

IT University
of Copenhagen

Analysis of the Ballot Shuffling Attack on Irish ballot counting for Proportional Representation by Single Transferable Vote (PR-STV)

Dermot Cochran

Copyright © 2015, Dermot Cochran

IT University of Copenhagen
All rights reserved.

Reproduction of all or part of this work
is permitted for educational or research use
on condition that this copyright notice is
included in any copy.

ISSN 1600-6100

ISBN 978-87-7949-352-0

Copies may be obtained by contacting:

IT University of Copenhagen
Rued Langgaards Vej 7
DK – 2300 Copenhagen S
Denmark

Telephone: +45 72 18 50 00

Telefax: +45 72 18 50 01

Web: www.itu.dk

Abstract

The current Irish legislation for counting of ballots does not fully comply with the true meaning of proportional representation by single transferable vote. This is due to the way in which second and subsequent transfers are handled, the legislative requirement to only count the last set of ballots transferred and the sensitivity of the counting algorithm to the order in which ballots are shuffled. A formal model of PR-STV is used to identify scenarios in which the counting under current electoral laws gives a different result compared to what would have happened using another ordering of the same ballots.

1. INTRODUCTION

The method of Proportional Representation by Single Transferable Vote (PR-STV) achieves proportionality through the use of a quota in multi-winner elections. The quota is calculated so that the number of candidates who can achieve a quota of votes is just less than the number of seats. The excess votes above the quota, known as the surplus, are available for re-distribution to other candidates, according to the preferences stated on each ballot. Different implementations of PR-STV use different methods for selecting which ballots belong to the surplus.

The Irish implementation of PR-STV is described in Part XIX of the 1992 Electoral Act (see Appendix A for the full text). In it there are two main shortcuts used for allocating the surplus:

- The ballots are divided into sub-piles according to the immediate next preference, but not taking account of lower preferences e.g. third and subsequent preferences (section 121 (2))
- When a candidate reaches the quota after receiving vote transfers from another candidate then only the last set of transfers received are used when allocating the surplus (section 121 (3))

Meeks found that in the current implementation of PR-STV in Ireland “there is an undesirable degree of arbitrariness involved, which has the potential to produce an outcome that violates social choice concerns and elect different parliamentarians.” [Meeks 2010]

This has also been noted by several Irish politicians as a reason for reform of PR-STV.

“The defects of the present system have often been pointed out...that the preferences of only some candidates are counted...the complicated and arbitrary method of counting and transferring votes, introduces a random element into the process and the ultimate result may be far from what the electorate really intended. The choice between two effective candidates can depend on the relative position of weaker candidates and the chance transfer of sub parcels of votes.”

(*Taoiseach* Jack Lynch, *Dáil Debates* 232: 1957, 28 February 1968)

In addition to the risk of random distortions there are two modes of attack to be considered:

- If the ballots are being shuffled and counted by hand then the order could be altered by sleight of hand before the ballots are numbered.
- If the ballots are cast and counted electronically, then the voting machine could manipulate the order of the ballots so as to bias the outcome, even if there is a paper audit trail.

2. MODEL OF THE PR-STV BALLOT COUNTING ALGORITHM

Figure 1 and table 1 describe a model of the PR-STV ballot counting algorithm in such a way as to capture the effect of different permutations of the order in which the ballots are counted. The full model, expressed using the Alloy modeling language [Jackson 2006], can be seen in Appendix B. The outcome ‘Shuffle Winner’ in figure 1 refers to a candidate who either wins or loses the election depending on the order in which the ballots are shuffled. The existence of such scenarios can be found by using a predicate that there is at least one ‘Shuffle Winner’ not caused purely by a random tie break due to equality of votes (section 122 (3) of the legislation).

3. RESULTS

The order of the shuffle can affect the outcome when there are some ballots with at least 3 preferences that agree on the top two preferences but diverge on some later preference, where the top two preferences refer to winning candidates i.e. to candidates who achieve a surplus of votes.

For example, if the first preference on the ballot turns out to be for the candidate with the first quota and the second preference turns out to be for the candidate with the second quota, then the influence of the third and subsequent preferences will depend on the position of the ballot in the shuffle.

An example is shown in table 2. In this example, due to the “last set received” rule, the surplus of candidate B is based on the vote transfers received from candidate A, so the ballot with B as first preference is never included in B’s surplus regardless of the order of the shuffle.

