

Multi-agent Team Formation for Design Problems

Leandro Soriano Marcolino¹, Haifeng Xu¹, David Gerber³, Boian Kolev², Samori Price², Evangelos Pantazis³, and Milind Tambe¹

¹ Computer Science Department, University of Southern California, CA, USA

² Computer Science Department, California State University, Dominguez Hills, USA

³ School of Architecture, University of Southern California, CA, USA

Department of Civil Engineering, University of Southern California, CA, USA

{sorianom, haifengx, dgerber}@usc.edu,

{bkolev1, sprice25}@toromail.csudh.edu, {epantazi, tambe}@usc.edu

Abstract. Design imposes a novel social choice problem: using a team of voting agents, maximize the number of optimal solutions; allowing a user to then take an aesthetical choice. In an open system of design agents, team formation is fundamental. We present the first model of agent teams for design. For maximum applicability, we envision agents that are queried for a single opinion, and multiple solutions are obtained by multiple iterations. We show that diverse teams composed of agents with different preferences maximize the number of optimal solutions, while uniform teams composed of multiple copies of the best agent are in general suboptimal. Our experiments study the model in bounded time; and we also study a real system, where agents vote to design buildings.

Keywords: Collaboration · Distributed AI · Team Formation · Design and Computation · Design Optimization · Design Automation

1 Introduction

Teams of voting agents are a powerful tool for finding the optimal solution in many applications [16, 17, 19]. Voting is a popular approach since it is easily parallelizable, it allows the re-use of existing agents, and there are theoretical guarantees for finding one optimal choice [2]. For design problems, however, finding one optimal solution is not enough. For example, it could be mathematically optimal under measurable metrics but lack aesthetic qualities or social acceptance by the target public. Besides, the solution could have a poor performance in some key objective of a multi-objective optimization problem. Essentially, designers need to explore a large set of optimal alternatives, to pick one solution not only according to her aesthetic taste (and/or the one of the target public), but also according to preferences that may be unknown or not formalized [25, 7].

Hence, we actually need systems that find as many optimal solutions as possible, allowing a human to explore such optimal alternatives to make a choice. Even if a user does not want to consider too many solutions, they can be filtered and clustered [5], and be presented in manageable ways [21], allowing her to easily make an informed choice. Therefore, a system of voting agents that produces a unique optimal solution is insufficient, and we propose the novel social choice problem of maximizing the number

of optimal alternatives found by a voting system. As ranked voting may suffer from noisy rankings when using existing agents [11], we study multiple plurality voting iterations, allowing great applicability and re-use of existing agents.

Traditionally, social choice studies the optimality of voting rules, assuming a certain noise models for the agents, and rankings composed of a linear order over alternatives [2, 1]. Hence, there is a single optimal choice, and a system is successful if it can return that optimal choice with high probability. More recently, several works have been considering cases where there is a partial order over alternatives [26, 20], or where the agents output pairwise comparisons instead of rankings [4]. However, these works still focus on finding an optimal alternative, or a fixed-sized set of optimal alternatives (where the size is known beforehand). Therefore, they still provide no help in finding the maximum set of optimal solutions. Moreover, they assume agents that are able to output comparisons among all actions with fairly good precision, and the use of multiple voting iterations has never been studied. When considering agents with different preferences, the field is focused on verifying if voting rules satisfy a set of axioms that are considered to be important to achieve fairness [18]. Meanwhile, the computational design literature has not yet found the potential of teams of voting agents. They study traditional optimization techniques [28], or swarms of agents that interact on the geometric space to emerge aesthetically complex shapes [22, 23].

In this work we bring together the social choice and computational design fields. We present a theoretical study of which kinds of teams are desirable for design problems, and how their size may effect optimality. In doing so, we show many novel results for the study of multi-agent systems. Instead of studying agents with different preferences in order to verify fairness axioms, we show here that agents with different preferences are actually fundamental when voting to find a “truth” (i.e., optimal decisions). On the other hand, agents with the same set of preferences significantly harm the performance, and in general the number of optimal solutions *decreases* as the size of the team grows. Such results were never seen before in the social choice literature. Our theoretical development draws a novel connection between social choice and *number theory*, instead of the traditional connections with *bayesian probability theory*. This novel connection allows us to show that the optimal diverse team size is constant with high probability, and a prime number of optimal actions may impose problems. We also show that we can maximize the number of optimal solutions with agents with different preferences as the team size grows, as long as the team size grows *carefully*. Moreover, we simulate design agents in synthetic experiments to further study our model, confirming the predictions of our theory and providing realistic insights into what happens when systems run with bounded computational time. Finally, we present experiments in a highly relevant domain: architectural design, where we show teams of real design agents that vote to choose the best qualifying and energy-efficient design solutions for buildings.

2 Related Work

Voting systems have been extensively used for many different applications [16, 17, 19]. Mostly because, under some assumptions, they provide optimality guarantees for finding an optimal choice [2]. For many real problems, however, finding one optimal

solution is not enough, and we actually want to find the maximum number of optimal solutions. However, most of the social choice literature is about finding a correct ranking in domains where there is a linear order over the alternatives, and hence a unique optimal decision [2, 1]. Recent works, however, are considering more complex domains. Xia and Conitzer (2011) [26] study the problem of finding k optimal solutions, where k is known beforehand, by aggregating rankings from each agent. However, not only do they need strong assumptions about the quality of the rankings of such agents, but they also show that calculating the *maximum likelihood estimation* (MLE) from the rankings is an NP-hard problem. Procaccia *et al.* (2012) [20] study a similar perspective, where the objective is to find the top k options given rankings from each agent, where, again, k is known in advance. However, in their case, they assume there still exists one unique truly optimal choice, hidden among these top k alternatives. Elkind and Shah (2014) [4], motivated by the crowdsourcing domain, study the case where instead of rankings, the voters output pairwise comparisons among all actions, which may not follow transitivity. However, their final objective is still to pick a single winner, not to maximize the set of optimal solutions found by a voting system. Finally, outputting a full comparison among all actions can be a burden for an agent [12]. Jiang *et al.* (2014) [11] show that actual agents can have very noisy rankings, and therefore do not follow the assumptions of previous works in social choice. Hence, as any agent is able to output at least one action (i.e., a single vote), we study here systems where agents vote across multiple iterations.

Concerning distributed optimization, our work is related to the study of distributed genetic algorithms [13]. Our experimental section relates to the “island model”, where populations evolve concurrently. Normally, however, the populations interact by transferring offsprings, not by voting, and a theoretical study of voting teams which must maximize the number of optimal solutions was never performed. Additionally, the use of multiple classifiers has been a very successful technique in machine learning, in the study of ensemble systems [19]. None of these works, however, explore the potential of multiple agents in maximizing the set of optimal solutions for design problems.

In computational design, automated methods that can provide a high number of optimal alternatives are highly desirable, as it is hard for the human designers to manually find optimal solutions, and they need a large solution pool in order to pick one that fits their aesthetic/subjective evaluation and/or to make a complex trade-off among different objectives that cannot be formalized into a single function [6, 24]. One common method for generating alternatives is to use genetic algorithms [27]. Other optimization methods have also been explored [28]. Another line of work in computational design uses a swarm of agents that move and interact in the geometric space, while depositing material, and hence emerging complex geometrical shapes [22, 23]. Although such works are able to design and create intricate geometries, they are not yet using these agent teams to optimize the designs, let alone finding a maximum number of optimal solutions.

3 Design Domains

We consider in this work domains where the objective is to find the highest number of optimal solutions. We show that design is one of such domains. One of the most common computational design approaches is to use *parametric designs* [8, 5], where

a human designer creates an initial design of a product using *computer-aided design* tools. However, instead of manually deciding all aspects of the product, she leaves *free parameters*, whose values can be modified to change the design of the product. It is up to the designer to decide which parameters are going to be available, their valid types and their valid range.

This approach is used because design is an inherently complex problem. Although a human is able to test and evaluate a few solutions looking for optimality, the number of different possibilities that she can manually create is highly limited, especially under the (common) hard time-constraints. In Figure 1, we show a simple example in the context of architectural design, where the parameters $X1$ and $Y1$ are being used to specify the position of the lower left corner of the building relative to the site boundary.

