

Software Review: OpenSTV and OpaVote

James Green-Armytage
j.armytage@gmail.com

Abstract

OpenSTV is a software application that implements several STV rules, as well as a number of other single winner and multi-winner voting rules. OpaVote is an online tally service with similar capabilities.

1 OpenSTV overview and pricing

OpenSTV¹ is a software application that can be used to compute the results of elections, using a variety of voting rules. These rules include a very large number of STV variants, as well as plurality, Borda, a few Condorcet methods, and some others. OpaVote is an online counterpart to OpenSTV, which provides the same menu of voting rules. Both OpenSTV and OpaVote were created by Jeffrey O'Neill,² who holds a Ph.D. in engineering from the University of Michigan and a J.D. from Cornell Law School.

OpenSTV was first released in 2003 and has been updated several times since, in a careful process of development and improvement. From the web site <http://www.openstv.org>, one can download OpenSTV as a self-installing program for either Windows or Mac; the installation program is quick and straightforward.

The primary command in OpenSTV is 'Run Election'. After selecting this command one is prompted to choose a file containing ballot information and a voting rule. Next, one encounters a submenu screen that permits

customization of the voting rule, for example by deciding the number of seats to be filled, the type of quota (Droop or Hare), etc. (The options in this second menu depend on your primary voting rule, and often allow a very high degree of customization.) Once these choices have been made, the program calculates the result, produces a text-based printout that names the winner(s), and provides relevant details about the counting process. For example, in the case of STV rules this includes vote totals as of each round, reports on when candidates are elected or eliminated, etc.

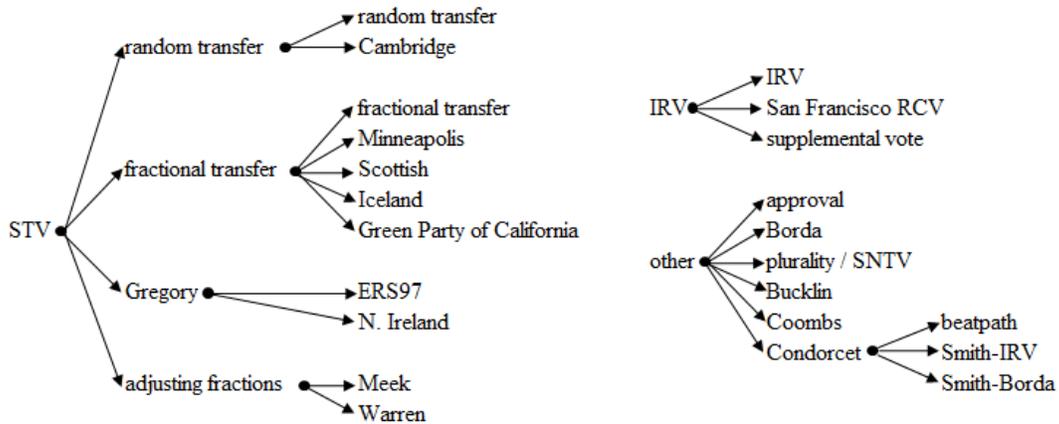
The program allows one to create and edit ballot files, which have .blt file extensions and can also be read with simple text-editing programs. For example, for an election in which paper ballots have been cast, one can enter the information from these ballots one at a time using the menu that follows from the 'New Ballot File' command, save the file, and then apply the desired voting rule using the 'Run Election' command. The ballot file creation/editing process permits ballots with either complete or incomplete rankings of the candidates, and review or alteration of the ballots that have already been entered.

OpenSTV was an open source program from 2003 to 2010 (hence the name), but Dr. O'Neill decided thereafter to change to closed source distribution, in part because of free-riding and a lack of outside contribution to development of the software. Currently, OpenSTV licenses are available on a yearly basis according to a three-tiered pricing structure: Individuals and small non-profits (with 0-3 employees) may buy a license for the minimum price of \$5, medium-sized non-profits (with 4-30 employees) are asked to pay the intermediate price of \$40, and other users (including large non-profits, businesses, and governments) are asked to pay \$400.

¹ The version of OpenSTV examined is 2.1.0.

² Jonathan Lundell and Dan Keshet assisted Dr. O'Neill with the development of the software.

