A Short Introduction to Computational Social Choice*

Yann Chevaleyre[1], Ulle Endriss[2], Jérôme Lang[3] and Nicolas Maudet[1]

[1]LAMSADE, Univ. Paris-Dauphine, France, {chevaley,maudet}@etud.dauphine.fr
[2]ILLC, University of Amsterdam, The Netherlands, ulle@illc.uva.nl
[3]IRIT, Univ. Paul Sabatier and CNRS, France, lang@irit.fr

Abstract. Computational social choice is an interdisciplinary field of study at the interface of social choice theory and computer science, promoting an exchange of ideas in both directions. On the one hand, it is concerned with the application of techniques developed in computer science, such as complexity analysis or algorithm design, to the study of social choice mechanisms, such as voting procedures or fair division algorithms. On the other hand, computational social choice is concerned with importing concepts from social choice theory into computing. For instance, the study of preference aggregation mechanisms is also very relevant to multiagent systems. In this short paper we give a general introduction to computational social choice, by proposing a taxonomy of the issues addressed by this discipline, together with some illustrative examples and an (incomplete) bibliography.

1 Introduction: What is Computational Social Choice?

Social choice theory is concerned with the design and analysis of methods for collective decision making. For a few years now, computer science and artificial intelligence (AI) have been taking more and more of an interest in social choice. There are two main reasons for that, leading to two different lines of research. The first of these is concerned with importing notions and methods from AI for solving questions originally stemming from social choice. The point of departure for this line of research is the fact that most of the work in social choice theory has concentrated on establishing abstract results regarding the existence (or otherwise) of procedures meeting certain requirements, but computational issues have rarely been considered. For instance, while it may not be possible to design a voting protocol that makes it impossible for a voter to cheat in one way or another, it may well be the case that cheating successfully turns out to be a computationally intractable problem, which may therefore be deemed an acceptable risk. This is where AI (and operations research, and more generally computer science) comes into play. Besides the complexity-theoretic analysis of voting protocols, other typical examples for work in computational social choice include the formal specification and verification of social procedures (such as fair

* Some parts of this paper appeared in the proceedings of ECSQARU-2005 [62].

division algorithms) using mathematical logic, and the application of techniques developed in AI and logic to the compact representation of preferences in combinatorial domains (such as negotiation over indivisible resources or voting for committees).

The second line of research within computational social choice goes the other way round. It is concerned with importing concepts and procedures from social choice theory for solving questions that arise in computer science and AI application domains. This is, for instance, the case for managing societies of autonomous software agents, which calls for negotiation and voting procedures. Another example is the application of techniques from social choice to developing page ranking systems for Internet search engines.

All of these are examples for a wider trend towards interdisciplinary research involving all of decision theory, game theory, social choice, and welfare economics on the one hand, and computer science, artificial intelligence, multiagent systems, operations research, and computational logic on the other. In particular, the mutually beneficial impact of research in game theory and computer science is already widely recognised and has lead to significant advances in areas such as combinatorial auctions, mechanism design, negotiation in multiagent systems, and applications in electronic commerce.

The purpose of this paper is to highlight some further areas of successful interdisciplinary research, focussing on the interplay of social choice theory with computer science, and to propose a taxonomy of the issues tackled by this new discipline of computational social choice. There are two distinct lines along which we could classify the topics addressed by computational social choice:

(a) the nature of the social choice problem dealt with; and
(b) the type of formal or computational technique studied.

These two dimensions are independent to some extent. We first give a (non-exhaustive) list of topics falling under (a):

preference aggregation — Aggregating preferences means mapping a collection $P = \langle P_1, \dots, P_n \rangle$ of preference relations (or *profiles*) of individual agents into a *collective* preference relation P^* (which implies circumventing Arrow's impossibility theorem [5] by relaxing one of its applicability conditions). Sometimes we are only concerned with determining a socially preferred alternative, or a subset of socially preferred alternatives rather than a full collective preference relation: a *social choice function* maps a collective profile P into a single alternative, while a *social choice correspondence* maps a collective profile P into an nonempty subset of alternatives. This first topic is less specific than the following ones, which mostly also deal with some sort of preference aggregation, but each in a much more specific context.

voting theory — Voting is one of the most popular ways of reaching common decisions. Researchers in social choice theory have studied extensively the properties of various families of voting rules, but have typically neglected computational issues. A whole panorama of voting rules has been proposed in the literature [14]. We shall only mention a few examples here. A *positional scoring rule* computes a score (a number) for each candidate from each individual preference profile and selects the candidate with the maximum sum of scores. The *plurality rule*, for instance, gives score 1 to the most preferred candidate of each voter and 0 to all others. The *Borda rule* assigns scores from m (the number of candidates) down to 1 to the candidates according to the preference profile of each voter. Another important concept is that of a *Condorcet winner*, *i.e.* a candidate preferred to any other candidate by a strict majority of voters. It is well-known that there are profiles for which no Condorcet winner exists. Obviously, when there exists a Condorcet winner then it is unique. A *Condorcet-consistent rule* is a voting rule electing the Condorcet winner whenever there is one.

resource allocation and fair division — Resource allocation of indivisible goods aims at assigning items from a finite set R to the members of a set of agents N, given their preferences over all possible bundles of goods. In *centralised* allocation problems the assignment is determined by a central authority to which the agents have given their preferences beforehand. In *distributed* allocation problems agents negotiate, communicate their interests, and exchange or trade goods in several rounds, possibly in a multilateral manner. An overview of issues in resource allocation may be found in [19]. We can distinguish two types of criteria when assessing the quality of a resource allocation, namely *efficiency* and *fairness*. The most fundamental efficiency criterion is *Pareto efficiency:* an allocation should be such that there is not alternative allocation that would be better for some agents without being worse for any of the others. An example for a fairness condition is *envy-freeness:* an allocation is envy-free iff no agent would rather obtain the bundle held by one of the others.