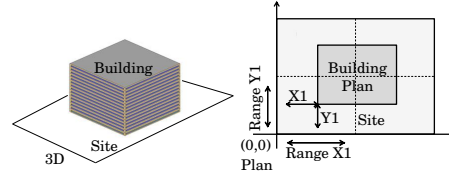


Fig. 1: A parametric design of a building, showing two parameters: $X1$ and $Y1$.

The design of a product normally occurs over multiple phases, where increasing levels of details are decided and optimized. Our work is focused on the initial design phase, when multiple possible design alternatives are analyzed in order to choose one for further study and optimization. This initial design phase is, however, very important to the final performance of a product [21]. For example, in the context of architectural design (as how we explore later in our experiments), it has been acknowledged that it has a high impact on the overall building performance [3]. Design problems are in general multi-objective, since a product normally must be optimized across different objectives. For example, a product should have a low cost, but at the same time high quality. Hence, there are a large number of optimal solutions, all tied in a *Pareto frontier*. For the computational system, these optimal solutions are all equivalent. However, a human may have unknown preferences, may dynamically decide to value some objective over another when handling intricate trade-offs, and/or may choose the option that most pleases her own aesthetic taste or the one of the target public/client.

Note that choosing a design according to aesthetics is an undefined problem, since there are no formal definitions to compare among different options. Hence, the best that a system can do is to provide a human with a large number of optimal solutions (according to other measurable factors), allowing her to freely decide among equally optimal solutions — but most probably with different aesthetic qualities. Therefore, it is natural that in design problems we are going to have many possible solutions, and we want to find as many optimal ones as possible. In fact, the exploration of a large space of possible alternatives is essential in design, as recently shown by other researchers [25, 14, 7]. There are many benefits in discovering a large number of optimal solutions:

Knowledge “Does not Hurt”: We argue that having more optimal solutions to choose from is not worse than having less. Although some works in psychology show that humans may get frustrated in the face of too many options, especially under time pressure [9, 10], we argue that if a designer has enough time or motivation to analyze only x solutions, she can do so with a system that provides more than x optimal solutions

by sampling the exact amount that she desires. However, she will never be able to do so with a system that provides less than x optimal solutions. Note also that the works in psychology [9, 10] were taken in the context of consumers deciding among products to purchase, not in the context of design exploration. As mentioned before, in design the necessity of large exploration spaces is recognized [25, 14, 7]. Moreover, as we discuss in detail later, voting systems could be combined with another system that identify and eliminate solutions that are similar by applying clustering and analysis techniques, and that presents the optimal alternatives to a human in a manageable way [5, 21].

Knowledge Increases Confidence in Optimality: In general design problems, the true Pareto frontier is unknown. Genetic algorithms are widely used in order to *estimate* it. The only knowledge available for the system to evaluate the optimality is in comparison with the other solutions that are also being evaluated during the optimization process [6]. Many apparently “optimal” solutions are actually discovered to be sub-optimal as we find more solutions. Hence, finding a higher number of optimal solutions decreases the risk of a designer picking a wrong choice that was initially outputted as “optimal” by a system (for example, the single agents, as we will show later).

Knowledge Increases Aesthetic Qualities: If a human has a larger set of optimal solutions to choose from, there is a greater likelihood that at least one of these solutions is going to be of high aesthetic quality according to her preferences, or the ones of the target public [7].

Knowledge Increases Diversity of Options: In general, when a system x has more optimal solutions available than a system y , it does not necessarily imply that the solutions in the system x are more similar, while the optimal solutions in y are more different/diverse. In fact, all things equal (i.e., the algorithms are equally able to find unique solutions), the greater the amount of optimal solutions, the higher the likelihood that we have more diverse solutions available. Of course we could have some algorithm x that produces many optimal solutions by creating small variations of one unique solution, but here we do not consider these potentially misleading systems. Again, we assume that such solutions could be identified and filtered by another system [5, 21].

4 Agent Teams for Design Problems

We consider here a team of agents that vote together at each decision point of the design of a product. For the sake of clarity and precision, we present in this section an idealized model. In Section 4.1 we generalize our model to more complex situations, and in Section 5 we generalize further by performing synthetic experiments. Let Φ be a set of agents ϕ , and Ω a set of world states ω . Each ω has an associated set of possible actions A_ω . For example, each world state may represent a parameter of a parametric design problem, and each action may represent a possible value for such parameter. At each world state, each agent ϕ outputs an action a , an optimal action according to the agent’s imperfect evaluation – which may or may not be a truly optimal action. Hence, there is a probability p_j that the agent outputs a certain action a_j . The teams take the action decided by *plurality voting* (i.e, the team takes the decision voted by the largest number of agents – we consider ties are broken uniformly at random).

We assume first that the world states are independent, and by taking an optimal action at all world states we find an optimal solution for the entire problem. That is, we assume first that by taking locally optimal decisions at each design decision point, a globally optimal solution is obtained. We generalize this assumption later, in Proposition 8 (in Section 4.1), where we consider design problems with correlated parameters.

In this paper our objective goes beyond finding one optimal solution, we want to maximize the number of optimal solutions that we can find. For greater applicability, we consider here agents that output a single action. Hence, we generate multiple solutions by re-applying the voting procedure across all world states multiple times (which are called *voting iterations* – one *iteration* goes across all world states, forming one solution). Formally, let \mathbf{S} be the set of (unique) optimal solutions that we find by re-applying the voting procedure through z iterations. Our objective is to maximize $|\mathbf{S}|$. We will show that, under some conditions, we can achieve that when $z \rightarrow \infty$.

We consider that at each world state ω there is a subset $\mathbf{Good}_\omega \subset \mathbf{A}_\omega$ of optimal actions in ω . An optimal solution is going to be composed by assigning any $a \in \mathbf{Good}_\omega$ in world state ω – for all world states. Conversely, we consider the complementary subset $\mathbf{Bad}_\omega \subset \mathbf{A}_\omega$, such that $\mathbf{Good}_\omega \cup \mathbf{Bad}_\omega = \mathbf{A}_\omega$, $\mathbf{Good}_\omega \cap \mathbf{Bad}_\omega = \emptyset$. We drop the subscripts ω when it is clear that we are referring to a certain world state.

One fundamental problem is selecting which agents should form a team. By the classical voting theories, one would expect the best teams to be uniform teams composed of multiple copies of the best agent [2, 15]. Here we show, however, that for design problems uniform teams need very strong assumptions to be optimal, and in most cases they actually converge to always outputting a single solution – an undesirable outcome. However, diverse teams are optimal as long as the team size grows *carefully*, as we explain later in Theorem 1.

We call a team *optimal* when: (i) $|\mathbf{S}| \rightarrow \prod_\omega |\mathbf{Good}_\omega|$ as $z \rightarrow \infty$, and (ii) all optimal solutions are chosen by the team with the same probability $1/\prod_\omega |\mathbf{Good}_\omega|$. Otherwise, even though the team still produces all optimal solutions, it would tend to repeat already generated solutions whose probability is higher. Since in practice there are time bounds, such condition is fundamental to have as many optimal solutions as possible in limited time. Also note that condition (ii) subsumes condition (i), but we keep both for clarity.

We first consider agents whose pdfs are independent and identically distributed. Let $p_j^{\mathbf{Good}}$ be the probability of voting for $a_j \in \mathbf{Good}$, and $p_k^{\mathbf{Bad}}$ be the probability of voting for $a_k \in \mathbf{Bad}$. Let $n := |\Phi|$ be the size of the team, and N_l be the number of agents that vote for a_l in a certain voting iteration. If $\forall a_j \in \mathbf{Good}, a_k \in \mathbf{Bad}, p_j^{\mathbf{Good}} > p_k^{\mathbf{Bad}}$, the team is going to find one optimal solution with probability 1 as $n \rightarrow \infty$:

Observation 1 *The probability of a team outputting one optimal solution goes to 1 as $n \rightarrow \infty$, if $p_j^{\mathbf{Good}} > p_k^{\mathbf{Bad}}, \forall a_j \in \mathbf{Good}, a_k \in \mathbf{Bad}$.*

Note that as the agents are independent and identically distributed, we can model the process of pooling the opinions of n agents as a multinomial distribution with n trials (and the probability of any class k of the multinomial corresponds to the probability p_k of voting for an action a_k). Hence, for each action a_l , the expected number of votes is given by $E[N_l] = n \times p_l$. Therefore, by the *law of large numbers*, if $p_j^{\mathbf{Good}} > p_k^{\mathbf{Bad}} \forall a_j \in \mathbf{Good}, a_k \in \mathbf{Bad}$, we have that $N_j > N_k$. Hence, the team will pick an action $a_j \in \mathbf{Good}$, in all world states, if n is large enough (i.e., $n \rightarrow \infty$).