In order to bias the shuffle during a hand count, a shuffler does not need to know in advance who the top two candidates will be. He or she only needs to ensure that ballots are placed higher in the pile if they contain third or subsequent preferences for the shuffler’s favorite candidate.

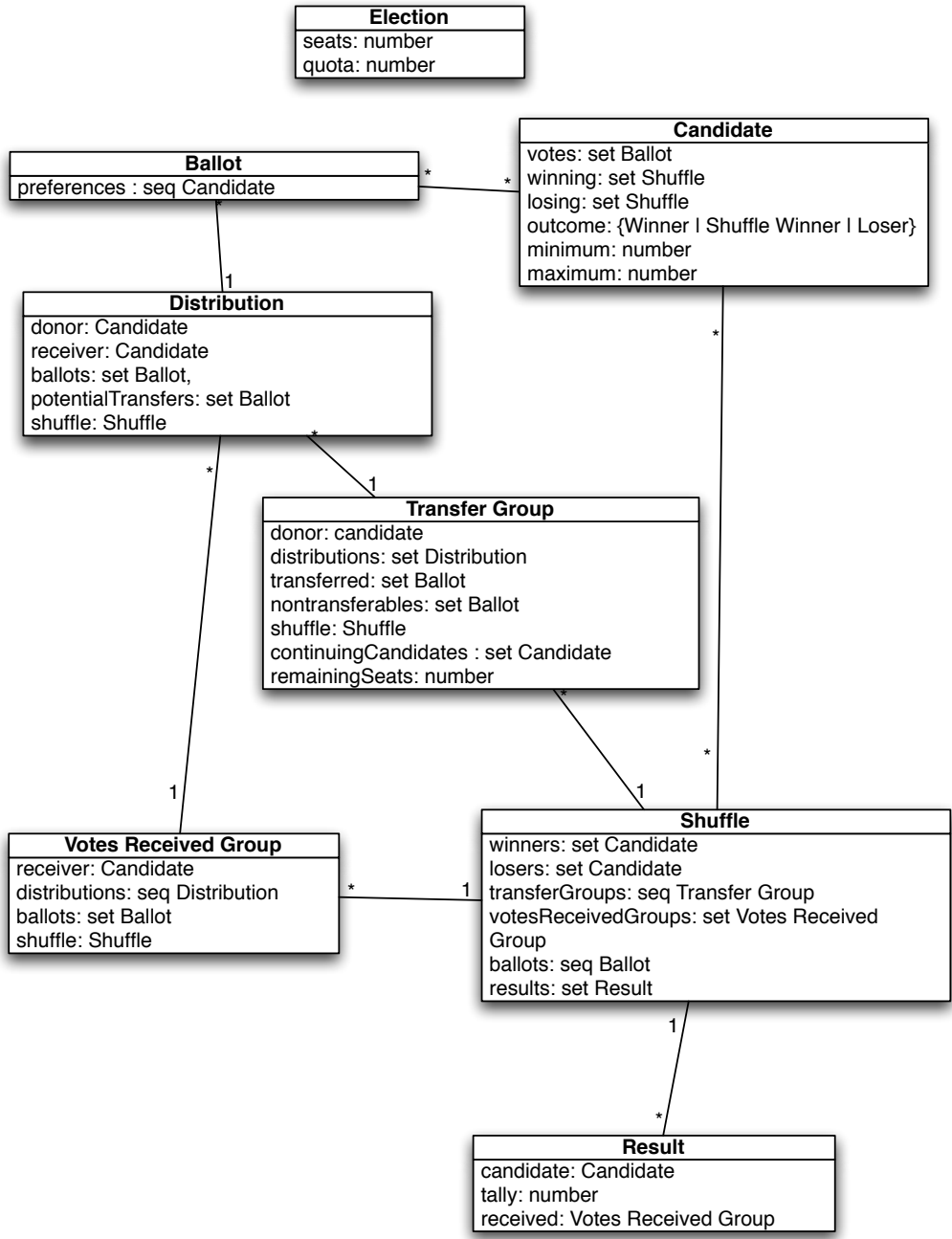


Figure 1 : Model of the effect of shuffles in PR-STV

Table 1 : Description of the model in figure 1

Entity	Field	Explanation
Ballot	Preferences	Sequence of candidates in order of preference on a ballot form

