A Distributed Protocol for Collective Decision-making in Combinatorial Domains

(Extended Abstract)

Minyi Li
Swinburne University of Technology
Hawthorn, Australia
myli@swin.edu.au

Quoc Bao Vo
Swinburne University of Technology
Hawthorn, Australia
BVO@swin.edu.au

Ryszard Kowalczyk
Swinburne University of Technology
Hawthorn, Australia
RKowalczyk@swin.edu.au

ABSTRACT

In this paper, we study the problem of collective decision-making over combinatorial domains. We focus on a particular social choice rule, namely *Smith/Minimax*. We introduce a distributed protocol for collective decision-making, which is general enough and does not restrict the choice of preference representation languages. The final decision chosen is guaranteed to be a Smith member, and is sufficiently close to the Smith/Minimax candidate. Moreover, it enables distributed decision-making and significantly reduces the amount of dominance testings (individual outcome comparisons) that each agent needs to conduct, as well as the number of pairwise comparisons.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems

Keywords

Collective decision-making; Smith set; Minimax

1. INTRODUCTION

Collective decision-making in combinatorial domains is one of the most active areas in AI. Many of the existing works consider logical or structured preferences, and mainly focus on analysing negative complexity results or finding reasonable restrictions on the preference relations such that sequential winners and direct winners always coincide, e.g., [3, 5, 4]. In this paper, we propose a protocol that enables the agents to identify efficient collective decisions in a distributed manner without explicitly submitting their prefereces. The final chosen collective decision is guaranteed to be a Smith member, [1] and is close to the Smith/Minimax candidate. Moreover, the proposed protocol does not require an exhaustive listing of candidates as employed in a majority of simple aggregation mechanisms [2]. It reduces the total number of alternatives that each agent needs to consider in practice, and requires significantly fewer outcome comparisons compared to an exhaustive search in most decision-making instances. Moreover, the proposed protocol is sufficiently general and applicable to most preference representation languages in combinatorial domains.

[1]We omit all the proofs of the theorems in this paper due to space limitation. A full version containing all detailed proofs can be found at `www.ict.swin.edu.au/personal/myli/DisProCD.pdf`

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013)*, Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.

2. PRELIMINARIES

Let $\mathbf{V} = \{X_1, \ldots, X_m\}$ be a set of m variables in a combinatorial domain, where each variable X takes values in a *local variable domain* $D(X)$. The possible alternatives (outcomes) are $D(X_1) \times \cdots \times D(X_m)$, denoted by O. If $\mathbf{X} = \{X_{\sigma_1}, \ldots, X_{\sigma_\ell}\}$ and $\mathbf{X} \subseteq \mathbf{V}$, with $\sigma_1 < \cdots < \sigma_\ell$ then $D_\mathbf{X}$ denotes $D_{X_{\sigma_1}} \times \cdots \times D_{X_{\sigma_\ell}}$ and $\mathbf{x}$ denotes an assignment of variable values to $\mathbf{X}$, i.e., $\mathbf{x} \in D_\mathbf{X}$. If $\mathbf{X} = \mathbf{V}$, $\mathbf{x}$ is a *complete assignment* (corresponds to an alternative); otherwise $\mathbf{x}$ is called a *partial assignment*. We allow concatenations of disjoint sets of variable values: for instance, let $\mathbf{X} = \{A, B\}$, $\mathbf{Y} = \{C, D\}$, $\mathbf{x} \in D_\mathbf{X}$ and $\mathbf{x} = a\bar{b}$, $\mathbf{y} \in D_\mathbf{Y}$ and $\mathbf{y} = c\bar{d}$, then $\mathbf{xy}\bar{e}$ denotes an assignment $a\bar{b}c\bar{d}\bar{e}$. If $\mathbf{X} \cup \mathbf{Y} = \mathbf{V}$, we call $\mathbf{xy}$ a completion of assignment $\mathbf{x}$. We denote by $\mathtt{Comp}(\mathbf{x})$ the set of completions of $\mathbf{x}$.

A preference order $\succeq$ is a reflexive, transitive and antisymmetric relation on $\mathcal{X}$ (recall that $\succeq$ is antisymmetric iff $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$, $\mathbf{x} \succeq \mathbf{y}$ and $\mathbf{y} \succeq \mathbf{x}$ implies $\mathbf{x} = \mathbf{y}$). $\succ$ denotes the strict relation induced from $\succeq$: $\mathbf{x} \succ \mathbf{y}$ iff $\mathbf{x} \succeq \mathbf{y}$ and *not* $\mathbf{y} \succeq \mathbf{x}$.

