

Collective Constraint Optimization

Alexander Schiendorfer, Guido Tack

June 27, 2021

1 Big Picture

We believe that in collective constraint optimization, there are at least these classes of objectives (and constraints) to consider:

- “Classical objective” objectives such as costs, **efficiency**, time, makespans etc.
- Balance/Fairness/Equity: comparable criteria such as nightshifts per month, distance to closest school (shops), fair allocation of resources e.g. matching internships,
 - Equitability typically requires some sort of anonymity: Everyone pays 20% tax whereas balance uses progressive/regressive taxation
 - Be aware that, e.g., the Shapley value characterizes a “fair” payment for an agent based on their marginal utility to a coalition of agents. If I add high profits to a group, I get a high payoff. We would consider this proportionality fair and balancing.
- Opinions, preferences that are hard to quantify or even if so, they are hard to objectively compare (e.g., maximize sum of utilities?). “I like a schedule more than another one if monday starts at noon”, “I’d rather have a direct connection to the main station than many adjacent small lines.” Voting probably makes most sense for such cases.

Fundamentally, we might distinguish harshly between repeated optimization (e.g., online) and single-shot-decisions. This applies to basically all classes of objectives mentioned above.

Most importantly, we envision approaches/algorithms that combine these objectives, depending on the specific problem at hand. For instance, as a case study, consider the problem of designing tour plans to sites of interest for a travel company responsible for a group of people. We can only decide a number of tours, not have each tour be completely personalized for an individual. Assume that there are constraints on availability of the sites (time windows), the possible travel time (e.g., within a day), minimal number of visits (over a year) to each of the sites, driving costs, etc. We can instantiate the above classes of objectives:

- There are obvious monetary costs involved. A travel company might simply be interested in maximizing their revenue. Simply restricting that, e.g., to the most expensive sites (or those with my largest margins) and

ignoring my customers' preferences may not pay off over the long term (I lose customers by means of lousy reviews). Said aspect incentivizes a company to even consider having the customers vote over tour plans. A driving cost factor could be the number of buses needed with respect to both capacity constraints and consumer satisfaction.

- With respect to balancing and fairness, we see two possible goals: i) regarding our tour guides or sites, we might have to ensure that the workload (i.e., the rosters) is spread evenly perhaps over multiple time periods; ii) for our travelers, there could be comparable metrics such as the number of feasible sites (e.g., a handicapped person should not have to only choose among incompatible options; similarly, a family could not be able to enter a “brutal” museum).
- There are hard-to-quantify preferences such as “I’d rather pick site a than b but this holds only if person C joins me” or “I want to be at Golden Gate only in the morning - I don’t care about the afternoon”. Such conflicts need to be resolved via voting since the outcome should be an acceptable compromise for the whole group.

The interplay leads to several possible algorithmic instantiations. CP, in general, helps us to identify only (objectively) meaningful solutions, respecting other objective functions as well. Voting takes precedence when we fail to propose other decision criteria:

- A travel agency might offer only plans that achieve a certain amount of strategic satisfaction. (I should not be excessive – “Kaffeefahrt”)
- Resource limitations (e.g., the number of available buses or the maximal budget for all sites, the tour guides we need) could be stated as constraints or be an objective.
- For comparable metrics concerning balance and equity, rawlsian approaches could be needed (assure that the minimal dissatisfaction is guaranteed).
- For repeated situations, learning about distributions of preferences could help us to properly judge a single occurrence of the optimization problem:
 - The preferences of this group are abnormally “bad” for some other-wise popular choice of site. Don’t force them to take it - we’ll find other groups.
 - This group likes a site that is usually not very popular. Make sure to have them pick it if it makes sense for us strategically, i.e., to meet our quota.

In summary, one very simple algorithm could proceed as follows:

1. Find out what optima (costs) are even possible. Define an upper acceptable bound.
2. Define and specify suitable minimal equity/balance constraints. Communicate them effectively to the users.

3. Optimize according to a social choice function defined by the users' (incomparable) preferences. Pick a choice set C that is quite likely to be equivalent in terms of these preferences (e.g., all approximate Copeland winners).
4. Optimize according to your original objectives (costs) to find the best option among all that won the voting.