Entity	Field	Explanation
Election	Seats	Number of elected positions to be filled
Election	Quota	The minimum number of votes received by a candidate in order to ensure election
Candidate	Votes	The set of ballots in which this candidate is identified as the first preference
	Winning	The set of ballot shuffles under which this candidate would win the election
	Losing	The set of ballot shuffles under which this candidate would lose the election
	Minimum	The minimum tally for this candidate, including transfers received, under any shuffle
	Maximum	The maximum tally for this candidate, including transfers received, under any shuffle
Distribution	Donor	The candidate from which the ballots are transferred, either elected with surplus, or excluded as lowest continuing candidate
	Receiver	The continuing candidate to which the transfers are made
	Ballots	The actual ballots transferred from the donor candidate to the receiving candidate
	Potential Transfers	The sub-pile of ballots from which the actual ballots were chosen based on the order of the shuffle
	Shuffle	The shuffle in which this distribution happens

Entity	Field	Explanation
Transfer Group	Donor	The candidate from which the ballots are to be transferred
	Distributions	The set of distributions to each receiving candidate
	Transferred	The set of actual ballots transferred
	Nontransferable	Ballots that could not be transferred to a continuing candidate
	Shuffle	The shuffle to which this transfer applies
	Continuing Candidates	The set of continuing candidates who are eligible to receive transfers
	Remaining Seats	The number of positions not yet filled
Votes Received Group	Receiver	The candidate to whom votes were transferred
	Distributions	The sequence of distributions from other candidates to this candidate
	Ballots	All ballots received by this candidate under this shuffle
	Shuffle	The shuffle in which these vote transfers were received
Shuffle	Winners	The set of candidates who are elected under this shuffle
	Losers	The set of candidates who lose the election under this shuffle
	Results	The numbers of votes received by each candidate under this shuffle
	Ballots	The order in which the ballots are shuffled
	Transfer Groups	The sequence in which ballots are transferred
	Votes Received Groups	The set of groups of distributions received by each candidate under this shuffle

Entity	Field	Explanation
Result	Candidate	The candidate to which this result applies
	Votes Received Group	The set of distributions in which this candidate received votes transfers
	Tally	The total votes for this candidate including original and transferred votes upon election or exclusion

Table 2: A small example of different outcomes from the same set of ballots

Ballots (first > second > third)	Shuffle (order of ballots)	Result
A > B > C A > B > D A > B > C A > B > D B > C > D (3 seats, quota = 2)	A > B > C before A > B > D	Candidate A is elected with a surplus of 2 ballots (A > B > C), which are transferred to B who then gets elected with a surplus of 1 ballot (A > B > C) which is transferred to C who gets the last seat.
	A > B > D before A > B > C	Candidate A's surplus contains 2 ballots (A > B > D) and B's surplus then contains 1 ballot (A > B > D) which is transferred to candidate D.
	A > B > C then A > B > D then A > B > C ...	Candidate A's surplus contains 1 each of (A > B > C) and (A > B > D) so B's surplus contains either (A > B > C) or (A > B > D) chosen at random.

4. CONCLUSIONS

The current system of PR-STV in Ireland is exposed to the risk of a ballot shuffling attack in any scenario where a second surplus is distributed. This risk is partly mitigated by the use of traditional paper-based voting instead of electronic voting. However the attack is still possible to a limited extent, if sleight of hand is used during the shuffling and numbering of ballots. This risk could be further reduced by allowing a losing candidate to ask for a full reshuffle as well as a full recount.

If Ireland were to re-consider electronic voting in the future, then the vote counting rules will need to be redesigned in a way that does not depend on the order in which ballots are counted, for example, the use of fractional transfers under the Meek method of STV [Meek 1994], or alternatively to compute several possible shuffles (of the same ballots) and then calculate the ratio of favorable shuffles to unfavorable shuffles for each candidate.

REFERENCES

Weeks, L. (2011) Tolerable Chance or Undesirable Arbitrariness? Distributing Surplus Votes Under PR-STV. *Parliamentary Affairs* (2011) 64 (3): 530-551

Meek, B. L. (1994) A New Approach to the Single Transferable Vote - The Problem of non-transferable Votes. *Voting matters*: Issue 1, March 1994

Jackson, D (2006) *Software Abstractions: Logic, Language and Analysis*, MIT Press

APPENDIX A

PART XIX of the Irish 1992 Electoral Act Rules for the Counting of the Votes

118.—(1) In this Part—

“continuing candidate” means any candidate not deemed to be elected and not excluded;

“count” means—

- (a) all the operations involved in the counting of the first preferences recorded for candidates;
- (b) all the operations involved in the transfer of the surplus of a candidate deemed to be elected;
- (c) all the operations involved in the transfer of the votes of an excluded candidate; or
- (d) all the operations involved in the transfer of the votes of two or more candidates together;