coalition formation — In many occasions, agents do not compete but instead cooperate, for instance to fullfill more efficiently a given task. Suppose for instance that agent x is rewarded 10 when he performs a given task alone, while agent y gets 20. Now if they form a team, the gain is up to 50 (think for instance of two musicians, playing either solo or in a duet). Coalition formation studies typically two questions: what and how coalitions will form for a given problem, and how should then the surplus be divided among the members of the coalition (after they have solved their optimisation problem). Central here is the notion of *stability:* an agent should have no incentive to leave the coalition. These questions are studied in the field of cooperative game theory [72], and different solution concepts have been introduced. For instance, the strongest of these, known as the *core*, requires that no other coalition could make its members better off.

judgement aggregation and belief merging — The field of judgement aggregation aims at studying how a group of individuals should aggregate their members' individual judgements on some interconnected propositions into corresponding collective judgements on these propositions. Such aggregation problems occur in many different collective decision-making bodies (especially committees and expert panels).[1] Belief merging is a closely related problem that is concerned with investigating ways to aggregate a number of individual belief bases into a collective one (connections between both problems are discussed by Eckert and Pigozzi [41, 78]).

ranking systems — The so-called "ranking systems" setting is a variation of classical social choice theory where the set of agents and the set of alternatives *coincide*. The most well-known family of such systems are page ranking systems in the context of search engines (and more generally, reputation systems) [4, 92].

As concerns the second dimension of our proposed taxonomy of topics in computational social choice, namely the classification according to the technical issues addressed rather than the nature of the social choice problem itself, here is now an (equally incomplete) list of issues:

- computationally hard aggregation rules;
- social choice in combinatorial domains;
- computational aspects of strategy-proofness and manipulation;
- distributed resource allocation and negotiation;
- communication requirements in social choice;
- logic-based analysis of social procedures.

The rest of the paper is organised according to this second dimension. For each of the items above we give some description of typical problems considered in the literature, together with some pointers to the bibliography.

2 Computationally Hard Aggregation Rules

Many aggregation and voting rules among those that are practically used are computable in linear or quadratic time in the number of candidates (and almost always linear in the number of voters). Therefore, when the number of candidates is small (which is typically the case for political elections where a single person has to be elected), computing the outcome of a voting rule does not require any sophisticated algorithms. However, there are also a few voting rules that are computationally complex. The following ones have been considered from the computational point of view.

Kemeny — Kemeny's aggregation rule consists of aggregating n individual profiles into a collective profile (called *Kemeny consensus*) being closest to the n profiles, with respect to a distance which, roughly speaking, is the

[1] An introduction to judgement aggregation, together with a bibliography, may be found on the website `http://personal.lse.ac.uk/LIST/doctrinalparadox.htm`.

sum, for all agents, of the numbers of pairs of alternatives on which the aggregated profile disagrees with the agent's profile. This aggregation rule can be turned into a voting rule: a Kemeny winner is a candidate ranked first in one of the Kemeny consensus. Computing a Kemeny consensus is NP-hard [9], and deciding whether a given candidate is a Kemeny winner is $\Delta_2^{\mathsf{P}}(O(\log n))$-complete [52]. Its practical computation has also been addressed [35, 23], while other work has focussed on approximating Kemeny's rule in polynomial time [2] .

Slater — Slater's rule aggregates n individual profiles $P_1, \ldots, P_n$ into a collective profile (called Slater ranking) minimising the distance to the majority graph M_P induced by P (M_P is the graph whose vertices are the candidates and that contains the edge $x \rightarrow y$ if and only if a strict majority of voters prefers x to y). Slater's rule is NP-hard, even under the restriction that pairwise ties cannot occur [2, 3, 22]. The computation of Slater rankings has been addressed by Charon and Hudry [18, 56] as well as Conitzer [22], who gives an efficient preprocessing technique for computing Slater rankings by partitioning the set of candidates into sets of "similar" candidates.

Dodgson — In this voting rule, proposed in 1876 by Dodgson (better known as Lewis Carroll), the election is won by the candidate(s) who is (are) "closest" to being a Condorcet winner: each candidate is given a score that is the smallest number of exchanges of adjacent preferences in the voters' preference orders needed to make the candidate a Condorcet winner with respect to the resulting preference orders. Whatever candidate (or candidates, in the case of a tie) has the lowest score is the winner. This problem was shown to be NP-hard by Bartholdi *et al.* [9], and $\Delta_2^{\mathsf{P}}(O(\log n))$-complete by Hemaspaandra *et al.* [50].

Young — The principle of Young's voting rule is similar to Dodgson's, but here the score of a candidate x is the smallest number of voters whose removal makes x a Condorcet winner. Deciding whether x is a winner according to this rule is $\Delta_2^{\mathsf{P}}(O(\log n))$-complete as well [84].

Banks — A Banks winner for a collection of profiles P is the top vertex of any maximal (with respect to inclusion) transitive subtournament of the majority graph M_P. The problem of deciding whether some fixed vertex v is a Banks winner for P is NP-complete [93, 55].

See also [54] for a partial overview of complexity results for preference aggregation problems.

3 Social Choice in Combinatorial Domains

As long as the set of alternatives is small in size, preferences can be represented *explicitly*. That is, we can simply list all alternatives together with their utility or their rank in the preference order. Unfortunately, in many problem domains the set of alternatives has a *combinatorial structure*. A combinatorial domain is a Cartesian product of finite value domains for each one of a set of variables: an alternative in such a domain is a tuple of values. Clearly, the size of such domains grows exponentially with the set of variables and becomes quickly very large, which makes explicit representations and straightforward elicitation and optimisation no longer reasonable. Logical or graphical *compact representation languages* aim at representing preference structures, the size of which would be prohibitive if represented expicitly, in as little space as possible. The literature on preference elicitation and representation for combinatorial domains has been growing fast, and due to the lack of space we omit giving references here. See for instance [33] for an (incomplete) overview of logic-based preference representation languages, together with results about expressivity and spatial efficiency.

