# Refinements and Randomised Versions of Some Tournament Solutions

# (Extended Abstract)


Justin Kruger
Université Paris-Dauphine
PSL Research University,
Paris, France
justin.g.kruger@gmail.com

Stéphane Airiau
Université Paris-Dauphine
PSL Research University,
Paris, France
stephane.airiau@dauphine.fr



## ABSTRACT

We consider voting rules that are based on the majority graph. Such rules typically output large sets of winners. Our goal is to investigate a general method which leads to refinements of such rules. In particular, we use the idea of parallel universes, where each universe is connected with a permutation over alternatives. The permutation allows us to construct resolute voting rules (i.e. rules that always choose unique winners). Such resolute rules can be constructed in a variety of ways: we consider using binary voting trees to select a single alternative. In turn this permits the construction of neutral rules that output the set the possible winners of every parallel universe. The question of which rules can be constructed in this way has already been partially studied under the heading of agenda implementability. We further propose a randomised version in which the probability of being the winner is the ratio of universes in which the alternative wins. We also investigate (typically novel) rules that elect the alternatives that have maximal winning probability. These rules typically output small sets of winners, thus provide refinements of known tournament solutions.



## Keywords

tournament; probabilistic rule; refinement; Condorcet consistency


## 1. INTRODUCTION

Social choice theory is mainly concerned with deterministic functions that aid in the selection of alternatives. These functions have as input some representation of social preferences; some examples of possible outputs are single alternatives, sets of alternatives, orderings over alternatives, and lotteries of alternatives. We use the term "rules" to refer in general to such deterministic functions. Here we investigate a general method for refining and probabilising rules that can be expressed in terms of parallel universes. Though the method is general, we specifically apply it to functions that take tournaments as inputs.

In mathematical terms a tournament is a complete directed graph. Rules based on tournaments include, but are not limited to, Copeland, the Top Cycle, the Banks set, and the Markov solution concept. We make the following slightly imprecise observation: rules based on tournaments are not very discriminating, both because they often output non-singleton sets, and because in this case the output sets tend to be relatively large. Rules based on tournaments are thus good candidates for rules to be made more discriminating.

After all, the whole purpose of a rule is to provide choice. For practical purposes, singleton outputs are often necessary, and in many cases singleton outputs are required for theoretical analyses of rules. At the same time, it is desirable that these choices are good. The typical approach to an irresolute rule is to apply an exogenous tie-breaker, though this violates the basic desirable property of neutrality. Our approach explicitly retains neutrality while providing some gains in the discriminating power of a rule. It is an interesting issue as to whether other desirable properties, such as monotonicity, can also be retained.


**Appears in:** *Proc. of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*



### 1.1 Previous work

There is a large literature on tournaments, ranging from the purely mathematical [13] to the explicitly social choice theoretic [11] [4]. The particular tournament rules that we are interested in concern tree structures, of which one strand of study originates from questions concerning voting by agenda. An early significant result is the implementation of what became known as the Banks solution concept [12, 1]. This in turn lead to more general questions of implementability: attempts to find the rules implementable by individual resolute trees were given by Srivastava and Trick [14] and continued by Trick [16]. The question of implementability for resolute procedures was settled by Horan [8], who went on to give full sufficient and necessary conditions for irresolute tournament rules [9].

The Banks set was the subject of an interesting exchange concerning the gap in computational complexity between calculating *all* and calculating *some* winners: Woeginger [17] showed that the full set is NP-hard, to which Hudry [10] replied, noting there is a greedy algorithm for calculating some Banks winner. These results are implicitly based upon the "parallel" nature of the Banks set. Such implicit treatment of the idea of parallel computations can be traced back to Tideman's Ranked Pair rule [15], but seems to have been first explicitly called such by Conitzer et al. [6] who investigate the rule variously known as instant run-off voting, single transferable vote or alternative vote. Freeman et al. [7] continued this study in a similar direction, while Brill and Fischer [5] applied a similar explicit treatment of parallel universes to Ranked Pairs.

Randomised rules in social choice have been studied in particular with reference to their ability to mitigate the Gibbard-Satterthwaite result [2]. More pertinent here is recent work by Brandt et al. [3] which characterises a randomised tournament solution that returns maximal lotteries.