“deemed to be elected” means deemed to be elected for the purpose of the counting of the votes but without prejudice to the declaration of the result of the poll;

“determine by lot” means determine in accordance with the following directions, namely, the names of the candidates concerned having been written on similar slips of paper, and the slips having been folded so as to prevent identification and mixed and drawn at random, the candidate or candidates shall in cases of exclusion be excluded in the order in which their names are drawn, and, in cases of surpluses, the surpluses shall be transferred in the order in which the names are drawn, and, in cases of equality of fractions, the fraction relating to the candidate whose name is first drawn shall be deemed to be the largest;

“non-transferable paper” means a ballot paper on which no second or subsequent preference is recorded for a continuing candidate; provided that a paper shall be deemed to have become a non-transferable paper whenever—

- (a) the names of two or more candidates (whether continuing candidates or not) are marked with marks which, in the opinion of the returning officer, indicate the same order of preference and are next in order of preference; or
- (b) the name of the candidate next in order of preference (whether a continuing candidate or not) is marked with a mark which, in the opinion of the returning officer, does not follow consecutively after some other mark on the ballot paper, or with two or more marks; or
- (c) it is void for uncertainty;

“original vote” in regard to any candidate means a vote derived from a ballot paper on which a first preference is recorded for that candidate;

“preference” shall be interpreted as follows—

- (a) “first preference” means any mark which, in the opinion of the returning officer, clearly indicates a first preference;
 - “second preference” means any mark which, in the opinion of the returning officer, clearly indicates a second preference standing in succession to a first preference;
 - “third preference” means any mark which, in the opinion of the returning officer, clearly indicates a third preference standing in succession to a second preference and so on;
- (b) “next available preference” means a preference which, in the opinion of the returning officer, is a second or subsequent preference recorded in consecutive order for a continuing candidate, the preferences next in order on the ballot paper for candidates already deemed to be elected or excluded being disregarded;

“surplus” means the number of votes by which the total number of the votes, original and transferred, credited to any candidate, exceeds the quota;

“transferable paper” means a ballot paper on which, following a first preference, a second or subsequent preference is recorded in consecutive numerical order for a continuing candidate;

“transferred vote” in regard to any candidate, means a vote derived from a ballot paper on which a second or subsequent preference is recorded for that candidate.

First count.

119.—(1) After the ballot papers have been mixed in accordance with [section 114](#) the returning officer shall, rejecting any that are invalid, arrange them in parcels according to the first preferences recorded for each candidate.

(2) The returning officer shall then count the number of papers in each parcel and credit each candidate with a number of votes equal to the number of valid papers on which a first preference has been recorded for such candidate and he shall ascertain the number of all valid papers.

The quota.

120.—(1) The returning officer shall then divide the number of all valid papers by a number exceeding by one the number of vacancies to be filled; the result increased by one, any fractional remainder being disregarded, shall be the number of votes sufficient to secure the election of a candidate and this number is referred to in this Act as “the quota”.

(2) Where at the end of any count the number of votes credited to a candidate is equal to or greater than the quota, that candidate shall be deemed to be elected.

Transfer of surplus.

121.—(1) Where at the end of any count the number of votes credited to a candidate is greater than the quota, the surplus shall be transferred in accordance with and subject to the provisions of this section to the continuing candidate or candidates indicated on the ballot papers in the parcel or sub-parcel of the candidate deemed to be elected according to the next available preferences recorded thereon.

(2) Where the votes credited to a candidate deemed to be elected whose surplus is to be transferred consist of original votes only, the returning officer shall examine all the papers in the parcel of that candidate and shall arrange the transferable papers in sub-parcels according to the next available preferences recorded thereon.

(3) Where the votes credited to a candidate deemed to be elected whose surplus is to be transferred consist of original and transferred votes, or of transferred votes only, the returning officer shall examine the papers contained in the sub-parcel last received by that candidate and shall arrange the transferable papers therein in further sub-parcels according to the next available preferences recorded thereon.

(4) In either of the cases referred to in *subsections (2) and (3)* the returning officer shall make a separate sub-parcel of the non-transferable papers and shall ascertain the number of papers in each sub-parcel of transferable papers and in the sub-parcel of non-transferable papers.