When the set of alternatives has a combinatorial structure, aggregation is a computationally hard problem. Moreover, since in that case preferences are often described in a compact representation language, aggregation should ideally operate directly on this language, without generating the individual nor the aggregated preferences explicitly. In what follows, we give some examples for the issues at stake for different types of problem in social choice.

voting — When the set of candidates has a combinatorial structure, even simple voting rules such as plurality and Borda become hard. The computational complexity of some voting procedures when applied to compactly represented preferences has been investigated in [61]; although that paper does not address the question of *how* the outcome can be computed within a reasonable amount of time. One approach would be to decompose the vote into *local* votes on individual variables (or small sets of variables), and then to gather the results. However, "multiple election paradoxes" [15] show that this can lead to suboptimal choices. Suppose, for instance, 100 voters have to decide whether or not to build a swimming pool (S), and whether or not to build a tennis court (T). 49 voters prefer a swimming pool and no tennis court ($S\bar{T}$), 49 voters prefer a tennis court and no swimming pool ($\bar{S}T$) and 2 voters prefer to have both (ST). Voting separately on each of the issues gives the outcome ST, although it received only 2 votes out of 100. The problem is that there is a *preferential dependence* between S and T. A simple idea then would be to exploit *preferential independencies* between variables. The question is to what extent we may use these independencies to *decompose* the computation of the outcome into smaller problems. Unfortunately, several well-known voting rules (such as plurality or Borda) cannot be decomposed, even when the preferential structure is common to all voters. Most of them fail to be decomposable even when all variables are mutually independent for all voters [63].

fair division — In fair division problems for indivisible resources, the set of alternatives is the set of allocations, the number of which grows exponentially with the number of resources. The need for compact representation arises from the following dilemma, formulated by several social choice theorists: either (a) allow agents to express any possible preference relation on the set of all subsets of items, and end up with an exponentially large representation (such as in [53]); or (b) severely restrict the set of expressible preferences, typically by assuming additive seperability between items, and then design procedures where agents express preferences between single items, thus giving up the possibility of expressing, say, complementarities and substitutabilities. This latter approach is the path followed by Brams *et al.* [13] and Demko and Hill [36], for instance. Compact representation and complexity issues for fair division have received little attention until now, apart for recent work by Lipton *et al.* [65], who study approximation schemes for envy-freeness, and Bouveret *et al.* [11, 12], who study the complexity of fair division problems with compactly represented preferences.

judgement aggregation and belief merging — Here the set of alternatives is the set of all possible truth assignments to a given set of propositional variables (in belief merging) or to a given set of propositional formulae (in judgement aggregation). The common point of logic-based merging approaches is that the set of alternatives corresponds to a set of propositional worlds; the logic-based representation of an agent's preferences (or beliefs) then induces a cardinal function (using ranks or distances) on worlds and aggregates these cardinal preferences. Relevant references that explicitly mention some social choice-theoretic issues include [59, 67, 21, 66]. Konieczny *et al.* [58] specifically address complexity issues for distance-based belief merging operators. As for judgement aggregation, computational issues seem to have been neglected do far. However, some authors [70, 37] give necessary and sufficient conditions for collective rationality, expressed in terms of minimal inconsistent subsets, which can be seen a first step towards addressing computational issues of judgement aggregation.

4 Computational Aspects of Strategy-proofness

Manipulating a voting rule consists, for a given voter or coalition of voters, in expressing an insincere preference profile so as to give more chance to a preferred candidate to be elected. Gibbard and Satterthwaite's theorem [47, 88] states that if the number of candidates is at least 3, then any nondictatorial voting procedure is manipulable for some profiles. However, by applying very specific restrictions on the class of allowed preferences, this theorem does not hold any more [68]. More formally, manipulation by a voter is defined as follows: given a collection of profiles of n voters $P = \langle P_1, \ldots, P_n \rangle$, let c be the elected candidate w.r.t. a given voting rule applied on P. We say that a voter j can *manipulate* the voting rule if there exists a profile ${P'}_j$ such that the voting rule applied on

$\langle P_1, \ldots, P_{j-1}, P'{}_j, P_{j+1}, \ldots, P_n \rangle$ elects a candidate $c' \neq c$ and that j ranks c' higher than c. Note that other manipulation schemes have also been studied, in particular manipulation made by the chairman [10], and manipulation by coalition of voters [75].

Let us show an example of manipulation by a voter. Consider three candidates c_1, c_2, c_3 and 5 voters, among which 2 voters have the preference profile $c_1 \succ c_2 \succ c_3$, 2 other voters have the profile $c_2 \succ c_1 \succ c_3$, and that the last voter has the profile $c_3 \succ c_1 \succ c_2$. If the plurality rule is used here, the last voter will have an interest to report an insincere preference profile with c_1 on the top, as his truly preferred candidate c_3 has no chance of winning.

In the general case, since it is theoretically impossible to make manipulation impossible, one can try to make it less efficient or more difficult. Making it less efficient can consist of making as little as possible of the others' votes known to the would-be manipulating voter – which may be difficult in some contexts—this situtation arises in real world elections, as opinion polls often fail to accurately reflect voters real intentions. Making manipulation more *difficult to compute* is a way followed recently by several authors [8, 7, 25, 24, 27], who address the computational complexity of manipulation for several voting rules. For instance, Single Transferable Vote is NP-hard to manipulate by single agents [7]. The line of argument is that if finding a successful manipulation is extremely hard computationally, then the voters will give up trying to manipulate and express sincere preferences. Note that, for once, the higher the complexity, the better.

Moreover, Conitzer and Sandholm [27] have shown that adding a *pre-round* to the voting process, consisting in eliminating half of the candidates by applying a *binary cup* rule, considerably increases the hardness of manipulation Unfortunately, applying a binary cup as a pre-round may eliminate highly ranked candidates, thus dropping interesting properties of the voting rule used afterwards. As an attempt to overcome this drawback, Elkind and Lipmaa [42] introduced a principle called *hybridization* generalizing the method of [27]. A hybridized voting rule $Hyb(X_k, Y)$ consists of k steps of rule X, followed by rule Y. They study the impact of hybridization on the complexity of manipulation in various cases (including hybridizing a voting rule with itself).