However, with a team made of copies of the same agent, the system is likely to lose the ability to generate new solutions as n increases. If, for each ω , we have an action a_m^ω such that $p_m^{Good} > p_j^{Good} \forall a_m^\omega \neq a_j^\omega$, the team converges to picking only action a_m^ω . Hence, $|S| = 1$, which is a very negative result for design problems. Therefore, contrary to traditional social choice, here it is not the case that increasing the team size always improves performance. We formalize this notion in Proposition 1 below, where we also show the conditions for a uniform team to be optimal. Let $p_{Good} := \sum_j p_j^{Good}$ be the probability of picking any action in **Good**. We re-write the probability of an action a_j^{Good} as: $p_j^{Good} := \frac{p_{Good}}{|Good|} + \lambda_j$, where $\sum_j \lambda_j := 0$. Hence, some λ_j are positive, and some are negative (unless they are all equal to 0). Let λ^+ be the set of $\lambda_j > 0$. Let λ^{High} be the maximum possible value for $\lambda_j \in \lambda^+$, such that the relation $p_j^{Good} > p_k^{Bad}$, $\forall a_j \in \mathbf{Good}, a_k \in \mathbf{Bad}$ is preserved. We show that when $z \rightarrow \infty$, $|S|$ is the highest as $\max \lambda^+ \rightarrow 0$, and the lowest (i.e., one) as $\min \lambda^+ \rightarrow \lambda^{High}$. Note that $\max \lambda^+ \rightarrow 0$ represents the situation where the probability is equally divided among all optimal actions, and $\min \lambda^+ \rightarrow \lambda^{High}$ represents the case where one optimal action receives a high probability in comparison with the other optimal actions.

Proposition 1. *The maximum value for $|S|$ is $\prod_\omega |Good_\omega|$. When $z, n \rightarrow \infty$, as $\max \lambda^+ \rightarrow 0$, $|S| \rightarrow \prod_\omega |Good_\omega|$. Conversely, as $\min \lambda^+ \rightarrow \lambda^{High}$, $|S| \rightarrow 1$.*

Proof. As $\max \lambda^+ \rightarrow 0$, $\lambda_j \rightarrow 0, \forall a_j$. Hence, $E[N_j] \rightarrow n \times \frac{p_{Good}}{|Good|}, \forall a_j \in \mathbf{Good}$. Because ties are broken randomly, at each world state ω , each $a_j \in \mathbf{Good}_\omega$ is selected by the team with equal probability $\frac{1}{|Good_\omega|}$. As $E[N_j] = E[N_k] \forall a_j, a_k \in \mathbf{Good}$, we have that at each ω it is possible to choose $|Good_\omega|$ different actions. Hence, there are $\prod_\omega |Good_\omega|$ possible combinations of solutions. At each voting iteration, ties are broken at each ω randomly, and one possible combination is generated. As $z \rightarrow \infty$, eventually we cover all possible combinations, and $|S| \rightarrow \prod_\omega |Good_\omega|$.

Conversely, as $\min \lambda^+ \rightarrow \lambda^{High}$, $E[N_j] \rightarrow n \times p_j^{Good}$ for one fixed a_j such that $p_j^{Good} > p_k^{Good}, \forall a_j \neq a_k \in \mathbf{Good}$. Consequently, $E[N_j] > E[N_k]$, at each ω . Hence, there is no tie in any world state, and the team picks a fixed a_j^ω at each world state. Therefore, even if $z \rightarrow \infty$, $|S| \rightarrow 1$. Note that we do not say here that the same action is picked across world states (as a_j^ω may differ for each ω), but that the same optimal solution is picked for all voting iterations. ■

Therefore, uniform teams need a very strong assumption to satisfy condition (i): the probability of voting for optimal actions must be uniformly distributed over all optimal actions (i.e., $\max \lambda^+ \rightarrow 0$). If $\max \lambda^+ \rightarrow 0$, condition (ii) is also satisfied as n grows, because of Observation 1 (i.e., the probability of outputting a suboptimal solution goes to 0) and because of the fact that all actions are equally likely to be chosen; hence each solution is chosen with equal probability $1/\prod_\omega |Good_\omega|$.

We show that, alternatively, we can use agents with different “preferences” (i.e., “diverse” agents), to maximize $|S|$. We consider here agents that have about the same ability in problem-solving, but they prefer different optimal actions. As the agents have similar ability, we consider here the probabilities to be the same across agents, except for the actions in **Good**, as each agent ϕ_i has a subset $Good^i \subset \mathbf{Good}$ consisting of its preferred actions (which are more likely to be chosen than other actions). We denote by

p_{ij} the probability of agent ϕ_i voting for action a_j . Hence, we define the pdf of the diverse agents as: $\forall a_j \in \mathbf{Good}^i$, let $p_{\mathbf{Good}^i} := \sum_j p_{ij}$, $p_{ij} := \frac{p_{\mathbf{Good}^i}}{|\mathbf{Good}^i|}$; $\forall a_j \in \mathbf{Good} \setminus \mathbf{Good}^i$, $p_{ij} := \frac{p_{\mathbf{Good}} - p_{\mathbf{Good}^i}}{|\mathbf{Good} \setminus \mathbf{Good}^i|}$; and $\forall a_k \notin \mathbf{Good}^i$, $a_j \in \mathbf{Good}^i$, $p_{ij} > p_{ik}$. $\mathbf{Good}^i \cap \mathbf{Good}^l$ (of agents ϕ_i and ϕ_l) is not necessarily \emptyset . The pdfs are strictly defined in this section for the sake of clarity and precision, but in the next section and in our synthetic experiments we generalize further. In Figure 2 we show an illustrative example.

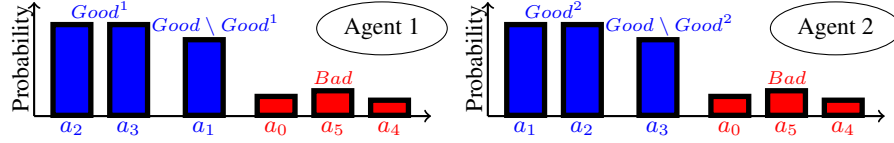


Fig. 2: Illustrative example of the probability distribution functions of two diverse agents.

Let's consider we can draw diverse agents from a distribution \mathcal{F} . Each agent ϕ_i has $r < |\mathbf{Good}|$ actions in its \mathbf{Good}^i , and we assume that all actions in \mathbf{Good} are equally likely to be selected to form \mathbf{Good}^i (since they are all equally optimal). Note that r is the same for all agents (as, again, we assume they have the same pdfs, but different preferences), and that we also cover the case where each agent prefers a single action (which would be $r := 1$). We show that by drawing n agents from \mathcal{F} , the team is optimal for large n with probability 1, as long as n is a multiple of a divisor (> 1) of each $|\mathbf{Good}_w|$. We also show that the minimum necessary optimal team size is constant with high probability as the number of world states grow. We start with the following:

Proposition 2. *If a team of size n is optimal at a world state, then $\gcd(n, |\mathbf{Good}|) > 1$. That is, n and $|\mathbf{Good}|$ are not co-prime.*

Proof (by contradiction). By the optimality requirement (ii), each action must be in the \mathbf{Good}^i set of the same number of agents. Otherwise, if an action a_i is preferred by a larger number of agents than another action a_j , the team would pick a_i with a larger probability than a_j . Hence, we must have that:

$$n \times r = k \times |\mathbf{Good}|, \quad (1)$$

where k is a constant $\in \mathbb{N}_{>0}$. k represents the number of agents that have a given action a_j in its \mathbf{Good}^i . Note that it must be the same for all optimal actions, and therefore we have a single constant. If n and $|\mathbf{Good}|$ are co-prime, then it must be the case that r is divisible by $|\mathbf{Good}|$. However, this yields $r \geq |\mathbf{Good}|$, which contradicts our assumption. Therefore, n and $|\mathbf{Good}|$ are not co-prime. ■

