hector, Georgios Yannakakis, and John Hallam (2014). “Don’t Classify Ratings of Affect; Rank them!” In: *IEEE Transactions on Affective Computing*.
- Martínez, Héctor P, Kenneth Hullett, and Georgios N Yannakakis (2010). “Extending Neuro-Evolutionary Preference Learning through Player Modeling”. In: *Computational Intelligence and Games (CIG), 2010 IEEE Symposium on*. IEEE, pp. 313–320.
- Martínez, Héctor Perez, Maurizio Garbarino, and Georgios N Yannakakis (2011). “Generic Physiological Features as Predictors of Player Experience”. In: *Affective Computing and Intelligent Interaction*. Springer, pp. 267–276.
- Martinez, Héctor Perez, Yoshua Bengio, and Georgios N Yannakakis (2013). “Learning Deep Physiological Models of Affect”. In: *Computational Intelligence Magazine, IEEE* 8.2, pp. 20–33.
- McFall, M.E., M.M. Murburg, G.N. Ko, and R.C. Veith (1990). “Autonomic Responses to Stress in Vietnam Combat Veterans with Posttraumatic Stress Disorder”. In: *Biological Psychiatry* 27.10, pp. 1165–1175.
- McMahan, A. (2003). “Immersion, Engagement and Presence”. In: *The Video Game Theory Reader*, pp. 67–86.
- Metallinou, Angeliki and Shrikanth Narayanan (2013). “Annotation and Processing of Continuous Emotional Attributes: Challenges and Opportunities”. In: *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*. IEEE, pp. 1–8.
- Mi-Clos Studio (2014). *Out There*. Mi-Clos Studio.
- Michalewicz, Z., D. Dasgupta, R.G. Le Riche, and M. Schoenauer (1996). “Evolutionary Algorithms for Constrained Engineering Problems”. In: *Computers & Industrial Engineering Journal* 30 (2), pp. 851–870.
- Mikkulainen, R, B Bryant, R Cornelius, I Karpov, K Stanley, and C Yong (2006). “Computer Intelligence in Games”. In: *Computational Intelligence: Principles and Practice, 155-191*. IEEE Computational Intelligence Society.
- Miyamoto, Shigeru and Takashi Tezuka (1985). *Super Mario Bros*. Nintendo.

- Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. (2015). “Human-Level Control through Deep Reinforcement Learning”. In: *Nature* 518.7540, pp. 529–533.
- Moody, G.B. (1993). “Spectral Analysis of Heart Rate without Resampling”. In: *Proceedings of Computers in Cardiology*. IEEE, pp. 715–718.
- Nemeroff, C.B., J.D. Bremner, E.B. Foa, H.S. Mayberg, C.S. North, and M.B. Stein (2006). “Posttraumatic Stress Disorder: A State-of-the-Science Review”. In: *Journal of Psychiatric Research* 40.1, pp. 1–21.
- Norling, E. and L. Sonenberg (2004). “Creating Interactive Characters with BDI Agents”. In: *Proceedings of the Australian Workshop on Interactive Entertainment (IE’04)*, pp. 69–76.
- Ortega, Juan, Noor Shaker, Julian Togelius, and Georgios N Yannakakis (2013). “Imitating Human Playing Styles in Super Mario Bros”. In: *Entertainment Computing* 4.2, pp. 93–104.
- Osborn, Joseph C and Michael Mateas (2014). “A Game-Independent Play Trace Dissimilarity Metric”. In: *Foundations of Digital Games*.
- Osborn, Joseph C, April Grow, and Michael Mateas (2013). “Modular Computational Critics for Games”. In: *Ninth Artificial Intelligence and Interactive Digital Entertainment Conference*.
- Parsons, T.D. and A.A. Rizzo (2008). “Affective Outcomes of Virtual Reality Exposure Therapy for Anxiety and Specific Phobias: A Meta-Analysis”. In: *Journal of Behavior Therapy and Experimental Psychiatry* 39.3, pp. 250–261.