(5) Where—

(a) the surplus is equal to the total number of papers in the sub-parcels of transferable papers, the returning officer shall transfer each sub-parcel of transferable papers to the continuing candidate indicated thereon as the voters’ next available preference,

(b) the surplus is greater than the total number of papers in the sub-parcels of transferable papers, the returning officer shall proceed as specified in *paragraph (a)* and shall in addition make a sub-parcel of a number of non-transferable papers equal to the difference between such total number and the surplus and set the papers therein aside as finally dealt with, such papers being, for the purposes of [section 127 \(1\)](#), described as non-transferable papers not effective,

and the non-transferable papers or the remaining non-transferable papers, as the case may be, also arranged as a sub-parcel shall be placed with the papers of the candidate deemed to be elected.

(6) Where the surplus is less than the total number of transferable papers the following provisions shall apply:

(a) the returning officer shall transfer from each sub-parcel of transferable papers to the continuing candidate indicated thereon as the voters' next available preference that number of papers which bears the same proportion to the number of papers in the sub-parcel as the surplus bears to the total number of transferable papers,

(b) the number of papers to be transferred from each sub-parcel shall be ascertained by multiplying the number of papers in the sub-parcel by the surplus and dividing the result by the total number of transferable papers,

(c) a note shall be made of the fractions (which may be expressed in decimal form), if any, in each quotient ascertained in respect of each candidate in accordance with *paragraph (b)* and if, owing to the existence of such fractions, the number of papers to be transferred is less than the surplus, so many of these fractions taken in the order of their magnitude (beginning with the largest) as are necessary to make the total number of papers to be transferred equal to the surplus shall be reckoned as of the value of unity and the remaining fractions shall be disregarded,

(d) where two or more fractions are of equal magnitude, and it is not possible for the purposes of *paragraph (c)* to reckon all of the said fractions as of the value of unity, that fraction shall be deemed to be the largest which arises from the largest sub-parcel, and if such sub-parcels are equal in size, that fraction shall be deemed to be the largest which relates to the candidate credited with the largest number of original votes. Where the numbers of such original votes are equal, regard shall be had to the total number of votes credited to such candidates at the first count at which they were credited with an unequal number of votes, and the fraction relating to the candidate credited with the greatest number of votes at that count shall be deemed to be the largest. Where the numbers of votes credited to such candidates were equal at all counts the returning officer shall determine by lot which fraction shall be deemed to be the largest.

(7) The papers to be transferred from each sub-parcel shall be those last filed in the sub-parcel, and each paper so transferred shall be marked to indicate the number of the count at which the transfer took place.

(8) The returning officer need not necessarily transfer the surplus of a candidate deemed to be elected whenever that surplus, together with any other surplus not transferred, is less than both

the difference between the quota and the number of votes credited to the highest continuing candidate and the difference between the numbers of the votes credited to the two lowest continuing candidates and either—

- (a) the number of votes credited to the lowest candidate is greater than one quarter of the quota, or
- (b) the sum of the number of votes credited to the lowest candidate together with that surplus and any other surplus not transferred is not greater than one quarter of the quota.

(9) Where at any time there are two or more surpluses which are to be transferred, the greater or greatest of such surpluses shall first be transferred and the remaining surplus or surpluses shall then, subject to *subsection (8)*, be transferred in the order of their descending magnitude.

(10) Where two or more candidates have each an equal surplus arising from the same count regard shall be had to the number of original votes credited to each candidate and the surplus of the candidate credited with the largest number of original votes shall be first dealt with. Where the numbers of such original votes are equal regard shall be had to the total number of votes credited to such candidates at the first count at which they had an unequal number of votes and the surplus of the candidate with the greatest number of votes at that count shall be first dealt with. Where the numbers of votes credited to such candidates were equal at all counts the returning officer shall determine by lot which surplus he will first deal with.

(11) Subject to *subsections (8) and (9)*, where two or more candidates have a surplus arising from different counts, a surplus which arises at the end of any count shall be transferred before a surplus which arises at a subsequent count.

Exclusion of candidate.

122.—(1) If at any time no candidate has a surplus (or when under [section 121](#) (8) an existing surplus is not transferred) and one or more vacancies remain unfilled, the returning officer shall

- (a) exclude the candidate credited with the lowest number of votes and examine all the papers of that candidate;
- (b) arrange the transferable papers in sub-parcels according to the next available preferences recorded thereon for continuing candidates and transfer each sub-parcel to the candidate for whom the preference is recorded;
- (c) make a separate sub-parcel of the non-transferable papers and set them aside as finally dealt with, such papers being, for the purposes of [section 127](#) (1), described as non-transferable papers not effective.