Given a set of alternatives O, an aggregation mechanism can be used to aggregate multiple agents' preferences and finally decide on one as the collective decision. In this paper, we consider a Condorcet-consistent method, called *Smith/Minimax*. Smith set is the smallest set of alternatives that every member of the set is pairwise preferred to every alternative not in the set. [2] Minimax (aka. the Simpson method) chooses an option that is defeated by the fewest agents in its worst defeat. The combination of Smith/Minimax is then a decision rule to choose a Minimax candidate in the Smith set.

3. THE PROTOCOL

3.1 Majority decision tree (MDTree)

For a collective decision-making problem over m variables $\mathbf{V} = \{X_1, \ldots, X_m\}$, we conceptualize the assignment of the variable values as a tree, known as the *majority decision tree* (MDTree). A MDTree assigns values to variables level by level. Let k be the maximum size of the variable domain: $\forall X \in \mathbf{V}$, $|D(X)| \leq k$, the MDTree $\mathcal{T}$ is then a k-ary tree. The depth of $\mathcal{T}$ is m with the root being at depth 0. Assume an order over variables $\sigma = X_{\sigma_1} > \cdots > X_{\sigma_m}$ is given for making the collective-decision. Then a MDTree $\mathcal{T}$ is generated following σ, and each level ℓ considers the value assigns to variable X_{σ_ℓ}. A node Φ at depth ℓ represents

[2]Smith set is originally introduced to make group decisions for complete tournament graphs. However, the definitions is easily extended to handle general graphs (incomplete tournaments).

a unique value assignment, denoted by $\texttt{Assg}_\Phi$, to the set of variables $\{X_{\sigma_1}, \ldots, X_{\sigma_\ell}\}$ specified by the path from the root to Φ, i.e., $\texttt{Assg}_\Phi \in D(X_{\sigma_1}) \times \cdots \times D(X_{\sigma_\ell})$. The root node represents an empty assignment, while a node Φ at depth m represents a unique alternative (outcome), as every variable has been assigned a value ($|\texttt{Assg}_\Phi| = m$).

The procedure starts from a root node with an empty assignment. The MDTree $\mathcal{T}$ is then iteratively created in a top-down process following the order $\sigma = X_{\sigma_1} > \cdots > X_{\sigma_m}$, where in each iteration, each agent can propose on a *leaf* node in $\mathcal{T}$. Note that different agents may propose on different nodes (possibly at different depths) in the same iteration. Once a leaf node Φ at depth ℓ ($\ell < m$) receives a majority of the participating agents' proposals, the subtree of Φ will be expanded with possible values assigned to the next variable $X_{\sigma_{\ell+1}}$. The child nodes of Φ will then be explored by the agents and be available for them to consider making proposals on in the next iteration. Notice that we say a node Φ receives a majority of agents' proposals, these proposals may arrive at Φ in different iterations. We formally define *open nodes* and *winning nodes* in a MDTree.

DEFINITION 1 (OPEN NODE). *A node Φ is marked as open if and only if it is a* leaf *node at depth ℓ that $\ell < m$ and Φ receives a majority of agents' proposals.*

DEFINITION 2 (WINNING NODE). *A node Φ is marked as a winning node if and only if it is a* leaf *node at depth $\ell = m$ (i.e., the deepest level), and Φ receives a majority of agents' proposals.*

A winning node, in essence, represents a complete assignment (alternative) that receives a majority of agents' proposals. Notice that once an open node being expanded, it is not an open node any more, as it is no longer a leaf node. Moreover, if a node is marked as a winning node, it will be put in a set $\mathbf{W}$ and may be chosen as a node for the agents to converge. That means, if a winning node Φ^* is chosen from $\mathbf{W}$, the agents will considered it as a *reference point* (i.e., status quo), and an agent will continue to propose on a node Φ in an iteration if and only if it is better than Φ^* (Φ will potentially bring him to a better outcome than $\texttt{Assg}_{\Phi^*}$). While when there is no winning node in the MDTree, $\Phi^* = Null$. This can be considered as a disagreement or a failure of the collective decision-making. We assume that all the agents prefer any alternatives to this failure. Therefore, all the agents will keep making proposals until they identify at least one winning node. And once they have identified one, they try to converge to that node unless they find some other more socially preferred nodes. Finally we know that if a node is marked as a winning node, it will not be feasible for the agents to make proposals on during later iterations. Therefore, in each iteration, a node Φ in $\mathcal{T}$ is feasible for an agent i ($i \in \{1, \ldots, n\}$) to make a proposal on if: *i)* it is a *leaf* node and *not* marked as a winning node; and *ii)* agent i has *not* proposed on Φ. In the following sections, we denote $\mathbf{L}_i$ as the set of leaf nodes that are currently feasible for agent i to make proposals on.

