

Ranked pairs

Ranked pairs (RP) or the **Tideman method** is an electoral system developed in 1987 by [Nicolaus Tideman](#) that selects a single winner using votes that express preferences.^{[1][2]} RP can also be used to create a sorted list of winners.

If there is a candidate who is preferred over the other candidates, when compared in turn with each of the others, RP guarantees that candidate will win. Because of this property, RP is, by definition, a [Condorcet method](#).

Contents

Procedure

Tally

Sort

Lock

Winner

Tied majorities

An example

The situation

Tally

Sort

Lock

Summary

Criteria

Independence of irrelevant alternatives

Comparison table

Notes

References

External links

Procedure

The RP (Ranked Pair) procedure is as follows:

1. Tally the vote count comparing each pair of candidates, and determine the winner of each pair (provided there is not a tie)

2. Sort (rank) each pair, by the largest **strength of victory** first to smallest last.^{[[vs 1](#)]}

3. "Lock in" each pair, starting with the one with the largest number of winning votes, and add one in turn to a graph as long as they do not create a **cycle** (which would create an ambiguity). The completed graph shows the winner.

The procedure can be illustrated using a simple example. Suppose that there are 27 voters and 4 candidates *w*, *x*, *y* and *z* such that the votes are cast as shown in the table of ballots.

| Ballots table | |
|---|-----------|
| <i>w</i> – <i>x</i> – <i>z</i> – <i>y</i> | 7 ballots |
| <i>w</i> – <i>y</i> – <i>x</i> – <i>z</i> | 2 ballots |
| <i>x</i> – <i>y</i> – <i>z</i> – <i>w</i> | 4 ballots |
| <i>x</i> – <i>z</i> – <i>w</i> – <i>y</i> | 5 ballots |
| <i>y</i> – <i>w</i> – <i>x</i> – <i>z</i> | 1 ballots |
| <i>y</i> – <i>z</i> – <i>w</i> – <i>x</i> | 8 ballots |

Tally

| | <i>w</i> | <i>x</i> | <i>y</i> | <i>z</i> |
|----------|----------|----------|----------|----------|
| <i>w</i> | – | 9 | 1 | –7 |
| <i>x</i> | –9 | – | 5 | 11 |
| <i>y</i> | –1 | –5 | – | 3 |
| <i>z</i> | 7 | –11 | –3 | – |

The vote tally can be expressed as a table in which the (*w*,*x*) entry is the number of ballots in which *w* comes higher than *x* **minus** the number in which *x* comes higher than *w*. In the example *w* comes higher than *x* in the first two rows and the last two rows of the ballot table (total 18 ballots) while *x* comes higher than *w* in the middle two rows (total 9), so the entry in the (*w*,*x*) cell is 18–9 = 9.

Notice the **skew symmetry** of the table.

Sort

| Sorted majorities | | | | | |
|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|
| (<i>x</i> , <i>z</i>) | (<i>w</i> , <i>x</i>) | (<i>z</i> , <i>w</i>) | (<i>x</i> , <i>y</i>) | (<i>y</i> , <i>z</i>) | (<i>w</i> , <i>y</i>) |
| 11 | 9 | 7 | 5 | 3 | 1 |

The positive majorities are then sorted in decreasing order of magnitude:

Lock

After two steps

| | <i>w</i> | <i>x</i> | <i>y</i> | <i>z</i> |
|----------|----------|----------|----------|----------|
| <i>w</i> | 0 | 1 | 0 | 1 |
| <i>x</i> | –1 | 0 | 0 | 1 |
| <i>y</i> | 0 | 0 | 0 | 0 |
| <i>z</i> | –1 | –1 | 0 | 0 |

The next stage is to examine the majorities in turn to determine which pairs to "lock in". This can be done by building up a matrix in which the (*x*,*y*) entry is initially 0, and is set to 1 if we decide that *x* is preferred to *y* and to –1 if we decide that *y* is preferred to *x*. These preferences are decided by the list of sorted majorities, simply skipping over any which are inconsistent with previous decisions.