- Pedersen, Christoffer Holmgård and Georgios N. Yannakakis (2012). “Epistemological Challenges for Operationalism in Player Experience Research”. In: *Proceedings of the 7th International Conference of Foundations of Digital Games*.
- Perala, C.H. (2007). *Galvanic Skin Response as a Measure of Soldier Stress*. Tech. rep. DTIC Document.
- Perez, Diego, Spyridon Samothrakis, Simon Lucas, and Philipp Rohlfshagen (2013). “Rolling Horizon Evolution versus Tree Search for Navigation in Single-Player Real-Time Games”. In: *Proceedings of the 15th annual conference on Genetic and evolutionary computation*. ACM, pp. 351–358.
- Perez, Diego, Spyridon Samothrakis, Julian Togelius, Tom Schaul, Simon Lucas, Adrien Couëtoux, Jeyull Lee, Chong-U Lim, and Tommy Thompson (2015). “The 2014 General Video Game Playing Competition”. In: *IEEE Transactions on Computational Intelligence and AI in Games*.
- Picard, Rosalind (2000). *Affective Computing*. Second. MIT Press.

- Picard, R.W., E. Vyzas, and J. Healey (2001). "Toward Machine Emotional Intelligence: Analysis of Affective Physiological State". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23.10, pp. 1175–1191.
- Pole, Nnamdi (2007). "The Psychophysiology of Posttraumatic Stress Disorder: A Meta-Analysis." In: *Psychological Bulletin* 133.5, p. 725.
- Popovic, S., M. Slamic, and K. Cosic (2005). "Scenario Self-Adaptation in VR Exposure Therapy for PTSD". In: *NATO Advanced Research Workshop—Novel Approaches to the Diagnosis and Treatment of Posttraumatic Stress Disorder*, pp. 13–15.
- Popović, S., M. Horvat, D. Kukulja, B. Dropuljić, and K. Ćosić (2009). "Stress Inoculation Training Supported by Physiology-Driven Adaptive Virtual Reality Stimulation". In: *electronic form only:: NE*.
- Rabin, Steve (2004). "Common Game AI Techniques". In: *AI Game Programming Wisdom 2*, pp. 3–14.
- Rao, A.S. and M.P. Georgeff (1997). "Modeling Rational Agents within a BDI-architecture". In: *Readings in Agents*, pp. 317–328.
- Reas, Casey and Benjamin Fry (2001). *Processing*.
- Reeves, Byron and Clifford Nash (1996). *The Media Equation: How People Treat Computers, Television, and New Media like Real People and Places*. Stanford: Cambridge University Press.
- Reger, G.M., K.M. Holloway, C. Candy, B.O. Rothbaum, J.A. Difede, A.A. Rizzo, and G.A. Gahm (2011). "Effectiveness of Virtual Reality Exposure Therapy for Active Duty Soldiers in a Military Mental Health Clinic". In: *Journal of traumatic stress* 24.1, pp. 93–96.
- Risi, Sebastian and Julian Togelius (2014). "Neuroevolution in Games: State of the Art and Open Challenges". In: *arXiv preprint arXiv:1410.7326*.
- Rizzo, A., G. Reger, G. Gahm, J.A. Difede, and B.O. Rothbaum (2009a). "Virtual Reality Exposure Therapy for Combat-Related PTSD". In: *Post-Traumatic Stress Disorder*, pp. 375–399.
- Rizzo, A.A., J. Difede, B.O. Rothbaum, S. Johnston, R.N. McLAY, G. Reger, G. Gahm, T. Parsons, K. Graap, and J. Pair (2009b). "VR PTSD Exposure Therapy Results with Active Duty OIF/OEF Combatants". In: *Medicine meets virtual reality*. Vol. 17, pp. 277–282.