As recently noted by Conitzer and Sandholm [31], computational hardness concepts such as NP- hardness or PSPACE-hardness are *worst case* settings. Thus, they only ensure that there exist cases in which manipulation gets hard to compute. In fact, these authors showed that under some mild assumptions, there are *no* voting rules that are hard to manipulate on average. To obtain this result, the authors first exhibit an algorithm which can be used by individual voters to compute an insincere profile, and then show that this algorithm succeeds in manipulating the vote on a large fraction of the instances.

We end up this Section by briefly mentioning the existence of complexity results for manipulation by the chairman [10, 51] and bribery in elections [46].

5 Distributed Resource Allocation and Negotiation

In recent years, concepts from social choice theory have become more and more salient in computer science research, in particular on topics such as distributed systems, multiagent systems, grid computing, and electronic commerce. Many of the issues addressed in these areas can be modelled in terms of negotiation between autonomous agents. In the case of grid computing, for instance, access to scarce computing resources may be allocated dynamically and in response to specific needs. Naturally, game theory provides the foundations for investigating the strategic aspects of such scenarios, while preference aggregation mechanisms originating in social choice theory may be used to identify socially desirable outcomes of negotiation.

As discussed already in the introduction, we can distinguish two types of criteria when assessing an allocation of resources: criteria pertaining to the *efficiency* of an allocation and those relating to *fairness* considerations. Both of these can often be described in terms of a *social welfare ordering* or a *collective utility function* [69]. In what follows, we give a few examples of efficiency and fairness criteria:

Pareto efficiency — An allocation *Pareto dominates* another allocation, if no agents are worse and some are better off in the former. A *Pareto efficient* allocation is an allocation that is not Pareto dominated by any other allocation. This is the weakest possible efficiency requirement.

utilitarianism — The *utilitarian social welfare* of an allocation is the sum of the individual utilities experienced by the members of society. Asking for maximal utilitarian social welfare is a very strong efficiency requirement; it licenses reallocations that benefit average utility.

egalitarianism — The *egalitarian social welfare* of an allocation is given by the individual utility of the poorest agent in the system. Aiming at maximising this value is an example for a basic fairness requirement. A refinement of this idea is the *leximin* ordering which, informally, works by comparing first the utilities of the least satisfied agents, and when these coincide, compares the utilities of the next least satisfied agents, and so on.

envy-freeness — An agent is said to be envious when it would rather get the bundle of resources allocated to one of the other agents. An allocation is *envy-free* when no agent is envious. If an envy-free allocation is not attainable, it may also be of interest to reduce envy as much as possible (which may, for instance, be measured in terms of the number of envious agents).

Efficiency and fairness criteria are often not compatible. For instance, for a given profile of agent preferences, there may be no allocation that is both Pareto efficient and envy-free. Some work in computational social choice has addressed the computational complexity of checking whether allocations meeting a certain combination of the above criteria exist for a given resource allocation scenario [12]. Complexity results pertaining to efficiency criteria alone have been known for somewhat longer already. Checking whether there exists an allocation such that utilitarian social welfare will exceed a given limit is known to be NP-complete, for instance [85].

Another line of work has been concerned with procedures for finding good allocations. At one end of the spectrum, *combinatorial auctions* are mechanisms for finding an allocation that maximises the revenue of the seller, where this revenue is the sum of the prices the other agents are willing to pay for the bundles allocated to them. Combinatorial auctions have received a lot of attention in recent years [34]; they are a very specific, purely utilitarian class of allocation procedures, in which considerations such as equity and fairness are not relevant. In this context, preference structures are valuation functions (positive and monotonic utility functions). Combinatorial auctions are also *centralised* allocation mechanisms. In *distributed* approaches to resource allocation, on the other hand, allocations emerge as a consequence of individual agents locally agreeing on a sequence of deals to exchange some of the items they currently have in their possession [87, 44]. In the context of distributed resource allocation, an interesting question is under what circumstances convergence to a socially optimal allocation can be guaranteed given certain known facts regarding the criteria used by individual agents to decide whether or not to implement a particular deal. Notions of social optimality considered in this field range from utilitarianism [87], over Pareto optimality and egalitarianism [44], to envy-freeness [20].

As another example for issues in distributed resource allocation and negotiation, we mention some work on establishing the complexity inherent to various allocation procedures. Dunne *et al.* [40] have analysed the computational complexity of decision problems arising in the context of distributed negotiation. For instance, checking whether a given allocation with superior utilitarian social welfare can be reached by means of a sequence of deals over single resources that are rational (in the sense of it being possible to arrange side payments such that both trading partners benefit) is NP-hard (in fact, this result has later been strengthened to a PSPACE-completeness result [39]). A related line of work has been concerned with the *communication complexity* of distributed negotiation mechanisms, analysing upper and lower bounds on the number of deals implemented until an optimal allocation is reached [38, 43].

For a much more thorough survey of research in multiagent resource allocation the reader is referred to [19].

6 Communication Requirements in Social Choice

One area where the interplay between social choice and (theoretical) computer science has been striking in recent years is that of the analysis of social choice problems in terms of their *communication complexity*. In most (if not all) social choice problems, there are some (potentially hard) communication requirements. Even if the procedure is centralised, the center needs at some point to elicit the preferences of the agents involved in the process in order to compute the outcome. Although it is sometimes possible to carefully design protocols that will make this task easier, general results (lower bounds) suggest that it is very often not realistic to rely on that. This in turn is a main motivation to study the problem of social choice under *incomplete knowledge*. We now briefly present a non-exhaustive overview of recent research on these aspects.