(2) Where the total of the votes of the two or more lowest candidates together with any surplus not transferred is less than the number of votes credited to the next highest candidate, the returning officer shall in one operation exclude such two or more lowest candidates provided that

- (a) the number of votes credited to the second lowest candidate is greater than one quarter of the quota, or

(b) where the number of votes credited to any one of such two or more lowest candidates does not exceed one quarter of the quota, it is clear that the exclusion of the candidates separately in accordance with *subsection (1)* and the transfer of any untransferred surplus could not result in a number of votes exceeding one quarter of the quota being credited to any such candidate.

(3) If, when a candidate has to be excluded under this section, two or more candidates have each the same number of votes and are lowest, regard shall be had to the number of original votes credited to each of those candidates, and the candidate with the lowest number of original votes shall be excluded and where the numbers of the original votes are equal, regard shall be had to the total numbers of votes credited to those candidates at the first count at which they had an unequal number of votes and the candidate with the lowest number of votes at that count shall be excluded and, where the numbers of votes credited to those candidates were equal at all counts, the returning officer shall determine by lot which shall be excluded.

Transfer of votes.

123.—(1) Where a transfer of votes is made under [section 121](#) or [122](#), each sub-parcel of papers transferred shall be placed on top of the parcel, if any, of papers of the candidate to whom the transfer is made, and that candidate shall be credited with a number of votes equal to the number of papers transferred to him.

(2) If after any transfer a candidate has a surplus, that surplus shall be dealt with in accordance with and subject to the provisions of [section 121](#) before any other candidate is excluded.

Filling of last vacancies.

124.—(1) When the number of continuing candidates is equal to the number of vacancies remaining unfilled, the continuing candidates shall thereupon be deemed to be elected.

(2) When only one vacancy remains unfilled, and the votes of some one continuing candidate exceed the total of all the votes of the other continuing candidates together with any surplus not transferred, that candidate shall thereupon be deemed to be elected.

APPENDIX B

Alloy Model of PR-STV Ballot Shuffling

/** Candidate outcomes of interest, for any given set of ballots:

Winner:	Always Elected regardless of order of shuffling
ShuffleWinner:	Might win depending on the order of the shuffle
Loser:	Never Elected regardless of order of shuffling */

enum Event {Winner, ShuffleWinner, Loser}

sig Candidate {

votes: set Ballot, -- First preference ballots assigned to this candidate

```

winning:          set Shuffle, -- Shuffles under which this candidate is elected
losing:          set Shuffle, -- Shuffles under which this candidate is not elected
outcome:         Event,      -- Overall election result for candidate
minimum:         Int,        -- Minimum tally under any shuffle
maximum:         Int         -- Maximum tally under any shuffle
} {
no shuffle: (losing + winning) | some result: shuffle.results | result.candidate = this and
    result.tally < minimum
some shuffle: (losing + winning) | some result: shuffle.results | result.candidate = this and
    result.tally = minimum
no shuffle: (winning + losing) | some result: shuffle.results | result.candidate = this and
    maximum < result.tally
some shuffle: (winning + losing) | some result: shuffle.results | result.candidate = this and
    maximum = result.tally
no s: Shuffle | s in winning & losing
all w: Shuffle | w in winning implies this in w.winners
all l: Shuffle | l in losing implies this in l.losers
outcome = Loser implies (#winning = 0 and #Shuffle = #losing)
outcome = Winner implies (#losing = 0 and #Shuffle = #winning)
Election.quota <= #votes implies outcome = Winner
outcome = ShuffleWinner implies (0 < #winning and 0 < #losing)
// First preference winners always elected
Election.quota <= #votes implies outcome = Winner
}

// A ballot contains a list of candidates in order of preference
sig Ballot {preferences: seq Candidate
} {
    0 < #preferences
    not preferences.hasDups
    all c: Candidate | preferences.first = c iff this in c.votes
}