- Robinson, Michael D and Gerald L Clore (2002). "Belief and Feeling: Evidence for an Accessibility Model of Emotional Self-Report". In: *Psychological Bulletin* 128.6, p. 934.
- Roden, T. and I. Parberry (2004). "From Artistry to Automation: A Structured Methodology for Procedural Content Creation". In: *Proceedings of the International Conference on Entertainment Computing*, pp. 151–156.
- Rollings, Andrew and Dave Morris (2004). *Game Architecture and Design: A New Edition*. New Riders Indianapolis.

- Rosenthal, Casey and Clare Bates Congdon (2012). “Personality Profiles for Generating Believable Bot Behaviors”. In: *Proceedings of the IEEE Conference on Computational Intelligence and Games*.
- Ross, K.G., G.A. Klein, P. Thunholm, J.F. Schmitt, and H.C. Baxter (2004). *The Recognition-Primed Decision Model*. Tech. rep. DTIC Document.
- Rubinstein, Ariel (1998). *Modeling Bounded Rationality*. Vol. 1. MIT Press.
- Schrum, Jacob and Risto Miikkulainen (2008). “Constructing Complex NPC Behavior via Multi-Objective Neuroevolution”. In: *AIIDE* 8, pp. 108–113.
- Schulte-Mecklenbeck, M., A. Kühberger, and R. Ranyard (2010). *A Handbook of Process Tracing Methods for Decision Research: A Critical Review and User’s Guide*. Psychology Pr.
- Scott, Susanne G. and Reginald A. Bruce (1995). “Decision-Making Style: The Development and Assessment of a New Measure”. In: *Educational and Psychological Measurement* 55.5, pp. 818–831.
- Setiono, Rudy and Wee Kheng Leow (1999). “On Mapping Decision Trees and Neural Networks”. In: *Knowledge-Based Systems* 12.3, pp. 95–99.
- Shaker, Noor, Mohammad Shaker, and Julian Togelius (2013a). “Evolving Playable Content for Cut the Rope through a Simulation-Based Approach”. In: *Ninth Annual AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.
- Shaker, Noor, Julian Togelius, and Georgios N. Yannakakis (2013b). *Turing Test Track - Mario AI Championship 2012*. URL: <http://www.marioai.org/> (visited on 03/10/2013).
- Siepmann, Martin, Jens Grossmann, Michael Mück-Weymann, and Wilhelm Kirch (2003). “Effects of Sertraline on Autonomic and Cognitive Functions in Healthy Volunteers”. In: *Psychopharmacology* 168.3, pp. 293–298.
- Simon, Herbert A (1955). “A Behavioral Model of Rational Choice”. In: *The Quarterly Journal of Economics*, pp. 99–118.
- Smith, Adam M, Chris Lewis, Kenneth Hullett, Gillian Smith, and Anne Sullivan (2011). “An Inclusive Taxonomy of Player Modeling”. In: *University of California, Santa Cruz, Tech. Rep. UCSC-SOE-11-13*.
- Smith, Adam M., Eric Butler, and Zoran Popovic (2013). “Quantifying over Play: Constraining Undesirable Solutions in Puzzle Design”. In: *Foundations of Digital Games*, pp. 221–228.
- Smith, Heide Jonas (2006). “Plans and Purposes: How Videogame Goals Shape Player Behaviour”. PhD thesis. IT University of Copenhagen.
- Stanley, Kenneth O and Risto Miikkulainen (2002). “Evolving Neural Networks through Augmenting Topologies”. In: *Evolutionary Computation* 10.2, pp. 99–127.
- Sturtevant, N (2013). “An Argument for Large-Scale Breadthfirst Search for Game Design and Content Generation via a Case Study of Fling”. In: *AI in the Game Design*

- Process, Workshop at the Ninth Annual AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment.*
- Sutton, Richard S and Andrew G Barto (1998). *Reinforcement Learning: An Introduction*. Cambridge University Press.