2 Mission Statement

Our core idea is to make voting theory, in particular social choice functions (SCF), amenable to constraint optimization. Why is that even useful and where do we need it? And why is it not straightforward?

2.1 Preliminaries for Voting

Let $A = \{1, \dots, m\}$ be a finite set of m alternatives and $N = \{1, \dots, n\}$ be a set of n voters. Each voter i has a linear ordering \preceq_i over A . A social welfare function (SWF) takes these n linear orderings over A and produces one (collective) welfare ordering \preceq over A . An SCF returns (a set of) winners. Let $V \subseteq A$ be the set of *feasible* alternatives which can actually be chosen. The majority relation $>^\mu$ expresses that a majority of agents prefers one alternative over the other: $a >^\mu b$ iff. $|\{i \in N : a \prec_i b\}| > |\{i \in N : b \prec_i a\}|$.

Ideally, we would pick a *Condorcet winner*, i.e., a solution that wins all pairwise competitions. Such a winner needs not exist, i.e., there can be cycles (Condorcet paradox). The larger A , the less likely we will get one. But still, the majority relation gives rise to a variety of extensions:

- A Copeland winner is an alternative that wins most pairwise comparisons (duels). A Condorcet winner wins *all* $n - 1$ duels.
- (Symmetric) Borda counting takes the magnitudes (the *net preference*), i.e., the weights on the duel edges, into account.

A tournament $(A, >^\mu)$ is a directed (possibly weighted) graph with alternatives as nodes and the majority relation as edges. Various *solution* concepts to a tournament implement an SCF and return a set $C(A) \subseteq A$ as their choice. Simple examples are *all alternatives* (boring) or all Condorcet non-losers, all Condorcet winners, all Copeland winners, etc.

For incomplete information about the set of alternatives: Let KA be set of known available alternatives (i.e., they are feasible), KU the set of known unavailable alternatives and U the set of alternatives with unknown status that get inserted into KA or KU over time. At all times, these sets are disjoint and in union result in A . Note that we assume V to be constant, i.e., the true feasible set. We further have:

$$KA \subseteq V \subseteq KA \uplus U = A \setminus KU$$

Initially, we are voting over alternatives in $KA \cup U$. We assume that our individual preferences are invariant to loss of candidates over the course of one execution of the algorithm.

2.2 Motivation

- *Why is it useful?*: Voting addresses the aggregation of somewhat unclear preference statements, situations where we have no crisp numeric utility. That includes preferences obtained from learning systems (ML produces a score, given a solution), interactive systems or preference degrees on different scales. In such cases, *we explicitly give up numeric comparability*. There is no transferable utility assumption and we accept the weaknesses of ordinal voting systems (ignorance of magnitudes etc.). But our method is applicable in such scenarios and we need to point them out.
- *Why is it not straightforward?*: In principle, we could run a naive approach precalculating V , i.e., all solutions, have them ranked by our agents and apply an SCF to determine the winner(s). However, in practice, V can either be too large for i) a human user to rank it or even ii) produce all solutions and have a machine autonomously do the ranking.

2.3 Assumptions, Tackle points, Approach

What are properties that we hope to be able to exploit?

- No-goods from collective choices can result in constraints that eliminate irrelevant parts from the search space. Propagation can help to identify viable candidates more systematically.
- We think that users can eliminate “obviously infeasible” alternatives from A . There is no point in voting over them. Examples include tours that visit some cities more than once, timetable that require a person to split in half etc. That might alleviate some of the specification efforts.
- Similarly, we assume that preferences are defined over (in principle) A , including all infeasible solutions. It does not suffice to only target V since users do not know V a priori.
- Our sets of alternatives A are typically structured. For instance, (A, d) might be a metric space with $d : A \times A \rightarrow \mathbb{R}$ denoting the distance (or $-d$ the similarity) of two options a_1, a_2 . This could be the Hamming-distance or Manhattan-distance of two solutions. Assuming some hidden scoring function s_i , we hope that there is some “fuzzy continuity”:

$$d(a_1) - d(a_2) < \varepsilon \Rightarrow s_i(a_1) \approx s_i(a_2)$$

If we have seen sufficiently many good alternatives (or similarly, bad) our hope is to propagate similarity and traverse the search space.