The first two majorities tell us that *x* is preferred to *z* and *w* to *x*, from which it follows by **transitivity** that *w* is preferred to *z*. Once these facts have been incorporated into the table it takes the form as shown after two steps. Notice again the skew symmetry.

The third majority tells us that *z* is preferred to *w*, but since we have already decided that *w* is preferred to *z* we ignore it.

The fourth majority tells us that x is preferred to y , and since we know that w is preferred to x we infer that w is preferred to y , giving us the table after 4 steps.

After four steps

| | w | x | y | z |
|---|----|----|---|---|
| w | 0 | 1 | 1 | 1 |
| x | -1 | 0 | 1 | 1 |
| y | -1 | -1 | 0 | 0 |
| z | -1 | -1 | 0 | 0 |

After five steps

| | w | x | y | z |
|---|----|----|----|---|
| w | 0 | 1 | 1 | 1 |
| x | -1 | 0 | 1 | 1 |
| y | -1 | -1 | 0 | 1 |
| z | -1 | -1 | -1 | 0 |

The fifth majority tells us that y is preferred to z , and this completes the table.

Winner

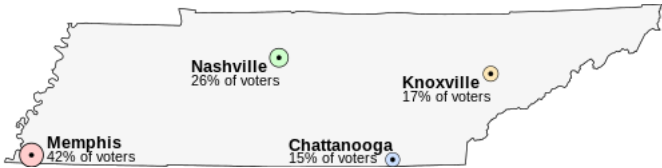
In the resulting graph for the locked pairs, the source corresponds to the winner. In this case w is preferred to all other candidates and is therefore identified as the winner.

Tied majorities

In the example the majorities are all different, and this is what will usually happen when the number of voters is large. If ties are unlikely, then it does not matter much how they are resolved, so a random choice can be made. However this is not Tideman's procedure, which is considerably more complicated. See his paper for details.^[1]

An example

The situation



Imagine that Tennessee is having an election on the location of its capital. The population of Tennessee is concentrated around its four major cities, which are spread throughout the state. For this example, suppose that the entire electorate lives in these four cities and that everyone wants to live as near to the capital as possible.

The candidates for the capital are:

- Memphis, the state's largest city, with 42% of the voters, but located far from the other cities
- Nashville, with 26% of the voters, near the center of the state
- Knoxville, with 17% of the voters
- Chattanooga, with 15% of the voters

The preferences of the voters would be divided like this:

| 42% of voters (close to Memphis) | 26% of voters (close to Nashville) | 15% of voters (close to Chattanooga) | 17% of voters (close to Knoxville) |
|---|---|---|---|
| 1. Memphis 2. Nashville 3. Chattanooga 4. Knoxville | 1. Nashville 2. Chattanooga 3. Knoxville 4. Memphis | 1. Chattanooga 2. Knoxville 3. Nashville 4. Memphis | 1. Knoxville 2. Chattanooga 3. Nashville 4. Memphis |

The results would be tabulated as follows:

Pairwise election results

| | | A | | | |
|--|-------------|--------------------|--------------------|--------------------|--------------------|
| | | Memphis | Nashville | Chattanooga | Knoxville |
| B | Memphis | | [A] 58% [B] 42% | [A] 58% [B] 42% | [A] 58% [B] 42% |
| | Nashville | [A] 42% [B] 58% | | [A] 32% [B] 68% | [A] 32% [B] 68% |
| | Chattanooga | [A] 42% [B] 58% | [A] 68% [B] 32% | | [A] 17% [B] 83% |
| | Knoxville | [A] 42% [B] 58% | [A] 68% [B] 32% | [A] 83% [B] 17% | |
| Pairwise election results (won-lost-tied): | | 0-3-0 | 3-0-0 | 2-1-0 | 1-2-0 |
| Votes against in worst pairwise defeat: | | 58% | N/A | 68% | 83% |

- [A] indicates voters who preferred the candidate listed in the column caption to the candidate listed in the row caption
- [B] indicates voters who preferred the candidate listed in the row caption to the candidate listed in the column caption

Tally

First, list every pair, and determine the winner:

| Pair | Winner |
|---------------------------------------|------------------|
| Memphis (42%) vs. Nashville (58%) | Nashville 58% |
| Memphis (42%) vs. Chattanooga (58%) | Chattanooga 58% |
| Memphis (42%) vs. Knoxville (58%) | Knoxville 58% |
| Nashville (68%) vs. Chattanooga (32%) | Nashville 68% |
| Nashville (68%) vs. Knoxville (32%) | Nashville 68% |
| Chattanooga (83%) vs. Knoxville (17%) | Chattanooga: 83% |

Note that absolute counts of votes can be used, or percentages of the total number of votes; it makes no difference since it is the ratio of votes between two candidates that matters.