- Thomsen, Yvonne Duval, Robert Jonassen, Dorthe Berntsen, Kim Berg Johannesen, Christina Rams Hoxer, Christoffer Holmgård Pedersen, Per Bach, Henrik Klausen, and Mette Bertelsen (2011). *Undersøgelse af psykiske efterreaktioner hos soldater udsendt til Afghanistan perioden februar – august 2009*. Forsvarsakademiets Forlag. Copenhagen.
- Thunholm, P. (2004). “Decision-Making Style: Habit, Style or Both?” In: *Personality and Individual Differences* 36.4, pp. 931–944.
- Togelius, Julian, Renzo De Nardi, and Simon M Lucas (2007). “Towards Automatic Personalised Content Creation for Racing Games”. In: *Proceedings of the IEEE Conference on Computational Intelligence and Games*. IEEE, pp. 252–259.
- Togelius, Julian, Sergey Karakovskiy, and Robin Baumgarten (2010). “The 2009 Mario AI Competition”. In: *IEEE Congress on Evolutionary Computation*. IEEE, pp. 1–8.
- Togelius, Julian, Georgios N. Yannakakis, Kenneth O. Stanley, and Cameron Browne (2011). “Search-Based Procedural Content Generation: A Taxonomy and Survey”. In: *IEEE Transactions on Computational Intelligence and AI in Games* 3.3, pp. 172–186.
- Togelius, Julian, Georgios N. Yannakakis, and Noor Shaker (2012). “Assessing Believability”. In: *Believable Bots: Can Computers Play Like People?* Ed. by Philip Hingston. Springer. Chap. 9, pp. 215–230.
- Tognetti, Simone, Maurizio Garbarino, Matteo Matteucci, and Andrea Bonarini (2011). “The Affective Triad: Stimuli, Questionnaires, and Measurements”. In: *Affective Computing and Intelligent Interaction*. Springer, pp. 101–110.
- Toy, Michael, Glenn Wichman, Ken Arnold, and Jon Lane (1980). *Rogue*.
- Tversky, Amos and Daniel Kahneman (1974). “Judgment under Uncertainty: Heuristics and Biases”. In: *Science* 185.4157, pp. 1124–1131.
- Tychsen, Anders and Alessandro Canossa (2008). “Defining Personas in Games Using Metrics”. In: *Proceedings of the 2008 Conference on Future Play: Research, Play, Share*. ACM, pp. 73–80.
- Unity Technologies (2005). *Unity*.
- Van Hoorn, Niels, Julian Togelius, Daan Wierstra, and Jürgen Schmidhuber (2009). “Robust Player Imitation using Multiobjective Evolution”. In: *IEEE Congress on Evolutionary Computation, 2009 (IEEE CEC 2009)*, pp. 652–659.
- Watkins, Christopher J. C. H. and Peter Dayan (1992). “Q-Learning”. In: *Machine Learning* 8.3-4, pp. 279–292.



- Weathers, Frank W., Brett T. Litz, Debra S. Herman, Jennifer A. Huska, Terrence M. Keane, et al. (1993). "The PTSD Checklist (PCL): Reliability, Validity, and Diagnostic Utility". In: *Annual Meeting of the International Society for Traumatic Stress Studies*.
- Westland, J Christopher (2011). "Electrodermal Response in Gaming". In: *Journal of Computer Networks and Communications* 2011.
- Wiederhold, Brenda K. and Mark D. Wiederhold (1998). "A Review of Virtual Reality as a Psychotherapeutic Tool". In: *CyberPsychology & Behavior* 1.1, pp. 45–52.
- (2008). "Virtual Reality for Posttraumatic Stress Disorder and Stress Inoculation Training". In: *Journal of Cybertherapy & Rehabilitation* 1.1, pp. 23–35.
- Wild Divine Developer Support*. <http://www.wilddivinedev.com>. URL: <http://www.wilddivinedev.com> (visited on 03/10/2013).
- Wild Divine*. <http://www.wilddivine.com>. URL: <http://www.wilddivine.com/>.