## 2. DEFINITIONS

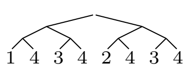
All our definitions are given with reference to a set of alternatives $A = \{1, 2, \ldots, m\}$. We refer to arbitrary elements of $A$ as $a$, $b$, $c$,...

A tournament is a trichotomous and asymmetric relation $T \subset A \times A$. We write $T_{\langle a,b\rangle}$ for the tournament where the relation between $a$ and $b$ is reversed.

A tournament function is a function from the set of all tournaments to non-empty subsets of $A$. There are various properties of tournament functions and rules more generally that are studied; these may ensure fairness or be seen as generally desirable properties for a rule. A tournament function is resolute if it always outputs a singleton. Resoluteness is incompatible with neutrality: a tournament function is neutral if permuting the elements in an input tournament results in a similarly permuted output. A desirable property is monotonicity, where if $aTb$ and $b$ is selected by the rule applied to $T$, then $b$ should also be selected by the rule applied to $T_{\langle a,b\rangle}$. Another potentially desirable property is Condorcet consistency, that requires that if for some alternative $a$, $aTb$ for all alternatives $b \neq a$, then $a$ should be the unique winner.

Many tournament functions can be defined using binary trees with labelled leaves [8]. Specifically, the leaves are labelled with alternatives in $A$. The tournament function determined by such a binary tree proceeds as follows. Starting at the leaves, pairs of children are compared according to the input tournament. Assuming the children are distinct, the alternative that dominates the other is placed at the parent node – if the children are identical i.e. the same alternative, this alternative is placed at the parent node. This procedure is continued up the tree and the alternative at the root is the output of the tournament function.

We can compactly represent trees as left associative words. For an illustrative example, the Banks tree, whose recursive definition is given below, is drawn below as a full tree next to its compact form.




14(34)(24(34))

Here are some examples of recursive definitions for such trees, which can be used to create tournament functions for sets of any finite cardinality:

**Simple tree** $\mathsf{st}(1, 2, \ldots, n) = 1\ (\mathsf{st}(2, \ldots, n))$

**Two-leaf tree** $\mathsf{tt}(1, 2, \ldots) = 12$.

**Balanced tree**

$$\mathsf{ft}(1, \ldots, \left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil + 1, \ldots, n) = \mathsf{ft}(1, \ldots, \left\lceil \frac{n}{2} \right\rceil)\ \Big(\ \mathsf{ft}(\left\lceil \frac{n}{2} \right\rceil + 1, \ldots, n)\ \Big)$$

**Banks tree** $\mathsf{bt}(1, 2, \ldots, n) = \mathsf{bt}(1, 3, \ldots, n)\ (\mathsf{bt}(2, 3, \ldots, n))$

**Iterative Condorcet tree** $\mathsf{it}(1, 2, \ldots, n) = 12 \ldots n\ \mathsf{it}(2, \ldots n)$

All basic tree based rules are resolute, thus none of the above produce neutral rules (for $|A| > 2$). We extend a basic rule based on a tree in three ways to create three new rules that we call parallel universe rules, argmax rules and frequency rules. Parallel universe rules output the set of winners for all possible permutations of the alternatives applied to the leaves of the tree. For the argmax version, we count how many times a permutation of the alternatives leads to each alternative being the winner. The alternatives that win for most permutations are the argmax winners. Finally, frequency rules output a lottery over the alternatives determined by how often a permutation produces each alternative.

## 3. PROPERTIES AND COMPARISON

All of our three extensions are neutral, thus satisfy a basic fairness notion that is broken when applying an exogenous tie-breaker. It is an interesting question to what extent other desirable properties are satisfied by these rules. There are various properties that depend upon the structure of the trees involved: whether repetitions are allowed in the leaves, or whether the set of leaves is a superset of the alternatives. For instance, it is easy to see that a rule is Condorcet consistent if the set of leaves is a superset of the alternatives, for the resolute function and the parallel and argmax versions.