```



```

// A distribution is a set of ballots transferred from one candidate to another
sig Distribution {
  donor:      Candidate,      // Candidate excluded or with surplus votes
  receiver:   Candidate,      // Candidate to whom the votes were transferred
  ballots:    set Ballot,      // Actual ballots selected for transfer
  potentialTransfers: set Ballot, // Potential ballots that might have been selected
  shuffle:    Shuffle          // Shuffle to which this distribution applies
} {
  ballots in potentialTransfers
  // Choice of actual transfers depends on order of shuffle
  all a, p : potentialTransfers | a in ballots and (not p in ballots) implies
    shuffle.ballots.idxOf[a] < shuffle.ballots.idxOf[p]

  // All ballots in a distribution are associated with the candidates in question
  all b: Ballot | b in potentialTransfers implies
    // The receiving candidate is ranked after than the donating candidate
    b.preferences.idxOf[donor] < b.preferences.idxOf[receiver] and
    // The receiver cannot be already the first preference on the ballot
    receiver in b.preferences.rest.elems

  // This distribution belongs to a transfer group
  some t: TransferGroup | this in t.distributions and t.donor = donor and t.shuffle = shuffle

  // This distribution belongs to a collection of votes received in this shuffle
  some r: VotesReceived |
    this in r.distributions.elems and r.receiver = receiver and r.shuffle = shuffle
  all r: VotesReceived |
    (this in r.distributions.elems and r.receiver = receiver) implies r.shuffle = shuffle

  0 < #ballots
}

```

```

// Groups of transfers from the same candidate, upon either election or exclusion
sig TransferGroup {
  distributions :          set Distribution,
  donor :                Candidate,
  shuffle :              Shuffle,
  continuingCandidates :  set Candidate,
  remainingSeats :       Int,
  transferred :          set Ballot,
    -- set of ballots transferred to next preference continuing candidate
  nontransferables :     set Ballot
    -- ballots with no further preference for any continuing candidate
} {
  all d : Distribution | d in distributions implies d.donor = donor
  this in shuffle.transferGroups.elems
  not donor in continuingCandidates
  0 < remainingSeats and remainingSeats < Election.seats
  remainingSeats < #continuingCandidates

  // A transfer group can be empty if loser had no votes or winner had only surplus
  #distributions = 0 implies no r : VotesReceived | r in shuffle.votesReceived and
    r.receiver = donor and
    (0 < #donor.votes + #r.ballots or
    (donor in shuffle.winners and Election.quota < #donor.votes + #r.ballots))

  // Selection of ballots for re-distribution
  some disj x, y : Ballot | x + y in donor.votes and x in transferred and (not y in transferred)
    implies (shuffle.ballots.idxOf[x] < shuffle.ballots.idxOf[y] or y in nontransferables)

  no disj x, y : distributions | x.receiver = y.receiver
  all d : Distribution | d in distributions implies d.receiver in continuingCandidates

```

```

no d : Distribution | some b : Ballot | b in nontransferables and
    d in distributions and b in d.ballots
no b : Ballot | some c : Candidate | b in nontransferables and
    c in continuingCandidates and
    b.preferences.idxOf[donor] < b.preferences.idxOf [c]
no b : Ballot | b in transferred & nontransferables

// Distribution in favour of the next preference continuing candidate
no d : Distribution | some b : Ballot | some c : Candidate |
    b in d.ballots and d in distributions and c in continuingCandidates and
    b.preferences.idxOf[c] < b.preferences.idxOf[d.receiver]

// Size of surplus from an elected candidate
some r : VotesReceived | r in shuffle.votesReceived and r.receiver = donor and
    donor in shuffle.winners implies
    ((Election.quota + sum d : distributions | #d.ballots) + #nontransferables =
    (#donor.votes + #r.ballots))

// Size of transfer from an excluded candidate
some r : VotesReceived | r in shuffle.votesReceived and r.receiver = donor and
    donor in shuffle.losers implies
    (((sum d : distributions | #d.ballots) + #nontransferables =
    (#donor.votes + #r.ballots))
    and (nontransferables in donor.votes + r.ballots))

all b : Ballot | some d : Distribution | d in distributions and b in transferred implies
    (b in d.ballots and d.ballots in transferred)
}

// All sets of vote transfers received by the same continuing candidate
sig VotesReceived {
    distributions : seq Distribution,

```

```

receiver : Candidate,
ballots : set Ballot,
shuffle : Shuffle
} {
  all d : Distribution | d in distributions.elems implies d.receiver = receiver
  this in shuffle.votesReceived
  all b : Ballot | b in ballots implies
    some d : Distribution | d in distributions.elems and b in d.ballots
  all d : Distribution | d in distributions.elems implies d.ballots in ballots
  not distributions.hasDups
}

// Final Tally for each candidate
sig Result {
  candidate:      Candidate,
  received:       VotesReceived,
  tally:          Int
} {
  tally = #candidate.votes + #received.ballots
  received.receiver = candidate
}

// The election result for a particular shuffling of the ballots
sig Shuffle {
  ballots:        seq Ballot,          // Ordered set of shuffled ballots
  losers:         set Candidate,       // Failed candidates
  winners:        set Candidate,       // Elected candidates
  results:        set Result,          // Final tally for each candidate
  transferGroups: seq TransferGroup,   // Order of transfers between candidates
  votesReceived: set VotesReceived
  // Sets of transfers received by each candidate
} {