- Wolpe, Joseph (1973). *The Practice of Behavior Therapy*. Pergamon Press New York.
- Wood, Dennis Patrick, Jennifer Webb-Murphy, Robert N McLay, Brenda K. Wiederhold, James L. Spira, Scott Johnston, Robert L. Koffman, Mark D. Wiederhold, Jeff Pyne, et al. (2011). "Reality Graded Exposure Therapy with Physiological Monitoring for the Treatment of Combat Related Post Traumatic Stress Disorder: A Pilot Study". In: *Studies in Health Technology and Informatics* 163, p. 696.
- Xamarin (2004). *Mono*.
- Xu, Qihan, Jonathan Tremblay, and Clark Verbrugge (2014). "Generative Methods for Guard and Camera Placement in Stealth Games". In: *Tenth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.
- Yannakakis, G. N. (2009). "Preference Learning for Affective Modeling". In: *Proceedings of the Int. Conf. on Affective Computing and Intelligent Interaction (ACII09)*. Amsterdam, The Netherlands.
- Yannakakis, Georgios N. and John Hallam (2008). "Entertainment Modeling through Physiology in Physical Play". In: *International Journal of Human-Computer Studies* 66.10, pp. 741–755.
- (2011). "Ranking vs. Preference: A Comparative Study of Self-Reporting". In: *Affective Computing and Intelligent Interaction*. Springer, pp. 437–446.
- Yannakakis, Georgios N. and Hector P. Martinez (2015). "Ratings are Overrated!" In: *Frontiers on Human-Media Interaction*.
- Yannakakis, Georgios N. and Julian Togelius (2014). "A Panorama of Artificial and Computational Intelligence in Games". In: *IEEE Transactions on Computational Intelligence and AI in Games* 99, p. 1.
- Yannakakis, Georgios N., Héctor P. Martínez, and Arnav Jhala (2010). "Towards Affective Camera Control in Games". In: *User Modeling and User-Adapted Interaction* 20.4, pp. 313–340.

- Yannakakis, Georgios N., Pieter Spronck, Daniele Loiacono, and Elisabeth André (2013). “Player Modeling”. In: *Artificial and Computational Intelligence in Games*. Saarbrücken/Wadern: Dagstuhl Publishing, pp. 45–55.
- Yannakakis, Georgios N., Antonios Liapis, and Constantine Alexopoulos (2014). “Mixed-Initiative Co-Creativity”. In: *Proceedings of the 9th Conference on the Foundations of Digital Games*.
- Yannakakis, G.N., M. Maragoudakis, and J. Hallam (2009). “Preference learning for Cognitive Modeling: A Case Study on Entertainment Preferences”. In: *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 39.6, pp. 1165–1175.
- Yee, N. (2006). “Motivations for Play in Online Games”. In: *CyberPsychology & Behavior* 9.6, pp. 772–775.
- Yee, Nick, Nicolas Ducheneaut, Les Nelson, and Peter Likarish (2011a). “Introverted Elves & Conscientious Gnomes: The Expression of Personality in World of Warcraft”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, pp. 753–762.
- Yee, Nick, Helen Harris, Maria Jabon, and Jeremy N Bailenson (2011b). “The Expression of Personality in Virtual Worlds”. In: *Social Psychological and Personality Science* 2.1, pp. 5–12.
- Zhai, Jing, Armando B Barreto, Craig Chin, and Chao Li (2005). “Realization of Stress Detection using Psychophysiological Signals for Improvement of Human-Computer Interactions”. In: *Proceedings of SoutheastCon*. IEEE, pp. 415–420.
- Zombie Studios (2014). *Daylight*. Atlus.
- Zook, Alexander, Brent Harrison, and Mark O Riedl (2015). “Monte-Carlo Tree Search for Simulation-based Strategy Analysis”. In: *Proceedings of the 10th Conference on the Foundations of Digital Games*.