3.2 BPA and BATCD Strategy

In this paper, we consider a distributive implementation and incomplete information settings. In each iteration, the only information that each agent i ($i \in \{1, \ldots, n\}$) obtains is $\mathbf{L}_i$ and a reference point Φ^*. We consider the following optimistic strategy of the agents. At each node $\Phi \in \mathbf{L}_i$, agent i ($i \in \{1, \ldots, n\}$) has a best possible alternative (BPA), denoted by $\texttt{BPA}_i(\Phi)$. It is the optimistic alternative that agent i can obtain from the subtree of Φ, i.e., the best (most preferred) alternative for agent i among the completions of $\texttt{Assg}_\Phi$ ($\texttt{Comp}(\texttt{Assg}_\Phi)$).

Consequently, if $\exists \Phi \in \mathbf{L}_i$, such that $\texttt{BPA}_i(\Phi) \succ_i \texttt{Assg}_{\Phi^*}$, agent i will propose in the current iteration. Moreover, we assume that the agents use an optimistic strategy for making proposals on the MDTree.

DEFINITION 3 (BATCD STRATEGY). *Given a list of feasible leaf nodes $\mathbf{L}_i$ and a reference point Φ^* in the current iteration, agent i ($i \in \{1, \ldots, n\}$) will make a proposal on a node $\Phi \in \mathbf{L}_i$ only if: i) $\forall \Phi' \in \mathbf{L}_i$, $\texttt{BPA}_i(\Phi') \not\succ \texttt{BPA}_i(\Phi)$; and ii) $\Phi^* = Null$ or* $\texttt{BPA}_i(\Phi) \succ_i \texttt{Assg}_{\Phi^*}$

4. THEORETICAL PROPERTIES

To prove the final chosen decision using the proposed protocol is a member of the Smith set, we first have the following proposition and corollary.

PROPOSITION 1. *For an individual agent i ($i \in \{1, \ldots, n\}$), given a* winning *node Φ^* and a node Φ (not necessarily at depth m), if $\texttt{BPA}_i(\Phi) \not\succ_i \texttt{Assg}_{\Phi^*}$ (agent i does not prefer any BPA of Φ to $\texttt{Assg}_{\Phi^*}$), then: $\forall o \in \texttt{Comp}(\texttt{Assg}_\Phi)$, $o \not\succ_i \texttt{Assg}_{\Phi^*}$.*

COROLLARY 1. *For a majority of agents $\mathbf{A}^* \subseteq \{1, \ldots, n\}$, given a* winning *node Φ^* and a node Φ (not necessarily at depth m), if $\forall i \in \mathbf{A}^*, \texttt{BPA}_i(\Phi) \not\succ_i \texttt{Assg}_{\Phi^*}$, then: $\forall o \in \texttt{Comp}(\texttt{Assg}_\Phi)$, $o \not\succ_{maj} \texttt{Assg}_{\Phi^*}$.*

THEOREM 1. *The final collective decision $\texttt{Assg}_{\Phi^*}$ chosen by the proposed protocol is a member of the Smith set.*

Except for the theoretical properties, we've also run experiments with the preference language "SLO SCPnet" [1] (the reader is referred to the full verson of the paper). In those experiments, we observe that, in more than 99.8% cases the final decision chosen by the proposed protocol is the Smith/Minimax candidate. Moreover, even in the cases that our protocol fails to choose a Smith/Minimax candidate, the maximum defeats of the alternative chosen is sufficiently close to that of a Smith/Minimax candidate.

5. CONCLUSION

In this paper, we proposed a very general distributed protocol for collective decision-making in combinatorial domains. The proposed protocol satisfied Smith criterion and the final outcome chosen is sufficiently close to the Smith/Minimax candidate. It works under incomplete information and significantly reduces the search space.

An important issue for future research is manipulation: whether the protocol is strategy-proof, or an agent can get a better alternative by not using the BATCD strategy. Also, it is important to investigate the feasibility of the proposed protocol when applied to different representation languages with different computational properties on preference queries.

6. ACKNOWLEDGEMENTS

This work is partially supported by the ARC Discovery Grant DP110103671.

7. REFERENCES

[1] Carmel Domshlak, Steven David Prestwich, Francesca Rossi, Kristen Brent Venable, and Toby Walsh. Hard and soft constraints for reasoning about qualitative conditional preferences. *J. Heuristics*, 12(4-5):263–285, 2006.

[2] Jérôme Lang. From preference representation to combinatorial vote. In *KR*, pages 277–290, 2002.

[3] Jérôme Lang and Lirong Xia. Sequential composition of voting rules in multi-issue domains. *Mathematical Social Sciences*, 57(3):304–324, May 2009.

[4] Lirong Xia, Vincent Conitzer, and Jérôme Lang. Voting on multiattribute domains with cyclic preferential dependencies. In *AAAI*, pages 202–207. AAAI Press, 2008.

[5] Lirong Xia, Jérôme Lang, and Mingsheng Ying. Strongly decomposable voting rules on multiattribute domains. In *AAAI*, pages 776–781. AAAI Press, 2007.