We illustrate Equation 1 with an example in Figure 3. In the figure we show 6 agents ($n := 6$), with 2 preferred actions each ($r := 2$). Note that each action is preferred by 4 agents, and hence we show a case where $k := 4$. As an example, we mark with a dashed circle one of the actions, a_2 . In such case, the team will have an equal probability

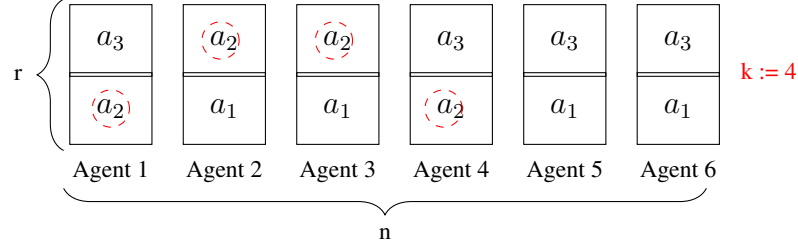


Fig. 3: Illustrative example of Equation 1. Each action is in the list of preferences of 4 agents ($k := 4$). As an example, we mark with a dashed circle one of the actions, a_2 .

of picking all optimal actions, and optimality condition (ii) would be satisfied if the probability of picking suboptimal actions is 0. If, for example, we now change agent 5 to prefer actions a_2 and a_3 (replacing action a_1 by a_2), then the team would be more likely to pick action a_2 by plurality voting than any other action, and it would be less likely to pick action a_1 than any other action. As the number of voting iterations is limited in actual applications, this situation is not desirable.

Note that we could also have a case where one agent prefers a larger number of actions than others. For example, we could change agent 5 to prefer actions a_1 , a_2 and a_3 . However, as the agent has a limited amount of probability distributed over the actions in the \mathbf{Good}^i set (i.e., $p_{\mathbf{Good}^i}$), we have that necessarily the probability of the agent voting for a_1 and a_3 would drop; hence, the team would pick a_2 more often than the other actions, and a_1 and a_3 less often than the other actions. This does not mean, however, that there is a single optimal configuration for each number of optimal actions $|\mathbf{Good}|$. There are multiple possible solutions for Equation 1, but in any possible solution we will find that the size of the team n and $|\mathbf{Good}|$ are not co-prime.

Proposition 2 is a necessary but not sufficient condition for optimality. If Equation 1 is satisfied, all optimal actions will be selected with the same probability, but it is still necessary for the probability of picking suboptimal actions to go to 0 to fully satisfy condition (ii). That will be the case if $p_{\mathbf{Good}^i} = 1$, or if $n \rightarrow \infty$, since $p_j^{\mathbf{Good}} > p_k^{\mathbf{Bad}}$, $\forall a_j \in \mathbf{Good}, a_k \in \mathbf{Bad}$. Note that Proposition 2 implies hard restrictions for world states where $|\mathbf{Good}|$ is prime, or for teams with prime size n : if n is prime, $|\mathbf{Good}|$ must be a multiple of n ; and if $|\mathbf{Good}|$ is prime, n must be a multiple of $|\mathbf{Good}|$.

Let's analyze across a set of world states Ω . For a team of fixed size n , Proposition 2 applies across all world states. Hence, the team size must be a multiple of a divisor (> 1) of each $|\mathbf{Good}_\omega|$. Note that the pdfs of the agents (and also r) may change according to ω . Let \mathbf{D} be a set containing one divisor of each world state (if two or more world states have a common divisor x , it will be representable by only one $x \in \mathbf{D}$). Hence, $\forall \omega$, $\exists d \in \mathbf{D}$, such that $d \mid |\mathbf{Good}_\omega|$; and $\forall d \in \mathbf{D}$, $\exists \mathbf{Good}_\omega$, such that $d \mid |\mathbf{Good}_\omega|$. There are multiple possible \mathbf{D} sets, from the superset of all possibilities \mathcal{D} . Therefore, we can now study the minimum size necessary for an optimal team. Applying Proposition 2 at each world state ω , we have that the minimum size necessary for an optimal team is $n = \min_{\mathbf{D} \in \mathcal{D}} \prod_{d \in \mathbf{D}} d$. Hence, our worst case is when each $|\mathbf{Good}_\omega|$ is a unique prime, as the team will have to be a product of all (unique) optimal action space sizes:

Proposition 3. *In the worst case, the minimum team size is exponential in the size of the world states $|\Omega|$. In the best case, the minimum necessary team size is a constant with $|\Omega|$.*

Proof. In the worst case, each added world state ω has a unique prime optimal action space size. Hence, the minimum team size is at least the product of the first $|\Omega|$ primes, which, by the *prime number theorem*, has growth rate $\exp((1 + o(1))|\Omega| \log |\Omega|)$. In the best case, each added \mathbf{Good}_ω has a common divisor with previous ones, and the minimum necessary team size does not change. ■

However, we show that the worst case happens with low probability, and the best case with high probability. Let G be the maximum possible $|\mathbf{Good}|$, and $M := |\Omega|$. Assume that each world state ω_j will have a uniformly randomly drawn number of optimal actions, denoted as m_j , for all $j = 1, \dots, M$ (i.e., $\forall \omega \in \Omega$). We assume that G is large enough, so that the probability that a given m_j has factor p is $1/p$.

Proposition 4. *The probability that the minimum necessary team size grows exponentially tends to 0, and the probability that it is constant tends to 1, as $M \rightarrow \infty$.*

Proof. It is sufficient to show that the probability that m_1, \dots, m_{M-1} are all co-prime with m_M tends to 0 as $M \rightarrow \infty$. That is, we show that when adding a new world state ω_M , its $|\mathbf{Good}_\omega|$ will have a common factor with the size of the \mathbf{Good} set of some of the other world states with high probability. Given any prime p , the probability that at least one of any independently randomly generated $M - 1$ numbers m_1, \dots, m_{M-1} has factor p is $1 - (1 - \frac{1}{p})^{M-1}$, while the probability that one independently randomly generated number m_M has factor p is $\frac{1}{p}$ (for large enough G). Therefore, the probability m_M shares common factor p with at least one of m_1, \dots, m_{M-1} is $\frac{1 - (1 - \frac{1}{p})^{M-1}}{p}$. The probability that m_M is co-prime with all m_1, \dots, m_{M-1} is:

$$\prod_{\text{all primes } p} \left[1 - \frac{1 - (1 - \frac{1}{p})^{M-1}}{p} \right],$$

which, as $M \rightarrow \infty$, tends to:

$$\prod_{\text{all primes } p} \left(1 - \frac{1}{p} \right) = \frac{1}{\zeta(1)} = 0,$$

where $\zeta(s)$ is the Riemann zeta function. The last equality holds true since:

$$\zeta(1) = \prod_{\text{all primes } p} \frac{1}{1 - p^{-1}} = \sum_{i=1}^{\infty} \frac{1}{i} \rightarrow \infty$$

(as shown by Euler). Hence, with high probability, when adding a new world state ω , $|\mathbf{Good}_\omega|$ will share a common factor with a world state already in Ω . ■

Finally, in the next theorem we show that a diverse team of agents is always optimal as the team grows, as long as it grows *carefully*. That is, we show that for large diverse

teams we will be able to satisfy the optimality conditions (i) and (ii), as long as the team size is a multiple of a divisor of $|\mathbf{Good}_\omega|$, $\forall \omega \in \Omega$. Again, we assume that G is large enough, so that the probability that a given m_j has factor p is $1/p$.

Theorem 1. *Let $D \in \mathcal{D}$ be a set containing one factor from each \mathbf{Good}_ω . For arbitrary n , the probability that we generate (by drawing from a distribution \mathcal{F}) an optimal team of size n converges to 0 as $|\Omega| \rightarrow \infty$. However, if $n = c \prod_{d \in D} d$, then the probability that the team is optimal tends to 1 as $c \rightarrow \infty$.*

Proof. For an arbitrary team size n , let P be the set of its prime factors. Given one $p \in P$, the probability that p is not a factor of $|\mathbf{Good}_\omega|$ is $1 - 1/p$. The probability that all $p \in P$ are not factors is: $\prod_p (1 - 1/p)$. As $0 < \prod_p (1 - 1/p) < 1$, the probability that at least one $p \in P$ is a factor of $|\mathbf{Good}_\omega|$ is $1 - \prod_p (1 - 1/p) < 1$. For $|\Omega|$ tests, the probability that at least one p is a factor in all of them is:

$$\left(1 - \prod_p (1 - 1/p)\right)^{|\Omega|},$$

which tends to 0, as $|\Omega| \rightarrow \infty$. Hence, the probability that $\gcd(n, |\mathbf{Good}_\omega|) = 1$ for at least one ω tends to 1, and the probability that the team can be optimal tends to 0.