Figure 1: Taxonomy of voting rules implemented in OpenSTV



2 OpaVote overview and pricing

To use OpaVote, one goes to www.opavote.org, logs in with a Google account, and then chooses the ‘My Elections’ tab, where one can create a new election or poll. The difference between these is that voting in an election is by invitation only; when creating the election, one supplies a list of email addresses, and each person on this list receives a link that allows them to vote once. On the other hand, a poll allows anyone to vote; it is possible to restrict people to one vote each (e.g. by using tracking cookies) or simply let them vote as many times as they wish.

Use of OpaVote on a small scale, i.e. with 400 or fewer voters, and 20 or fewer candidates is free; this seems like a good option for people who are interested in familiarizing themselves with the software before making a commitment. Some of OpenSTV’s options for voting rule customization are not available in OpaVote, but the number of choices is still quite large, and thus sufficient for most casual purposes.

An OpaVote election or poll costs five cents for each additional voter beyond 400, and one dollar for each additional candidate beyond 20. Also, if an election or poll remains online for more than two weeks, this costs five dollars per additional two weeks.

3 Voting rules implemented

OpenSTV has an initial menu of twelve rules: approval, Borda, Cambridge STV, Condorcet, ERS97 STV, instant runoff voting, Meek STV, Minneapolis STV, N. Ireland STV, plurality/FPTP/SNTV, San Francisco RCV, and Scottish STV. However, ticking the ‘Show All Methods’ box in the options menu yields eight more choices: Bucklin, Coombs, fractional transfer STV, Green Party of California STV, Iceland STV, random transfer STV, supplementary vote, and Warren STV. So, there are twenty rules to choose from altogether.

Figure 1 arranges OpenSTV’s voting rules by category. The program implements eleven different STV rules, which can be divided into four sub-categories: random transfer, fractional transfer, Gregory, and adjusting fractions.

When a candidate reaches the quota, the random transfer STV rules (Cambridge, and a generic, customizable random transfer rule) transfer the surplus in the form of whole ballots, while leaving the remainder with the original candidate. Which votes are transferred and which votes remain depends on the order in which the votes are entered into the system; therefore, to make the process truly random, the ballots should be shuffled before being input.

Instead of transferring some votes at full value when a candidate has a surplus, the

fractional transfer STV rules (Minneapolis, Scottish, Iceland, Green Party of California, and a generic, customizable fractional transfer rule) transfer all votes at a fractional value. The Gregory STV rules (ERS97 and N. Ireland) operate in much the same manner, except that when one candidate's elimination or surplus transfer brings another candidate above the quota, only these last-received votes are transferred again (which is convenient for hand-counting purposes).

In the adjusting fractions STV methods, Meek and Warren, the fractional part of votes that elected candidates retain is updated throughout the count. Under one of the other STV rules, a tactical voter might indicate a hopeless candidate for his first choice, so that once this candidate is eliminated, his vote will be transferred with its full weight to his favourite candidate who has not yet been elected, as opposed to being reduced in value during the process of being transferred from his sincere favourite candidate(s). Meek and Warren thwart this tactic by bringing this vote to the elected sincere favourite after the hopeless candidate has been eliminated, and recalculating the share of votes that the elected candidate must retain, thus decreasing the weight of the tactical ballot and increasing the weight of the ballots already transferred from that candidate, so that they have the same value. These rules are difficult to implement with a hand count, so their inclusion in OpenSTV should be counted as a major virtue.

In its single-winner case, STV is known as instant runoff voting (IRV), the alternative vote, ranked choice voting, etc. The primary difference between the IRV and STV rules as implemented here is that the former do not allow for the transfer of surplus votes. (That is, OpenSTV does allow one to use IRV to elect more than one seat; this works as a series of elimination-and-transfer rounds that continue until the number of remaining candidates is equal to the number of seats.) OpenSTV gives the user a choice of a generic customizable IRV rule, the San Francisco RCV rule, and the supplemental vote rule, which uses only the first two rankings on the ballot and allows for only two rounds of counting.

In addition to these several STV and IRV rules, OpenSTV gives six additional choices: approval, Borda, plurality, Bucklin, Coombs,

and Condorcet. Of these, all except Bucklin and Condorcet can be used for multi-winner elections.