- Over the course of its runtime, our algorithm partitions U into KA and KU . But there are even more sets, such as KAW and KAN , the known alternatives that are winners and the known alternatives that are non-winners. Hypothetically, if I find one element of KAW , I can stop. We have

$$KAW \uplus KAN \subseteq W \uplus N = V \subseteq KA \uplus U$$

If we could directly shift feasible alternatives from U to KAN (e.g., by propagation, inference, etc.) we could save work/voting. Propagating constraints resulting from KAN could help reduce U . For instance, a greedy heuristic might constrain a potential Copeland winner to achieve more than $x\%$ wins over KA in order to be considered. This could be propagated as a sum that affects U .

- We hope that finding a winner on KA is likely a winner over V .

2.4 Open questions/issues

- If $KA \cup U$ is much larger than V , can we make reliable statements about the winner over V ? Or when can I stop exploring U ?
- Is there a good strategy for picking KA incrementally, e.g., greedy or sim. annealing? Can we design it as a statistical sample of the universe A such that the Copeland or Borda scores are distributed similarly? That would help us devise a stopping criterion (it is *very* unlikely that a much better Copeland winner is lurking outside there).
- Counting argument: Can I chose preferences such that a very small area of the search space changes the behavior of the voting rule entirely? Is it really likely that a (now) very good candidate becomes bad with a larger KA' , resp.?
- Is there some equivalent to tractable classes for preferences/voting rules? Some subclass of problems that make it easier or even tractable? For instance, separability in terms of combinatorial preferences. Or even reduction to variables of interest; compare to tree-width since few variables really interact with respect to satisfaction degrees.
- Does it make sense to have our preferences be adaptive to KA ? For instance, if I know that there are solutions with me having a free Monday – but I accepted a worse solution before that – can I revise my ballot? Risky that agents submit very narrow approval sets. But on the other hand I gain trust that the system offers better solutions when I keep disapproving.
- Should we conduct a study that explores our preference landscapes with respect to continuity?
- Is there a connection between PageRank / Stationary Markov Chains and weighted tournament solutions?
- Sometimes, a matching problem can be asymmetric: Our companies stay the same but our students only have one shot. Unfairness can propagate over years.
- How can we deal with symmetries in our allocations and CP solvers? That could require us to form classes of equivalent elements, allocate the classes (perhaps including cardinality). This could lead to 2-step procedures where we first calculate “optimal” allocations and then resolve this “fairly” by means of lotteries (in case of scarce resources). Such symmetries could be conditional or dynamic, e.g., if two agents only differ in their preference over opinions that are not even feasible.

2.5 Vision

- Our main goal is “Collective Decision Making in Constraint Optimization Problems”: This encompasses majority based democratic rules as well as “social” (equity/fairness) concepts.
- Since we have constraints, we can try to formulate certain minimal requirements to agents that we have to consider. This is important for situations where we can objectively measure fairness/imbalance (e.g., access to schools in 5km).
- We think that there are several learning situations. For instance, we might start with a somewhat coarse understanding of our agents’ preferences and learn about them over the course of optimization. Similarly,
- The availability of MiniZinc voting procedures might lead to increased use of collective optimization in several situations (if fully automatic, e.g.,).
- We hope to find “good” solutions that lead to consensus. Is there a way we can quantify our uncertainty about our users’ preferences? Can evidence about votes update our beliefs about these preferences?
- Example: For our metro line design example, we might have a customer profile that tells us which connections are likely to be relevant (and how our population is comprised of students, employees, elderly, etc). Is that profile consistent with the voting behavior of our population? Can we feed back evidence into that model?

3 Problem Instances

3.1 Single Bus Tour Design

We have a group of n agents that want to decide which sites to visit and when. We have time windows (opening/closing hours), distance travel times, and, of course, personal preferences about

- Which sites do we want to see?
- How much time do we accept to travel?

For that problem, we assume the capacity of the bus to be sufficient.

3.2 Multi Bus Tour Design

We can even decide how to plan more bus trips (with capacities)

3.3 Transport Network Design

In transport network design, we plan how buses/trains should be connected as to maximize satisfaction.