However, if:

$$n = c \prod_{d \in D} d,$$

then $\gcd(n, |\mathbf{Good}_\omega|) \neq 1 \forall \omega \in \Omega$, satisfying the necessary condition in Proposition 2 at all world states. Let n_j be the number of agents ϕ_i that have a_j in its \mathbf{Good}^i , and $P(n_j = n_k)$ be the probability that $n_j = n_k$ (that is, the probability that the same number of agents have a_j and a_k in their \mathbf{Good}^i). As each a_j has equal probability of being in a \mathbf{Good}^i , for a large number of drawings from \mathcal{F} (i.e., $c \rightarrow \infty$), we have that $P(n_j = n_k) \rightarrow 1, \forall a_j, a_k \in \mathbf{Good}_\omega, \forall \omega$, by the *law of large numbers*. Hence, each optimal solution will be selected with the same probability. Moreover, as $p_j^{\mathbf{Good}} > p_k^{\mathbf{Bad}}$, $\forall a_j \in \mathbf{Good}, a_k \in \mathbf{Bad}$, the probability of picking a suboptimal solution converges to 0 (as $n \rightarrow \infty$ with $c \rightarrow \infty$), and hence the probability of picking each of the optimal solutions converges to $1 / \prod_\omega |\mathbf{Good}_\omega|$ (satisfying optimality condition (ii)). ■

If it is expensive to test values for n such that Theorem 1 is satisfied, we can choose $n = c \prod_\omega |\mathbf{Good}_\omega|$, as it immediately implies the conditions of the theorem. Moreover, if we know the size of all $|\mathbf{Good}_\omega|$, we can check if n and $|\mathbf{Good}_\omega|$ are co-prime in $O(h)$ time (where h is the number of digits in the smaller number), using the *Euclidean algorithm*. Hence, we can test all world states in $O(|\Omega|h)$ time.

4.1 Generalizations

In this section we present several generalizations from our initial idealized model, in order to cover more realistic situations. We start by generalizing our theory to cases where the agents do not have only a probability of $p_{ij} := \frac{p_{\mathbf{Good}^i}}{|\mathbf{Good}^i|}$ or $p_{ij} := \frac{p_{\mathbf{Good}} - p_{\mathbf{Good}^i}}{|\mathbf{Good} \setminus \mathbf{Good}^i|}$ to

vote for actions in *Good* (depending if the action is in Good^i or not), but now can have different probabilities distributed over the actions in *Good*. Hence, we now model each agent as having a set of Good^i sets, each with its own probability distributed over the actions in the set. For this generalization, we still consider that the agents have the same pdf, but different preferences. That is, the agents may have different actions at each Good^i set, but their size and the number of sets is the same across agents.

Hence, we denote each Good^i set j as ${}_j\text{Good}^i$. Each also has its own $p_{{}_j\text{Good}^i}$ total probability, that will be equally distributed among all actions in ${}_j\text{Good}^i$, in a similar fashion as before. As mentioned, the content of each ${}_j\text{Good}^i$ set may differ across agents, but we consider the $p_{{}_j\text{Good}^i}$ to be the same across agents. Note that the case where each action has a different probability is defined as the situation where each $|{}_j\text{Good}^i| := 1$. Similarly as before, we consider that each agent ϕ_i has ${}_j r < |\text{Good}|$ actions at each ${}_j\text{Good}^i$, and all actions in *Good* are equally likely to be selected to form each ${}_j\text{Good}^i$. In Figure 4 we show an illustrative example of the pdf of two agents with multiple ${}_j\text{Good}^i$ sets.

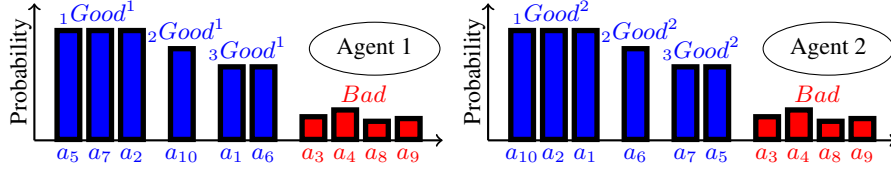


Fig. 4: Illustrative example of the probability distribution functions of two agents with multiple ${}_j\text{Good}^i$ sets.

Proposition 5. *Theorem 1 still applies under the more general model stated above. That is, if $n = c \prod_{d \in D} d$, then the probability that the team is optimal tends to 1 as $c \rightarrow \infty$.*

Proof. Similarly as before, for each ${}_j\text{Good}^i$ we must have that:

$$n \times {}_j r = {}_j k \times |\text{Good}|, \quad (2)$$

so that for each ${}_j\text{Good}^i$ we have that ${}_j k$ agents have a given action a in its ${}_j\text{Good}^i$. As the total probability $p_{{}_j\text{Good}^i}$ of each set is the same across agents, we have that each optimal action will be selected by the team with the same probability when Equation 2 is satisfied for all ${}_j\text{Good}^i$. Hence, across world states, each optimal solution will also have the same probability of being selected. Similarly as in Proposition 2, for Equation 2 to be satisfied, we must have that n and $|\text{Good}|$ are not co-prime, and that will be true when $n = c \prod_{d \in D} d$.

Let ${}_j n_l$ be the number of agents ϕ_i that have a_l in its ${}_j\text{Good}^i$, and $P({}_j n_l = {}_j n_m)$ be the probability that ${}_j n_l = {}_j n_m$. Like before, as each a_l has equal probability of being in a ${}_j\text{Good}^i$, for a large number of drawings from \mathcal{F} (i.e., $c \rightarrow \infty$), we have that $P({}_j n_l = {}_j n_m) \rightarrow 1, \forall a_l, a_m \in \text{Good}_\omega, \forall \omega$, by the law of large numbers.

Notice that this happens for all j \mathbf{Good}^i sets. Hence, all optimal actions will be selected with the same probability by the team. Like before, as $p_l^{\mathbf{Good}} > p_m^{\mathbf{Bad}}, \forall a_l \in \mathbf{Good}, a_m \in \mathbf{Bad}$, the probability of picking a suboptimal action converges to 0 (as $n \rightarrow \infty$ with $c \rightarrow \infty$), and hence the probability of picking each of the optimal solutions converges to $1/\prod_{\omega} |\mathbf{Good}_{\omega}|$ (satisfying optimality condition (ii)). ■

Now we present our second generalization. We show that Theorem 1 still applies for agents ϕ_i with different probabilities over optimal actions $p_{\mathbf{Good}^i}$. We consider here a more general definition of *optimal* team: the difference between the probabilities of picking each optimal solution and $1/\prod_{\omega} |\mathbf{Good}_{\omega}|$ must be as small as possible.

Hence, let p_j^{Φ} be the probability of team Φ picking optimal action a_j , the optimal team is such that $\vartheta := \sum_{a_j} |p_j^{\Phi} - 1/|\mathbf{Good}_{\omega}||, \forall a_j \in \mathbf{Good}_{\omega}$ is minimized ($\forall \omega \in \Omega$). We focus here in a single world state ω , as by minimizing ϑ in each world state we are also making the difference between the probability of picking each optimal solution and $1/\prod_{\omega} |\mathbf{Good}_{\omega}|$ as small as possible. Hence, the original definition in the previous section is the case where $\vartheta := 0$.

Proposition 6. *Theorem 1 still applies when $|p_{\mathbf{Good}^i} - p_{\mathbf{Good}^j}| \leq \epsilon, \forall \phi_i, \phi_j$, for small enough $\epsilon > 0$.*

Proof. Let Φ be an optimal team, where $p_{\mathbf{Good}^i}$ is the same for all agents ϕ_i . Hence, the probability of all actions in \mathbf{Good} being selected by the team is the same. I.e., $p_k^{\Phi} = p_l^{\Phi}, \forall a_k, a_l \in \mathbf{Good}$, and $\vartheta := 0$. Let $\Delta := \sum_{a_k \in \mathbf{Good}} \sum_{a_l \in \mathbf{Good}} |p_k^{\Phi} - p_l^{\Phi}|$ be the difference between the probabilities of the team taking each optimal action. In the rest of the proof we will disturb the probabilities $p_{\mathbf{Good}^i}$ of sets of agents, which will change Δ . We focus in studying the variation in Δ , as minimizing the variation in Δ also minimizes the variation in ϑ .