The design of protocols that *elicit* the agents' preferences is a key problem. Take the case of a combinatorial auction involving $|R|$ items: fully revealing an agent's preferences would require $2^{|R|} - 1$ bundles to be valued, and that for each of the bidder agents. Now put yourself into the shoes of that auctioneer: you would of course wonder whether you are really obliged to ask that many "value queries". Maybe a sequential approach would ease the process by avoiding unnecessary queries? The key point consists in finding the *relevant* preferences to elicit from the agents: whose preferences are to be elicited about which outcomes? As an example from voting theory, assume that we have 4 candidates A, B, C, D and 9 voters, 4 of which vote $C \succ D \succ A \succ B$, 2 of which vote $A \succ B \succ D \succ C$ and 2 of which vote $B \succ A \succ C \succ D$, the last vote being still unknown. If the plurality rule is chosen then the outcome is already known (the winner is C) and there is no need to elicit the last voter's profile. If the Borda rule is used then the partial scores are $A : 14$, $B : 10$, $C : 14$, $D : 10$; therefore the outcome is not determined. However, we do not need to know the totality of the last vote, but we only need to know whether the last voter prefers A to C or C to A. Can you always design such a clever protocol? Communication complexity may be helpful in answering that question.

Communication complexity [60] is concerned with the problem of determining the amount of information that needs to be exchanged between agents in order to compute a given function f, when the input of that function is distributed among those agents. The computational resources needed to do so are irrelevant here. More technically, the communication complexity is defined as the worst-case of the best protocol that you may find to compute that function. For unstructured problems, it is unlikely that you can do better than the naive upper bound which will consist, for each agent, of revealing his entire input. In some cases however, the combinatorial structure of the problem can be exploited so that the communication burden can be alleviated. Communication complexity offers a bag of techniques that can used to derive lower bounds on communication requirements. Perhaps the most popular of these techniques is the *fooling set*. A fooling set consists of a set of input vectors that would each give the same result to the function, but such that you could somehow mix any pair of vectors and get a different value. A central result says that exhibiting a fooling set of size m guarantees a lower bound of $\log m$ on the communication complexity.

voting — As a first example, we present rather informally the argument advanced by Conitzer and Sandholm [29] that allows to conclude that the communication complexity of the Condorcet voting rule is $\Omega(nm)$, where n is the number of voters and m the number of candidates. In this case, the function f that players have to compute is interpreted as the voting rule that will return the winning candidate, given the vote vector of all the voters. Assume C is the set of candidates. The idea is to construct a set of vote vectors such that the first voter would prefer any candidate of some set $S_i \subseteq C$ to a, and a to any other candidate ($S_i \succ a \succ \overline{S_i}$), while the following would prefer ($\overline{S_i} \succ a \succ S_i$), and so on. Finally, the last voter would prefer a against any other candidate. As one can easily see, a is indeed preferred to any other candidate in that set (by a single vote). There is an exponential number (in nm) of possible such vectors to be constructed. Now this set would indeed be "fooling" iff, for any pair of such vectors, it would be possible to mix the votes of the vectors and obtain a different Condorcet winner. Consider any pair of vote vectors. By construction, there must be a candidate, say b, that is ranked below a by a given voter in one vector of the pair, while being ranked above in the other vector. By replacing that latter vote in the first vote, you would make b preferred by a single vote. This set is indeed a fooling set, whose size allows to derive the lower bound on communication complexity stated above. Conitzer and Sandholm [29] have analysed the communication complexity of several other voting rules, and Segal [90] studies a particular subclass of social choice rules.

coalition formation — As a further example of the use of the fooling set technique, we mention the work of Procaccia and Rosenschein [82] who analyse the communication complexity of coalition formation. More precisely, they analyse the communication complexity of computing the expected payoff of an arbitrary player (not for all the players) before joining a coalition: here again, maybe only limited communication could be sufficient for that player to compute its payoff. This is done in the context of the coalition model proposed by Shehory and Kraus [91], where each agent only knows the resources it initially holds and its own utility function. Procaccia and Rosenschein prove communication results regarding various solution concepts (core, equal excess, Shapley value, etc.). Most of these results show that when the number of agents (n) is not too large, this problem does not involve prohibitive communication costs ($\Omega(n)$).

resource allocation — Let us return to the canonical example of combinatorial auctions discussed before. Here the distributed inputs are the agents' valuations over possible bundles, and the function would return the optimal allocation. Can we do better than those $2^{|R|} - 1$ queries then? In general, the answer is no, in the sense that at least one agent has to reveal its full valuation. Nisan and Segal [71] have shown this, and the communication requirement remains exponential when all valuations are submodular. Only when the valuations of the agents exhibit very specific structures does it become possible to improve on that bound. We refer the reader to the review chapter by Segal [89] for further details on that topic.

In many situations then, the communication complexity will be too heavy a burden to be supported by the agents. For combinatorial auctions, Segal even claims that *"the communication bottleneck appears to be more severe than the computational one"* [89]. One consequence is that the central authority who has to compute the function will often have to deal with *incomplete preferences* (note however that this is not the only reason: it may simply be the case that the agents' preferences are intrinsically incomplete, for instance). Technically, incomplete knowledge about an agent's preferences comes down to partial preferences (*i.e.* partial preorders on the set of alternatives).[2] This in turn raises further interesting questions as to how difficult it is to compute an outcome given incomplete preferences. For instance, the computational complexity of *vote elicitation* has been investigated by Conitzer and Sandholm [26].

A second way of coping incomplete preferences consists of "living" with incompleteness and to consider all complete extensions of the initial incomplete preference profile. More formally, if $\mathcal{R} = \langle R_1, \ldots, R_n \rangle$ is an n-tuple of incomplete preference relations, then define $Ext(\mathcal{R}) = Ext(R_1) \times \ldots \times Ext(R_n)$, where $Ext(R_i)$ is the set of all complete extensions R_i. For a given social choice function f, one can then define $f(\mathcal{R}) = \{f(R'_1, \ldots, R'_n) \mid (R'_1, \ldots, R'_n) \in Ext(\mathcal{R})\}$. In particular, if f is a voting rule, an element of $\bigcup f(\mathcal{R})$ is a "possible winner", whereas an element of $\bigcap f(\mathcal{R})$ is a "necessary winner". For instance, in the voting example presented at the beginning of this section, for the incomplete profile $\mathcal{R}$ consisting of the first 8 votes (with no information on the 9th vote), if f is the plurality rule then C is a necessary winner (and there is no other possible winner); if f is the Borda rule then A and C are the two possible winners (and there is no necessary winner). Because the cardinality of $Ext(\mathcal{R})$ grows exponentially with the number of alternatives, computing possible and necessary winners is generally hard. Some recent work has addressed the computation of possible and necessary winners for several families of voting rules [57, 64, 80]. The problem of strategy-proofness (see also Sect. 4) has been investigated in [79].