Sort

The votes are then sorted. The largest majority is "Chattanooga over Knoxville"; 83% of the voters prefer Chattanooga. Thus, the pairs from above would be sorted this way:

| Pair | Winner |
|---------------------------------------|-----------------|
| Chattanooga (83%) vs. Knoxville (17%) | Chattanooga 83% |
| Nashville (68%) vs. Knoxville (32%) | Nashville 68% |
| Nashville (68%) vs. Chattanooga (32%) | Nashville 68% |
| Memphis (42%) vs. Nashville (58%) | Nashville 58% |
| Memphis (42%) vs. Chattanooga (58%) | Chattanooga 58% |
| Memphis (42%) vs. Knoxville (58%) | Knoxville 58% |

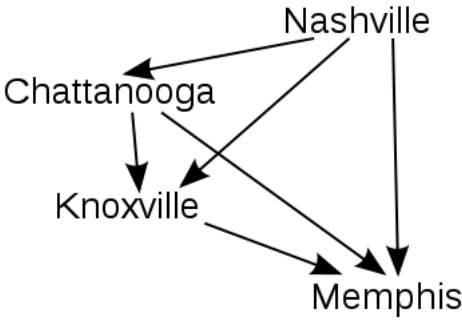
Lock

The pairs are then locked in order, skipping any pairs that would create a cycle:

- Lock Chattanooga over Knoxville.
- Lock Nashville over Knoxville.
- Lock Nashville over Chattanooga.
- Lock Nashville over Memphis.
- Lock Chattanooga over Memphis.
- Lock Knoxville over Memphis.

In this case, no cycles are created by any of the pairs, so every single one is locked in.

Every "lock in" would add another arrow to the graph showing the relationship between the candidates. Here is the final graph (where arrows point away from the winner).



In this example, Nashville is the winner using RP, followed by Chattanooga, Knoxville, and Memphis in second, third, and fourth places respectively.

Summary

In the example election, the winner is Nashville. This would be true for any Condorcet method.

Using the First-past-the-post voting and some other systems, Memphis would have won the election by having the most people, even though Nashville won every simulated pairwise election outright. Using Instant-runoff voting in this example would result in Knoxville winning even though more people preferred Nashville over Knoxville.

Criteria

Of the formal voting criteria, the ranked pairs method passes the majority criterion, the monotonicity criterion, the Smith criterion (which implies the Condorcet criterion), the Condorcet loser criterion, and the independence of clones criterion. Ranked pairs fails the consistency criterion and the participation criterion. While ranked pairs is not fully independent of irrelevant alternatives, it still satisfies local independence of irrelevant alternatives.

Independence of irrelevant alternatives

Ranked pairs fails independence of irrelevant alternatives. However, the method adheres to a less strict property, sometimes called independence of Smith-dominated alternatives (ISDA). It says that if one candidate (X) wins an election, and a new alternative (Y) is added, X will win the election if Y is not in the Smith set. ISDA implies the Condorcet criterion.