We would like to note that although our extensions are applied to the specific case of tournament rules, they can generally be applied to any rules with a "parallel" nature. From this perspective, the more interesting question concerns inheritance results: that is, if the basic rule satisfies some property, then do its extended versions also satisfy this property. For instance, it is easy to see that monotonicity is inherited by parallel universe versions of rules, and that a (weak) probabilistic version is inherited by frequency rules. The unresolved question is whether monotonicity is inherited by argmax rules.

### 3.1 Success of argmax versions as refinements

Aside from the (open) question of monotonicity, we are interested in how effective argmax rules are at refining the set of winners. We have tested this on some example tournaments. The outcome of all of our rules only concern alternatives in the Top Cycle: any Condorcet losers will not affect the outcome of the vote. Thus we restrict attention to what are called non-reducible tournaments, those in which the whole tournament is returned by the Top Cycle. Moon [13] provides a list of all non-isomorphic small tournaments, from which we see that there are only 34 non-reducible tournaments of size 6. We applied our rules to all of these, and compared them with the Markov solution concept, generally considered among the most discriminating tournament solutions. Of these 34 tournaments, the Banks set contains three alternatives 14 times, four alternatives 8 times, five alternatives 9 times and six alternatives 3 times. In contrast the argmax rule applied to the Banks tree outputs a single winner 32 times, two winners 1 time and three winners 2 times. Argmax applied to the Simple tree and the Iterative Condorcet tree get similar (though distinct) results. Copeland, often considered a fairly discriminating solution concept, which is here equivalent to argmax applied to the Two-leaf tree, outputs a single winner 18 times, two winners 7 times, three winners 5 times and four winners 4 times. Thus for these tournaments the argmax rules are significantly more discriminating than the full parallel universe versions.

## REFERENCES


[1] Jeffrey S Banks. Sophisticated voting outcomes and agenda control. *Social Choice and Welfare*, 1(4):295–306, 1985.

[2] Salvador Barbera. Majority and positional voting in a probabilistic framework. *The Review of Economic Studies*, 46(2):379–389, 1979.

[3] Florian Brandl, Felix Brant, and Hans G Seedig. Consistent probabilistic social choice. *Econometrica: Journal of the Econometric Society*, 84(5):1839–1880, 2016.

[4] Felix Brandt, Markus Brill, and Paul Harrenstein. Tournament solutions. In *Handbook of Computational Social Choice*, chapter 3. Cambridge University Press, 2016.

[5] Markus Brill and Felix Fischer. The price of neutrality for the ranked pairs method. In *Proceedings of AAAI-12*, pages 1299–1305, 2012.

[6] Vincent Conitzer, Matthew Rognlie, and Lirong Xia. Preference functions that score rankings and maximum likelihood estimation. In *Proceedings of IJCAI-09*, pages 109–115, 2009.

[7] Rupert Freeman, Markus Brill, and Vincent Conitzer. General tiebreaking schemes for computational social choice. In *Proceedings of AAMAS-15*, 2015.

[8] Sean Horan. Implementation by agenda voting. *COMSOC 2012*, pages 239–250, 2012.

[9] Sean Horan. Implementation of majority voting rules. *preprint*, 2013.

[10] Olivier Hudry. A note on "Banks winners in tournaments are difficult to recognize" by G. J. Woeginger. *Social Choice and Welfare*, 23(1):113–114, 2004.

[11] Jean-François Laslier. *Tournament Solutions and Majority Voting*. Springer, 1997.

[12] Nicholas R Miller. Graph-theoretical approaches to the theory of voting. *American Journal of Political Science*, pages 769–803, 1977.

[13] John W Moon. *Topics on Tournaments in Graph Theory*. Holt, Rinehart and Winston, 1968.

[14] Sanjay Srivastava and Michael A Trick. Sophisticated voting rules: The case of two tournaments. *Social Choice and Welfare*, 13(3):275–289, 1996.

[15] T Nicolaus Tideman. Independence of clones as a criterion for voting rules. *Social Choice and Welfare*, 4(3):185–206, 1987.

[16] Michael A Trick. Small binary voting trees. 2006.

[17] Gerhard. J. Woeginger. Banks winners in tournaments are difficult to recognize. *Social Choice and Welfare*, 20(3):523–528, 2003.