We prove by mathematical induction. Assume we change the $p_{\mathbf{Good}^i}$ of x agents ϕ_i , and Δ is as small as possible. Now we will change $x + 1$ agents. Let's pick one agent ϕ_i and increase its $p_{\mathbf{Good}^i}$ by $\delta \leq \epsilon$. It follows that $p_k^{\Phi} > p_l^{\Phi}, \forall a_k \in \mathbf{Good}^i, a_l \notin \mathbf{Good}^i$, and the new $\Delta' := \sum_{a_k \in \mathbf{Good}} \sum_{a_l \in \mathbf{Good}} |p_k^{\Phi} - p_l^{\Phi}| > \Delta$.

If we add one more agent ϕ_j , such that $\mathbf{Good}^j \cap \mathbf{Good}^i = \emptyset$, the probability of voting for actions $a_m \in \mathbf{Good}^j$ increases. For small enough ϵ , $p_{\mathbf{Good}^j}$ will be too large to precisely equalize the probabilities, and it follows that $p_m^{\Phi} > p_k^{\Phi} > p_l^{\Phi}, \forall a_m \in \mathbf{Good}^j, a_k \in \mathbf{Good}^i, a_l \notin \mathbf{Good}^i \cup \mathbf{Good}^j$, and $\Delta'' := \sum_{a_k \in \mathbf{Good}} \sum_{a_l \in \mathbf{Good}} |p_k^{\Phi} - p_l^{\Phi}| > \Delta'$. The same applies for each newly added agent, until we have a new team such that $n = c \prod_{d \in \mathbf{D}} d$ (again, satisfying the conditions of the theorem).

The base case follows trivially. If we did not change the probability of any agent (i.e., $x := 0$), and we now increase $p_{\mathbf{Good}^i}$ of a single agent $\phi_i, p_k^{\Phi} > p_l^{\Phi}, \forall a_k \in \mathbf{Good}^i, a_l \notin \mathbf{Good}^i$, and $\Delta' > \Delta$. By the same argument as before, adding more agents will only increase Δ' , until $n = c \prod_{d \in \mathbf{D}} d$. ■

Thirdly, we also generalize to the case where the number of preferred actions r changes for each agent. We consider that the number of actions in the \mathbf{Good}^i of each agent ϕ_i (r^i) is decided according to a uniform distribution on the interval $[1, r']$.

Proposition 7. *If $n = r' \times c \prod_{d \in \mathbf{D}} d$, the probability that the team is optimal $\rightarrow 1$ as $c \rightarrow \infty$.*

Proof. For large n , the number of agents with $r^i = 1, \dots, r'$ is the same. Therefore, if for each subset $\Phi^i \subset \Phi$, such that $r^\phi = i, \forall \phi \in \Phi^i$, we have that $p_k^{\Phi^i} = p_l^{\Phi^i}, \forall a_k, a_l \in \text{Good}$, we will have that $p_k^{\Phi} = p_l^{\Phi}, \forall a_k, a_l \in \text{Good}$. Given an optimal team of size n , we have r' subsets Φ^i of size n/r' each. It follows by Theorem 1 that $n/r' = c \prod_{d \in D} d$, and:

$$n = r' \times n/r' = r' \times c \prod_{d \in D} d,$$

hence n also follows the necessary conditions in Proposition 2. Similarly as in Theorem 1, as $n \rightarrow \infty$ with $c \rightarrow \infty$, the probability of picking a suboptimal solution converges to 0, and the probability of picking each of the optimal solutions converges to $1 / \prod_{\omega} |\text{Good}_{\omega}|$ (satisfying optimality condition (ii)). ■

Lastly, we discuss the assumption that world states are independent. In design problems they could actually be correlated. We present below a constructive proof showing that we can use our model to study design problems with correlated parameters.

Proposition 8. *The previous results apply for problems with correlated parameters.*

Proof. Let's consider a design problem with a set \mathcal{Y} of parameters v . We can divide \mathcal{Y} in \mathcal{Y}_k sets, where all $v \in \mathcal{Y}_k$ are correlated, but v_i and v_j are independent, $\forall v_i \in \mathcal{Y}_i, v_j \in \mathcal{Y}_j, i \neq j$. That is, all parameters v in a \mathcal{Y}_k set are correlated, but the parameters between two different \mathcal{Y}_k sets are independent. This can always be performed, as in the worst case where all parameters are correlated, we can have a single $\mathcal{Y}_k := \mathcal{Y}$.

Now, instead of modeling each design parameter v as a world state ω (as in our original model), we can model each set \mathcal{Y}_k as a world state ω . Hence, instead of an action a being one value assigned to a parameter v , an action a now represents one full combination of values to each v_k in a set \mathcal{Y}_k . Therefore, instead of voting at each parameter v , each agent ϕ_i now votes for one combination of value assignments (of correlated parameters) at each set \mathcal{Y}_k . As all sets \mathcal{Y}_k are independent, we still have agents voting for independent world states ω and the previous results still apply. In the worst case, where all parameters of the problem are correlated, we would have agents voting for entire solutions, and the model would be considered as having a single world state. ■

5 Synthetic Experiments

We run synthetic experiments, where we simulate design agents and evaluate diverse and uniform teams (henceforth *diverse* and *uniform*). We randomly create pdfs for the agents, and simulate voting iterations across a series of world states. We repeat all our experiments 100 times, and in the graphs we plot the average and the confidence interval of our results (according to a *t-test* with $p := 0.01$). We run 1000 voting iterations (z), and measure how many optimal solutions the team is able to find. We study a scenario where the number of actions ($|\mathcal{A}| := 100$, and the number of optimal actions per world state ($|\text{Good}_{\omega}|$) is, respectively: $\langle 2, 3, 5, 5, 5 \rangle$, in a total of 750 optimal solutions.

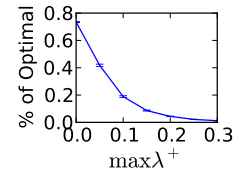


Fig. 5: Percentage as $\max \lambda^+$ grows.

At each repetition of our experiment, we randomly create a pdf for the agents. We start by studying the impact of $\max \lambda^+$ in *uniform*. When creating the *uniform* team, the total probability of playing any of the optimal actions (i.e., p_{Good}) is randomly assigned (uniform distribution) between 0.6 and 0.8. We fix the size of the team (25) and evaluate different $\max \lambda^+$ in Figure 5. As expected from Proposition 1, for $\max \lambda^+ := 0$ the system finds the highest number of optimal solutions; and as $\max \lambda^+$ increases, it quickly drops.

We then study the impact of increasing the number of agents, for *uniform* and *diverse*. To generate a *diverse* team, we draw randomly a r_ω in an interval U for each world state, that will be the size of $|Good^i|$. We study three variants: *diverse**, where $U := (0, |Good_\omega|]$; *diverse*, where $U := (0, |Good_\omega|)$, and *diverse Δ* , where we allow agents to have different r_ω^i , also drawn from $(0, |Good_\omega|)$. We independently create pdfs randomly for each agent ϕ_i . For each agent we draw a number between 0.6 and 0.8 to distribute over the set of optimal actions, and randomly decide r_ω actions to compose its $Good^i$ set. We distribute equally 80% of the probability of voting over optimal actions on the actions of that set.

As we can see, in Figure 6, the number of solutions *decreases* for *uniform* as the number of agents grows. Normally, in social choice, we expect the performance to improve as teams get larger, so this is a novel result. It is, however, expected from our Proposition 1. *Diverse*, on the other hand, improves in performance for all 3 versions, as predicted by our theory. However, the system seems to converge for a fixed z , as the performance does not increase much after around 20 agents. Hence, in Figure 7 we study larger *diverse* (continuous line) and *diverse Δ* teams (dashed line), going all the way up to 1800 agents. We also study four different number of voting iterations (z , shown in the figure by different lines): 1000, 2000, 3000, 4000. As we can see, although adding more agents was not really improving the performance in the experimental scenario under study, there is clearly a statistically significant improvement ($p < 0.01$) by increasing the number of voting iterations, with the system improving from finding around 53% of the optimal solutions, all the way up to finding more than 80% of them. However, there is a *diminishing returns* effect, as the impact of adding more iterations decreases as the actual number of iterations grow larger. We also note that *diverse Δ* is better than *diverse*, and the difference increases as z grows.