```

```

all c : Candidate | some r : Result | r in results and c = r.candidate

// All votes received are included in the results
all v : VotesReceived | some r : Result |
    r in results and v = r.received and v.receiver = r.candidate

// No loser has a higher tally than any winner in this shuffle
no disj loser, winner : results |
    winner.candidate in winners and
    loser.candidate in losers and
    winner.tally < loser.tally

// First transfer group must either be from highest winner above quota
transferGroups.first.donor in winners implies
    Election.quota < #transferGroups.first.donor.votes and
    (no c : Candidate | #transferGroups.first.donor.votes < #c.votes)

// or else from lowest continuing candidate (based on first preference votes)
transferGroups.first.donor in losers implies
    no c : Candidate | #c.votes < #transferGroups.first.donor.votes
// Each transfer group occurs only once
not transferGroups.hasDups

// Only continuing candidates can receive transfers
all disj x, y : TransferGroup | all d : Distribution | x + y in transferGroups.elems and
    d in y.distributions and x.donor = d.receiver implies
    transferGroups.idxOf[y] < transferGroups.idxOf[x]

// The number of continuing candidates decreases with each round of transfers
all disj x, y : TransferGroup | transferGroups.idxOf [x] < transferGroups.idxOf [y] implies
    (y.remainingSeats <= x.remainingSeats and
    y.continuingCandidates in x.continuingCandidates and

```

y.donor in x.continuingCandidates)

// Any non-continuing candidate was a donor in a previous round of transfers
// unless they had no votes or not surplus

all c : Candidate | all current : TransferGroup | some previous : TransferGroup |
(current + previous in transferGroups.elems
and transferGroups.idxOf[previous] < transferGroups.idxOf[current] and
c not in current.continuingCandidates) implies c = previous.donor

// The number of remaining seats decreases each time that a surplus is transferred

all disj x, y : TransferGroup | (transferGroups.idxOf [x] < transferGroups.idxOf [y] and
y.donor in winners) implies
(y.remainingSeats < x.remainingSeats or #x.distributions = 0)

// Surplus is always taken from the last set received

all disj surplus, previous : TransferGroup | all partOfSurplus : Distribution |
(surplus + previous) in transferGroups.elems and
surplus.donor.outcome = Winner and partOfSurplus in surplus.distributions and
previous.receiver = surplus.donor and
transferGroups.idxOf[previous] < transferGroups.idxOf[surplus] and
partOfSurplus.ballots in previous.transferred implies
(no other : TransferGroup | some x : Distribution |
x in other.distributions and x.receiver.outcome = Winner and
transferGroups.idxOf[previous] < transferGroups.idxOf[other])

// When candidate is excluded, then all their ballots are available for transfer

all t : transferGroups.elems | some b : Ballot | t.donor in losers and
(b in t.donor.votes or some r : votesReceived | (r.receiver = t.donor and b in r.ballots))
implies ((b in t.nontransferables) or (some d : t.distributions | b in d.ballots))

no disj x,y : transferGroups.elems | x.donor = y.donor

// All shuffles in this scenario are permutations of the same set of ballots

```

all b: Ballot | b in ballots.elems
// The same ballot does not appear twice in the shuffle
not ballots.hasDups
// All seats have been filled
#winner = Election.seats
// No overlap between winners and losers
no c: Candidate | c in winners and c in losers
// No missing candidates
#Candidate = #winners + #losers
all c: Candidate | c in winners or c in losers
// Some outcomes are fixed regardless of shuffle
all w: Candidate | w.outcome = Winner implies w in winners
all c: Candidate | c.outcome = Loser implies c in losers
}

```

```

// The parameters of the election

```

```

one sig Election {

```

```

    seats:    Int,

```

```

    quota:    Int

```

```

} {

```

```

// There is at least one ballot cast in this election

```

```

0 < #Ballot

```

```

// There is at least one seat and more candidates than available seats

```

```

0 < seats and seats < #Candidate

```

```

// Quota is (1 + (ballots / (1 + seats)))

```

```

quota = 1 + div[#Ballot, seats + 1]

```

```

}

```

```

-- End.

```