Diminishing the amount of information to be transmitted is also of the utmost importance when one considers *privacy* issues in social choice. The work of Brandt and colleagues (see e.g. [16, 17]), in particular, is very representative of this line of research. One example for a significant result is the fact that social choice functions that are non-dictatorial, Pareto-optimal, and monotonic *cannot* be implemented by distributed protocols guaranteeing *unconditional full* privacy (that is, privacy which does not rely either on trusted third parties or computational intractability to protect the agents' preferences).

7 Logic-based Analysis of Social Procedures

A final area of applications of tools familiar from computer science to problems in social choice theory is the use of mathematical logic for the specification and verification, or more generally analysis, of social procedures. In the same way as computer scientists have long been using logic to formally specify the

[2] Note that this interpretation of incomplete preferences is *epistemic*: this has nothing to do with *intrinsic* or *ethical* incompleteness where it does not make sense to compare some alternatives to some others, or it is unethical to do so.

behaviour of computer systems, so as to allow for the automatic verification of certain desirable properties of such systems, suitable logics may be used to specify social procedures such as voting protocols or fair division algorithms. Rohit Parikh [74] has coined the term *social software* for this line of research and argued that (extensions of) dynamic logic [49] may be particularly suited for formalising such social procedures.

In what follows, we briefly discuss three lines of work that are being pursued under the broad heading of social software. This is not an exhaustive list, but it does give a good taste of what kinds of questions are being investigated.

logics for social software — Modal logic is typically the overall framework in which this kind of research is carried out. The most important kind of modal logic for social software is dynamic logic (the logic of programs). Parikh [73] and Pauly [76], amongst others, have proposed various extensions of dynamic logic to account for concepts such as strategies (as in game theory). Another important familiy of modal logics are epistemic logics, which are relevant to social software as they allow us to model the knowledge of the different agents participating ina social mechanism. Dynamic epistemic logic [6] is being applied to study updates of the states of knowledge of these agents. Pauly and Wooldridge [77] also explore the use of logic in context if economic mechanism design. Finally, Agotnes *et al.* [1] have recently proposed a logic for modelling social welfare functions.

specification and verification of social procedures — Once suitable logics have been developed, the central aim of social software is to put these logics to use for the analysis of social procedures. Probably the first such example is Parikh's specification of a cake-cutting algorithm using his game logic based on dynamic logic [73]. Recently, a variant of propositional dynmaic logic has also been used to model some of the results on convergence to a socially optimal allocation by means of distributed negotiation mentioned in Section 5 [45].

coalition formation — Pauly [76] introduces a modal logic (*coalition logic*) to specifically allow reasoning about actions that are undertaken by coalitions of agents (typically more than two agents here, as opposed to the game logic of Parikh [73], which justifies this new modal logic). The logic includes a new modality (*effectivity*), which represents the fact that a group of agents can bring about a given action. The satisfiability problem of the logic lies in PSPACE, which confirms that considering that actions can be brought about by groups of agents increases the complexity of related reasoning problems.

8 Conclusion

In this paper we have given a short (and hence incomplete) survey of some research issues where social choice and computer science can interact. Due to space considerations, many interesting lines of research have only been mentioned in passing or even been omitted altogether. Two such cases are the large body of work on computational aspects of coalition formation [28, 30, 86, 48], and the method of automated mechanism design [32]. In conclusion, computational social choice has by now become a very active area of research, with many important new results being published every year. So while this short survey can only offer a glimpse at current research and is bound to become out of date rather soon, we nevertheless hope to have been able to convey a sense of the types of questions that are being investigated in this exciting new field.

References

1. T. Agotnes, W. van der Hoek, and M. Wooldridge. Towards a logic of social welfare. In *Proceedings of LOFT-2006*, 2006.
2. N. Ailon, M. Charikar, and A. Newman. Aggregating inconsistent information: ranking and clustering. In *Proceedings of STOC-2005*, 2005.
3. N. Alon. Ranking tournaments. *SIAM Journal of Discrete Mathematics*, 20(1–2):137–142, 2006.
4. A. Altman and M. Tennenholtz. Ranking systems: The PageRank axioms. In *Proceedings of EC-2005*, 2005.
5. K. Arrow. *Social Choice and Individual Values.* John Wiley and Sons, 1951. 2nd edition 1963.
6. A. Baltag, L. Moss, and S. Solecki. The logic of public annoucements, common knowledge, and private suspicion. In *Proceedings of TARK-1998*, 1998.
7. J. Bartholdi and J. Orlin. Single transferable vote resists strategic voting. *Social Choice and Welfare*, 8(4):341–354, 1991.
8. J. Bartholdi, C. Tovey, and M. Trick. The computational difficulty of manipulating an election. *Social Choice and Welfare*, 6(3):227–241, 1989.
9. J. Bartholdi, C. Tovey, and M. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 6(3):157–165, 1989.
10. J. Bartholdi, C. Tovey, and M. Trick. How hard is it to control an election? *Mathematical and Computer Modeling*, 16(8/9):27–40, 1992.
11. S. Bouveret, H. Fargier, J. Lang, and M. Lemaître. Allocation of indivisible goods: A general model and some complexity results. In *Proceedings of AAMAS-2005*, 2005.
12. S. Bouveret and J. Lang. Efficiency and envy-freeness in fair division of indivisible goods: logical representation and complexity. In *Proceedings of IJCAI-2005*, 2005.
13. S. Brams, P. Edelman, and P. Fishburn. Fair division of indivisible items. Technical Report RR 2000-15, C.V. Starr Center for Applied Economics, New York University, 2000.
14. S. Brams and P. Fishburn. Voting procedures. In K. Arrow, A. Sen, and K. Suzumura, editors, *Handbook of Social Choice and Welfare*, chapter 4. Elsevier, 2004.
15. S. Brams, D. M. Kilgour, and W. Zwicker. The paradox of multiple elections. *Social Choice and Welfare*, 15:211–236, 1998.
16. F. Brandt. Social choice and preference protection - towards fully private mechanism design. In *Proceedings of EC-2003)*, 2003.
17. F. Brandt and T. Sandholm. Unconditional privacy in social choice. In *Proceedings of TARK-2005*, 2005.