Comparison table

The following table compares Ranked Pairs with other preferential single-winner election methods:

| Comparison of preferential electoral systems | | | | | | | | | | | | |
|--|------------|------------------|----------|-----------------|----------------|-----------------|-------|------|------|------------------------|-------------------|----------------------------|
| System | Mono-tonic | Condorcet winner | Majority | Condorcet loser | Majority loser | Mutual majority | Smith | ISDA | LIIA | Independence of clones | Reversal symmetry | Participation, consistency |
| <u>Schulze</u> | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No | Yes | Yes | No |
| <u>Ranked pairs</u> | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No |
| <u>Tideman's Alternative</u> | No | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No | Yes | No | No |
| <u>Kemeny–Young</u> | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No | Yes | No |
| <u>Copeland</u> | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No | No | Yes | No |
| <u>Nanson</u> | No | Yes | Yes | Yes | Yes | Yes | Yes | No | No | No | Yes | No |
| <u>Black</u> | Yes | Yes | Yes | Yes | Yes | No | No | No | No | No | Yes | No |
| <u>Instant-runoff voting</u> | No | No | Yes | Yes | Yes | Yes | No | No | No | Yes | No | No |
| <u>Smith/IRV</u> | No | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No | Yes | No | No |
| <u>Borda</u> | Yes | No | No | Yes | Yes | No | No | No | No | No | Yes | Yes |
| <u>Baldwin</u> | No | Yes | Yes | Yes | Yes | Yes | Yes | No | No | No | No | No |
| <u>Bucklin</u> | Yes | No | Yes | No | Yes | Yes | No | No | No | No | No | No |
| <u>Plurality</u> | Yes | No | Yes | No | No | No | No | No | No | No | No | Yes |
| <u>Contingent voting</u> | No | No | Yes | Yes | Yes | No | No | No | No | No | No | No |
| <u>Coombs</u> ^[3] | No | No | Yes | Yes | Yes | Yes | No | No | No | No | No | No |
| <u>MiniMax</u> | Yes | Yes | Yes | No | No | No | No | No | No | No | No | No |
| <u>Anti-plurality</u> ^[3] | Yes | No | No | No | Yes | No | No | No | No | No | No | Yes |
| <u>Sri Lankan contingent voting</u> | No | No | Yes | No | No | No | No | No | No | No | No | No |
| <u>Supplementary voting</u> | No | No | Yes | No | No | No | No | No | No | No | No | No |
| <u>Dodgson</u> ^[3] | No | Yes | Yes | No | No | No | No | No | No | No | No | No |

Notes

1. In fact, there are different ways how the strength of a victory is measured. This article uses Tideman's original method based on margins of victory.

References

1. Tideman, T. N. (1987-09-01). "Independence of clones as a criterion for voting rules" (<https://doi.org/10.1007/BF00433944>). *Social Choice and Welfare*. 4 (3): 185–206. doi:10.1007/BF00433944 (<https://doi.org/10.1007%2FBF00433944>). ISSN 1432-217X (<https://www.worldcat.org/issn/1432-217X>).

2. Schulze, Markus (October 2003). "A New Monotonic and Clone-Independent Single-Winner Election Method" (<https://web.archive.org/web/20200711055629/www.votingmatters.org.uk/ISSUE17/INDEX.HTM>). *Voting matters* (www.votingmatters.org.uk). McDougall Trust. 17. Archived from the original (<http://www.votingmatters.org.uk/ISSUE17/INDEX.HTM>) on 2020-07-11. Retrieved 2021-02-02.

3. Anti-plurality, Coombs and Dodgson are assumed to receive truncated preferences by apportioning possible rankings of unlisted alternatives equally; for example, ballot $A > B = C$ is counted as $\frac{1}{2} A > B > C$ and $\frac{1}{2} A > C > B$. If these methods are assumed not to receive truncated preferences, then later-no-harm and later-no-help are not applicable.

External links

- Descriptions of ranked-ballot voting methods (<https://accuratedemocracy.com/download/software/rbvote/desc.html>) by Rob LeGrand
- Example JS implementation (<https://gist.github.com/asafh/a8e9af7a3e5282cbb27>) by Asaf Haddad
- Pair Ranking Ruby Gem (<https://bitbucket.org/bparanj/rasam>) by Bala Paranj
- A margin-based PHP Implementation of Tideman's Ranked Pairs (<https://github.com/pivot-libre/tideman/>)

Retrieved from "https://en.wikipedia.org/w/index.php?title=Ranked_pairs&oldid=1089131648"

This page was last edited on 22 May 2022, at 03:06 (UTC).

Text is available under the Creative Commons Attribution-ShareAlike License 3.0; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.