Approval counts all rankings as 'approvals'; to indicate 'disapproval' of a candidate, a voter should leave that candidate unranked. Each approval earns a candidate one point. Given C candidates, Borda gives a candidate C points for each ballot on which it is ranked first, 0 points for each ballot on which it is ranked last or not ranked at all, and $C - n + 1$ points for each ballot on which it is ranked as the voter's n th choice. Plurality, also known as 'first past the post' (FPTP), or 'single non-transferable vote' (SNTV) when used for multi-winner elections, gives a candidate one point for each ballot on which the candidate is ranked first. For approval, Borda, or plurality, with S seats to be filled, the winners are the S candidates with the most points.

Coombs works like IRV, except that it eliminates the candidate with the most last-choice votes instead of the fewest first-choice votes (with unranked candidates being treated as tied for last place). As with the IRV implementation here, Coombs continues with its eliminations until the number of remaining candidates matches the number of seats. Bucklin first checks to see whether a majority of voters rank any candidate in first place, then checks whether a majority of voters rank any candidate in first or second place, and so on, until such a candidate is found.

Finally, the 'Condorcet Voting' option provides the choice of three interesting Condorcet-efficient rules, all of which choose a candidate from the Smith set (also known as the minimal dominant set, i.e. the smallest set of candidates such that every candidate inside the set is ranked above every candidate outside the set by a majority of voters). The first of these is referred to as 'Schwartz sequential dropping', and is also known as 'beatpath' or 'Schulze'. (See Markus Schulze's article in issue 17 of *Voting matters*.) Since the OpenSTV version of this rule calculates and explains the results using a beatpath matrix, the second of these names seems most descriptive to me. The second Condorcet-efficient rule that one may choose eliminates candidates not in the Smith set, and then performs an IRV tally on the remaining candidates. (My article in issue 29 of *Voting matters* examines this along with three

other very similar methods, and concludes that they share attractive qualities, such as an unusually high resistance to both strategic voting and strategic nomination.) The third Condorcet-efficient rule eliminates candidates not in the Smith set, and then performs a Borda tally on the remaining candidates. (This is similar to the ‘Black’ rule, except that it eliminates candidates outside the Smith set before performing its Borda tally in the case of a majority rule cycle.)

4 Other features and remarks

4.1 Graphing election results

While OpenSTV presents election results in a primarily text-based form, OpaVote includes colour-coded bar charts showing the vote totals for each candidate as of each round of counting. The latter presentation can also be generated by OpenSTV using the following steps: First, run an election, proceeding through to the text-based display of the results. Second, from the file menu, choose to ‘Save Results as HTML...’, indicate a file name, and hit ‘Save’. Third, add an extension of ‘.html’ or ‘.htm’ to the name of the resulting file. Now, it should open in a web browser and display the results as OpaVote does.

4.2 Generating illustrative examples

In order to better understand the properties of different voting rules, it’s often helpful to devise ballot profiles that illustrate how their results diverge in particular cases. Usually these examples are described in a manner such as ‘6 voters prefer D, then A, then B, then C; 5 voters prefer B, then C, then A, then D; 4 voters prefer

C, then A, then B, then D.’ Happily, it is very easy to create examples like this in such a way that they can be tallied by OpenSTV. For example, the situation above can be captured by simply creating a text (.txt) file with nothing but the following content:

```
6: D A B C
5: B C A D
4: C A B D
```

OpenSTV will recognize this as a valid ballot file, and count an election using this set of votes.

4.3 Ballot files from public elections

At present, the OpenSTV web site houses ballot files for about fifty public elections that have used ranked ballots and taken place over the last several years, e.g. in San Francisco, Scotland, Minneapolis, etc. These can be found via the ‘Ranked-Choice Voting’ tab, saved, and then run with OpenSTV.

Organizations using OpenSTV and OpaVote: At <http://www.openstv.org/openstv-users>, there is a long list of forty or so organizations that have used OpenSTV or OpaVote, including several non-profits, universities/student unions, etc.

5 Conclusion

Dr. O’Neill has an impressive, scholarly knowledge of the different STV variants, and he has clearly taken great care in applying this knowledge to the creation of a program that is accessible and user-friendly. The result is a valuable contribution for those who are interested in studying ranked ballot voting rules, and implementing them in practice.