18. I. Charon and O. Hudry. Slater orders and Hamiltonian paths of tournaments. *Electronic Notes in Discrete Mathematics*, 5:60–63, 2000.
19. Y. Chevaleyre, P. E. Dunne, U. Endriss, J. Lang, M. Lemaître, N. Maudet, J. Padget, S. Phelps, J. A. Rodríguez-Aguilar, and P. Sousa. Issues in multiagent resource allocation. *Informatica*, 30:3–31, 2006.
20. Y. Chevaleyre, U. Endriss, S. Estivie, and N. Maudet. Reaching envy-free states in distributed negotiation settings. In *Proceedings of IJCAI-2007*, 2007.
21. S. Chopra, A. Ghose, and T. Meyer. Social choice theory, belief merging, and strategy-proofness. *International Journal on Information Fusion*, 7(1):61–79, 2006.
22. V. Conitzer. Computing Slater rankings using similarities among candidates. In *Proceedings of AAAI-2006*, 2006.
23. V. Conitzer, A. Davenport, and J. Kalagnanam. Improved bounds for computing Kemeny rankings. In *Proceedings of AAAI-2006*, 2006.
24. V. Conitzer, J. Lang, and T. Sandholm. How many candidates are required to make an election hard to manipulate? In *Proceedings of TARK-2003*, 2003.
25. V. Conitzer and T. Sandholm. Complexity of manipulating elections with few candidates. In *Proceedings of AAAI-2002*, 2002.
26. V. Conitzer and T. Sandholm. Vote elicitation: Complexity and strategy-proofness. In *Proceedings of AAAI-2002*, 2002.
27. V. Conitzer and T. Sandholm. Universal voting protocols to make manipulation hard. In *Proceedings of IJCAI-2003*, 2003.
28. V. Conitzer and T. Sandholm. Computing shapley values, manipulating value division schemes, and checking core membership in multi-issue domains. In *AAAI*, pages 219–225, 2004.
29. V. Conitzer and T. Sandholm. Communication complexity of common votiong rules. In *Proceedings of EC-2005*, 2005.
30. V. Conitzer and T. Sandholm. Complexity of constructing solutions in the core based on synergies among coalitions. *Artif. Intell.*, 170(6-7):607–619, 2006.
31. V. Conitzer and T. Sandholm. Nonexistence of voting rules that are usually hard to manipulate. In *Proceedings of AAAI-2006*, 2006.
32. V. Conitzer and T. W. Sandholm. Complexity of mechanism design. In *Proceedings of UAI-2002*, 2002.
33. S. Coste-Marquis, J. Lang, P. Liberatore, and P. Marquis. Expressive power and succinctness of propositional languages for preference representation. In *Proceedings of KR-2004*, 2004.
34. P. Cramton, Y. Shoham, and R. Steinberg, editors. *Combinatorial Auctions*. MIT Press, 2006.
35. A. Davenport and J. Kalagnanam. A computational study of the Kemeny rule for preference aggregation. In *Proceedings of AAAI-2004*, 2004.
36. S. Demko and T. P. Hill. Equitable distribution of indivisible items. *Mathematical Social Sciences*, 16:145–158, 1998.
37. F. Dietrich and C. List. Judgment aggregation by quota rules. *Journal of Theoretical Politics*, 2006. Forthcoming.
38. P. E. Dunne. Extremal behaviour in multiagent contract negotiation. *Journal of Artificial Intelligence Research*, 23:41–78, 2005.
39. P. E. Dunne and Y. Chevaleyre. Negotiation can be as hard as planning: Deciding reachability properties of distributed negotiation schemes. Technical Report ULCS-05-009, Department of Computer Science, University of Liverpool, 2005.
40. P. E. Dunne, M. Wooldridge, and M. Laurence. The complexity of contract negotiation. *Artificial Intelligence*, 164(1–2):23–46, 2005.

41. D. Eckert and G. Pigozzi. Belief merging, judgment aggregation, and some links with social choice theory. In *Belief Change in Rational Agents: Perspectives from Artificial Intelligence, Philosophy, and Economics*, Dagstuhl Seminar Proceedings 05321, 2005.
42. E. Elkind and H. Lipmaa. Hybrid voting protocols and hardness of manipulation. In *Proceedings of ISAAC-2005*, 2005.
43. U. Endriss and N. Maudet. On the communication complexity of multilateral trading: Extended report. *Journal of Autonomous Agents and Multiagent Systems*, 11(1):91–107, 2005.
44. U. Endriss, N. Maudet, F. Sadri, and F. Toni. Negotiating socially optimal allocations of resources. *Journal of Artificial Intelligence Research*, 25:315–348, 2006.
45. U. Endriss and E. Pacuit. Modal logics of negotiation and preference. In *Proceedings of JELIA-2006*, 2006.
46. P. Faliszewski, E. Hemaspaandra, and L. A. Hemaspaandra. The complexity of bribery in elections. In *Proceedings of AAAI-2006*, 2006.
47. A. Gibbard. Manipulation of voting schemes. *Econometrica*, 41:587–602, 1973.
48. T. gotnes, W. van der Hoek, and M. Wooldridge. On the logic of coalitional games. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 153–160, 2006.
49. D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, 2000.
50. E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. Exact analysis of Dodgson elections: Lewis Carroll's 1876 system is complete for parallel access to NP. *JACM*, 44(6):806–825, 1997.
51. E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. Anyone but him: The complexity of precluding an alternative. In *AAAI*, 2005.
52. E. Hemaspaandra, H. Spakowski, and J. Vogel. The complexity of Kemeny elections. Jenaer Schriften zur Mathematik und Informatik, 2003.
53. D. Herreiner and C. Puppe. A simple procedure for finding equitable allocations of indivisible goods. *Social Choice and Welfare*, 19:415–430, 2002.
54. O. Hudry. Computation of median orders: Complexity results. In *Proceedings of the DIMACS-LAMSADE Workshop on Computer Science and Decision Theory*. Annales du LAMSADE 3, 2004.
55. O. 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.
56. O. Hudry. Improvements of a branch and bound method to compute the Slater orders of tournaments. Technical report, ENST, 2006.
57. K. Konczak and J. Lang. Voting procedures with incomplete preferences. In *Proceedings of the Multidisplinary Workshop on Advances in Preference Handling*, 2005.
58. S. Konieczny, J. Lang, and P. Marquis. DA^2 merging operators. *Artificial Intelligence*, 157(1-2):49–79, 2004.
59. S. Konieczny and R. P. Pérez. Propositional belief base merging or how to merge beliefs/goals coming from several sources and some links with social choice theory. *European Journal of Operational Research*, 160(3):785–802, 2005.
60. E. Kushilevitz and N. Nisan. *Communication Complexity*. Cambridge University Press, 1997.
61. J. Lang. Logical preference representation and combinatorial vote. *Annals of Mathematics and Artificial Intelligence*, 42(1):37–71, 2004.
62. J. Lang. Some representation and computational issues in social choice. In *Proceedings of ECSQARU-2005*, 2005.