As we can see, although theoretically possible, it is still a challenge to have a system that can find all the possible optimal solutions. Moreover, it would be expensive to pool the votes of agents through a large number of voting iterations. However, as we show next, we can actually approximate this process in a real system, by pooling only a small number of solutions from each agent, and executing many voting iterations by aggregating different combinations of these solutions.

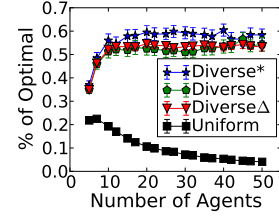


Fig. 6: Percentage of optimal solutions as # agents grows.

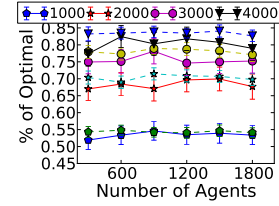


Fig. 7: Percentage for larger teams.

6 Experiments in Architectural Design

6.1 Architectural Design Domain

We study a real system for architectural building design. This is a fundamental domain, since the design of a building impacts its energy usage during its whole life-span [6, 3]. We use Beagle [6], a multi-objective design optimization software that assists users in the early stage design of buildings. First, the designer creates a parametric design, containing (as discussed in Section 3) a set of parameters that can be modified within a specified range, allowing the creation of many variations. The ranges are defined according to the legislation (i.e., setback, maximum height, etc), or the intention of the designer (for example, the general shape of the building). We use designs from Gerber and Lin (2013) [6], shown in Figure 8: *base*, a simple building type with uniform program (i.e., tenant type); *office park*, a multi-tenant grouping of towers; and *contemporary*, a double “twisted” tower that includes multiple occupancy types, relevant to contemporary architectural practices.

Beagle uses a genetic algorithm (GA) to optimize the building design based on three objectives: energy efficiency, financial performance and area requirements. In detail, the objective functions are: $S_{obj} : \max SPCS$; $E_{obj} : \min EUI$; $F_{obj} : \max NPV$. SPCS is the Spatial Programming Compliance Score, EUI is the Energy Use Intensity and NPV is the Net Present Value, defined as follows.

SPCS defines how well a building conforms to the project requirements (by measuring how close the area dedicated to different activities is to a given specification). Let L be a list of activities (in our designs, $L = \langle \text{Office, Hotel, Retail, Parking} \rangle$), $area(l)$ be the total area in a building dedicated to activity l and $requirement(l)$ be the area for activity l given in a project specification. SPCS is defined as: $SPCS := 100 \times \left(1 - \frac{\sum_{l \in L} |area(l) - requirement(l)|}{|L|} \right)$.

EUI regulates the overall energy performance of the building. This is an estimated overall building energy consumption in relation to the overall building floor area. The process to obtain the energy analysis result is automated in Beagle through Autodesk Green Building Studio (GBS) web service.

Finally, **NPV** is a commonly used financial evaluation. It measures the financial performance for the whole building life cycle, given by: $NPV := \left(\sum_{t=1}^T \frac{c_t}{(1+r)^t} \right) - c_0$, where T is the Cash Flow Time Span, r is the Annual Rate of Return, c_0 is the construction cost, and $c_t := \text{Revenue} - \text{Operation Cost}$.

Many options affect the execution of the GA, including: initial population size, size of the population, selection size, crossover ratio, mutation ratio, maximum iteration. Further details about Beagle are at Gerber and Lin (2013) [6]. In the end of the optimization process, the GA outputs a set of solutions. These are considered “optimal”, according to the internal evaluation of the GA, but are not necessarily so. As in our theory, for each parameter the assigned value is going to be one of the optimal ones with a certain



Fig. 8: Parametric designs.

probability. In fact, most of the solutions outputted by the GAs are later identified as sub-optimal and eliminated in comparison with better ones found by the teams.

We model each run of the GA as an agent ϕ . Each parameter of the parametric design is a world state ω , where the agents decide among different actions \mathbf{A} (i.e., possible values for the current parameter). Our model assumes independent multiple voting iterations across all world states. However, in general it could be expensive to pool agents for votes in a large number of iterations. Therefore, in order to test the applicability of the predictions of our model in more realistic scenarios, in our experiments we pool only 3 solutions per agent, but run multiple voting iterations by aggregating over all possible combinations of them. That is, at each combination we pick one solution per agent, and vote across all the parameters, in a total of 81 voting iterations with 4 agents.

6.2 Empirical Results

We run experiments across the different parametric designs shown in Figure 8. These are designs with increasing complexity. More details about the designs and the meanings of each parameter are available in Gerber and Lin (2013) [6]. We create 4 different agents, using different options for the GA, as shown in Table 1 (Initial Population and Maximum Iteration were kept as constants: 10 and 5. PZ = Population Size, SZ = Selection Size, CR = Crossover Ratio, MR = Mutation Ratio). We are dealing here with real (and consequently complex) design problems, where the true set of optimal solutions is unknown. We approach the problem in a comparative fashion: when evaluating different systems, we consider the union of the set of solutions of all of them. That is, let \mathbf{H}_x be the set of solutions of system x ; we consider the set $\mathcal{H} := \bigcup_x \mathbf{H}_x$. We compare all solutions in \mathcal{H} , and consider as optimal the best solutions in \mathcal{H} , forming the set of optimal solutions \mathcal{O} . We use the concept of Pareto dominance: the best solutions in \mathcal{H} are the ones that *dominate* all other solutions (i.e., they are better in all 3 objectives). As we know which system generated each solution $o \in \mathcal{O}$, we estimate the set of optimal solutions \mathbf{S}_x of each system.

Although our theory focuses on *plurality voting*, we also present results using the *mean* and the *median* of the opinions of the agents. That is, given one combination (a set of one solution from each agent), we also generate a new solution by calculating the *mean* or the *median* of the values from each agent across all parameters. Also, when performing the voting aggregation (*vote*), we consider values that are the same up to 3 decimal places as equal. Concerning *uniform*, we evaluate a team composed of copies of the “best” agent. By “best”, we mean the agent that finds the highest number of optimal solutions. According to Proposition 1, such an agent should be the one with the lowest $\max \lambda^+$, and we can predict that voting among copies of that agent generates a large number of optimal solutions. Hence, for each design, we first compare all solutions of all agents, to estimate which one has the largest set of optimal solutions \mathbf{S} . We, then, run that agent multiple times, creating *uniform*. For *diverse*, we consider one copy of each agent in Table 1. We aggregate the solutions of *diverse* and *uniform*. We run 81 aggregation iterations, by selecting 3 solutions from each agent ϕ_i , in its set of solutions

Table 1: GA parameters.

Agent	PZ	SZ	CR	MR
Agent 1	12	10	0.8	0.1
Agent 2	18	8	0.6	0.2
Agent 3	24	16	0.55	0.15
Agent 4	30	20	0.4	0.25

H_i , and aggregating all possible combinations of these solutions. We evaluate together the solutions of all agents and all teams (i.e., we construct \mathcal{H} with the solutions of all systems), in order to estimate the size of S_x of each system.

Since the true optimal solutions set is unknown, we first plot the percentage of unique solutions found by each system in relation to the total number of unique optimal solutions in \mathcal{H} . Hence, in Figure 9 (a), we show the percentage of optimal solutions for all systems, in relation to $|\mathcal{O}|$. For clarity, we represent the result of the individual agents by the one that had the highest percentage. As we can see, in all parametric designs the teams find a significantly larger percentage of optimal solutions than the individual agents. The agents find less than 1% of the solutions, while the teams are in general always close to or above 15%. In total (considering all aggregation methods and all agents), for all three parametric designs the agents find only about 1% of the optimal solutions, while *uniform* finds around 51% and *diverse* 47%. Looking at *vote*, in *base diverse* finds a larger percentage of optimal solutions than *uniform* (around 9.4% for *uniform*, while 11.6% for *diverse*). In *office park* and *contemporary*, however, *uniform* finds more solutions than *diverse*. Based on Proposition 1, we expect that this is caused by the best agent having a lower $\max \lambda^+$ in *office park* and *contemporary* than in *base*.