63. J. Lang. Vote and aggregation in combinatorial domains with structured preferences. In *Proceedings of IJCAI-2007*, 2007.
64. J. Lang, M. Pini, F. Rossi, K. Venable, and T. Walsh. Winner determination in sequential majority voting with incomplete preferences. In *Proceedings of multidisciplinary ECAI06 Workshop about Advances on Preference Handling*, 2006.
65. R. Lipton, E. Markakis, E. Mossel, and A. Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of EC-2004*, 2004.
66. P. Maynard-Zhang and D. Lehmann. Representing and aggregating conflicting beliefs. *Journal of Artificial Intelligence Research*, 19:155–203, 2003.
67. T. Meyer, A. Ghose, and S. Chopra. Social choice, merging, and elections. In *Proceedings of ECSQARU-2001*, 2001.
68. H. Moulin. On strategy-proofness and single peakedness. *Public Choice*, 35:437–455, 1980.
69. H. Moulin. *Axioms of Cooperative Decision Making*. Cambridge University Press, 1988.
70. K. Nehring and C. Puppe. Consistent judgement aggregation: A characterization. Technical report, Univ. Karlsruhe, 2005.
71. N. Nisan and I. Segal. The communication requirements of efficient allocations and supporting prices. *Journal of Economic Theory*, 2006. To appear.
72. M. J. Osborne and A. Rubinstein. *A Course in Game Theory*. MIT Press, 1994.
73. R. Parikh. The logic of games and its applications. *Annals of Discrete Mathematics*, 24:111–140, 1985.
74. R. Parikh. Social software. *Synthese*, 132(3):187–211, 2002.
75. P. K. Pattanaik. On the stability of sincere voting situations. *Journal of Economic Theory*, 6, 1973.
76. M. Pauly. *Logic for Social Software*. PhD thesis, ILLC, University of Amsterdam, 2001.
77. M. Pauly and M. Wooldridge. Logic for mechanism design: A manifesto. In *Proc. 5th Workshop on Game-theoretic and Decision-theoretic Agents*, 2003.
78. G. Pigozzi. Belief merging and the discursive dilemma: an argument-based account to paradoxes of judgment aggregation. *Synthese*, 2007. To appear.
79. M. Pini, F. Rossi, K. Venable, and T. Walsh. Strategic voting when aggregating partially ordered preferences. In *Proceedings of AAMAS-2006*, 2006.
80. M. Pini, F. Rossi, K. Venable, and T. Walsh. Winner determination in sequential majority voting with incomplete preferences. In *Proceedings of multidisciplinary ECAI06 Workshop about Advances on Preference Handling*, 2006.
81. M. S. Pini, F. Rossi, K. Venable, and T. Walsh. Aggregating partially ordered preferences: possibility and impossibility results. In *Proceedings of TARK-2005*, 2005.
82. A. Procaccia and J. S. Rosenschein. The communication complexity of coalition formation among autonomous agents. In *Proceedings of AAMAS-2006*, 2006.
83. F. Rossi, K. Venable, and T. Walsh. mCP nets: representing and reasoning with preferences of multiple agents. In *Proceedings of AAAI-2004*, pages 729–734, 2004.
84. J. Rothe, H. Spakowski, and J. Vogel. Exact complexity of the winner for Young elections. *Theory of Computing Systems*, 36(4):375–386, 2003.
85. M. Rothkopf, A. Pekeč, and R. Harstad. Computationally manageable combinational auctions. *Management Science*, 44(8):1131–1147, 1998.
86. A. Rusinowska, H. de Swart, and J.-W. van der Rijt. A new model of coalition formation. *Social Choice and Welfare*, 24(1):129–154, 2005.
87. T. Sandholm. Contract types for satisficing task allocation: I Theoretical results. In *Proc. AAAI Spring Symposium: Satisficing Models*, 1998.

88. M. Satterthwaite. Strategyproofness and Arrow's conditions. *Journal of Economic Theory*, 10:187–217, 1975.
89. I. Segal. *The Communication Requirements of Combinatorial Allocation Problems.* In Cramton et al. [34], 2006.
90. I. Segal. The communication requirements of social choice rules and supporting budget sets. *Journal of Economic Theory*, 2006. To appear.
91. O. Shehory and S. Kraus. *Coalition Formation among autonomous agents.* Springer-Verlag, 1995.
92. M. Tennenholtz. Transitive voting. In *Proceedings of EC-2004*, 2004.
93. G. J. Woeginger. Banks winners in tournaments are difficult to recognize. *Social Choice and Welfare*, 20(3):523–528, 2003.