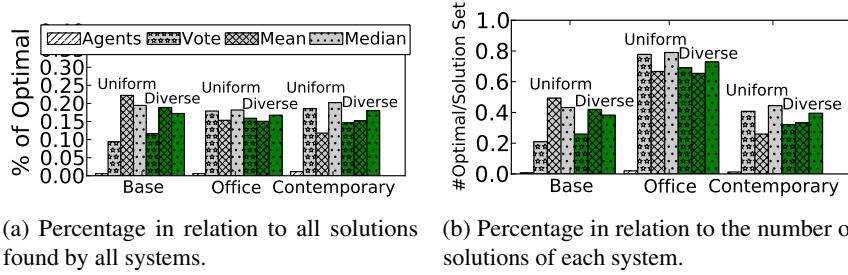


Fig. 9: Percentage of optimal solutions of each system.

Figure 9 (b) shows the percentage of optimal solutions found, in relation to the size of the set of evaluated solutions of each system. That is, let O_x be the set of optimal solutions of system x , in \mathcal{O} . We show $\frac{|O_x|}{|H_x|}$. Concerning *vote*, the teams are able to find a new optimal solution around 20% of the time for *base*, around 73% of the time for *office park* and around 36% of the time for *contemporary*. Meanwhile, for the individual agents it is close to 0%. We can see that teams have great potential for generating new optimal solutions, as expected from our theory. However, as studied in our synthetic experiments, we can expect some *diminishing returns* when increasing the number of voting iterations. It is interesting to note that the performance of the teams is much higher for *office park* than for the other two parametric designs. In *base* and *contemporary*, the building mass is parametrized into a single volume, while in *office park* the building mass has multiple volumes. Hence, a possible explanation is that the division in multiple volumes facilitated the generation of multiple optimal solutions, since these can be combined

in many different ways. We also plot in Figure 10 (a) the percentage of solutions that were reported to be optimal by each agent, but were later discovered to be suboptimal by evaluating \mathcal{H} . A large amount of solutions are eliminated, close to 100%, helping the designer to avoid making a poor decision, and increasing her confidence that the set of optimal solutions found represent well the “true” Pareto frontier. Moreover, we test for duplicated solutions across different aggregation methods, different teams and different agents. The number is small: only 4 in *contemporary*, and none in *base* and *office park*. Hence, we are providing a high coverage of the Pareto frontier for the designer. We show the total number of optimal solutions in Figure 10 (b).

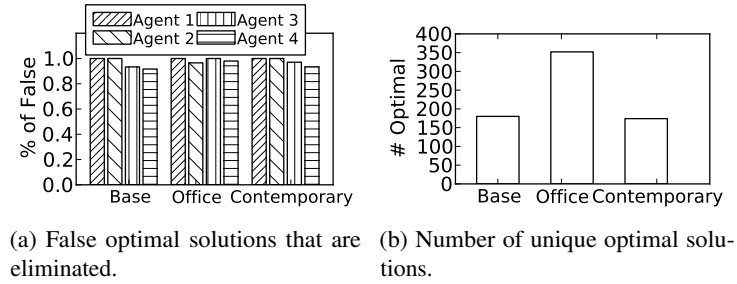


Fig. 10: Additional analysis.

7 Conclusion

Design imposes a novel problem to social choice: maximize the number of optimal solutions. We present a new model for agent teams, that shows the potential of voting agents to be *creative*, by generating a large number of optimal solutions to the designer. Our analysis, which builds a new connection with *number theory*, shows that: (i) uniform teams are in general suboptimal, and converge to a unique solution; (ii) diverse teams are optimal as long as the team’s size grows *carefully*; (iii) the minimum optimal team size is constant with high probability; (iv) the worst case for teams is a prime number of optimal actions. Experiments considered bounded time and relaxed assumptions. We also show results in architecture, where teams find a large number of solutions.

Acknowledgments: This research is supported by MURI grant W911NF-11-1-0332, and the National Science Foundation under grant 1231001.

References

1. Caragiannis, I., Procaccia, A.D., Shah, N.: When do noisy votes reveal the truth? In: EC. pp. 143–160. ACM, New York, NY, USA (2013)
2. Conitzer, V., Sandholm, T.: Common voting rules as maximum likelihood estimators. In: UAI. pp. 145–152 (2005)

3. Echenagucia, T.M., Capozzoli, A., Cascone, Y., Sassone, M.: The early design stage of a building envelope. *Applied Energy* 154, 577–591 (2015)
4. Elkind, E., Shah, N.: Electing the most probable without eliminating the irrational: Voting over intransitive domains. In: *UAI* (2014)
5. Erhan, H., Wang, I., Shireen, N.: Interacting with thousands: A parametric-space exploration method in generative design. In: *ACADIA* (2014)
6. Gerber, D.J., Lin, S.H.E.: Designing in complexity: Simulation, integration, and multidisciplinary design optimization for architecture. *Simulation* (Apr 2013)
7. Gero, J., Sosa, R.: Complexity measures as a basis for mass customization of novel designs. *Environment and Planning B: Planning and Design* 35(1), 3–15 (2008)
8. Globa, A., Donn, M., Moloney, J.: Abstraction versus cased-based: A comparative study of two approaches to support parametric design. In: *ACADIA* (2014)
9. Haynes, G.A.: Testing the boundaries of the choice overload phenomenon. *Psychology & Marketing* 26(3), 204–212 (2009)
10. Iyengar, S., Lepper, M.: When choice is demotivating: Can one desire too much of a good thing? *Journal of Personality and Social Psychology* 79, 995–1006 (2000)
11. Jiang, A.X., Marcolino, L.S., Procaccia, A.D., Sandholm, T., Shah, N., Tambe, M.: Diverse randomized agents vote to win. In: *NIPS* (2014)
12. Kalech, M., Kraus, S., Kaminka, G.A., Goldman, C.V.: Practical voting rules with partial information. *Journal of Autonomous Agents and Multi-Agent Systems* 22, 151–182 (2011)
13. Knysh, D.S., Kureichik, V.M.: Parallel genetic algorithms: A survey and problem state of the art. *Journal of Computer and Systems Sciences International* 49(4), 579–589 (2010)
14. van Langen, P., Brazier, F.: Design space exploration revisited. *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing* 20, 113–119 (2006)
15. List, C., Goodin, R.E.: Epistemic democracy: Generalizing the Condorcet Jury Theorem. *Journal of Political Philosophy* 9, 277–306 (2001)
16. Mao, A., Procaccia, A.D., Chen, Y.: Better Human Computation Through Principled Voting. In: *AAAI* (2013)
17. Marcolino, L.S., Xu, H., Jiang, A.X., Tambe, M., Bowring, E.: Give a hard problem to a diverse team: Exploring large action spaces. In: *AAAI* (2014)
18. Nurmi, H.: *Comparing Voting Systems*. Springer (1987)
19. Polikar, R.: Ensemble learning. In: Zhang, C., Ma, Y. (eds.) *Ensemble Machine Learning: Methods and Applications*. Springer (2012)
20. Procaccia, A.D., Reddi, S.J., Shah, N.: A maximum likelihood approach for selecting sets of alternatives. In: *UAI* (2012)
21. Smith, B.N., Xu, A., Bailey, B.P.: Improving interaction models for generating and managing alternative ideas during early design work. In: *Graphics Interface Conference* (2010)
22. Snooks, R.: Encoding behavioral matter. In: *Proceedings of the International Symposium on Algorithmic Design for Architecture and Urban Design. ALGODE* (2011)
23. Vehlken, S.: Computational swarming: A cultural technique for generative architecture. *Footprint – Delft Architecture Theory Journal* 15 (2014)
24. Welch, C., Moloney, J., Moleta, T.: Selective interference: Emergent complexity informed by programmatic, social and performative criteria. In: *ACADIA* (2014)
25. Woodbury, R.F., Burrow, A.L.: Whither design space? *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 20, 63–82 (2006)
26. Xia, L., Conitzer, V.: A maximum likelihood approach towards aggregating partial orders. In: *IJCAI* (2011)
27. Zavala, G.R., Nebro, A.J., Luna, F., Coello, C.A.C.: A survey of multi-objective metaheuristics applied to structural optimization. *Structural and Multidisciplinary Optimization* 49 (2014)
28. Zhao, F., Li, G., Yang, C., Abraham, A., Liu, H.: A human–computer cooperative particle swarm optimization based immune algorithm for layout design. *Neurocomputing* 132 (2014)