A General Framework for the Logical Representation of Combinatorial Exchange Protocols *

Extended Abstract

Munyque Mittelmann
Université de Toulouse - IRIT
Toulouse, France
munyque.mittelmann@irit.fr

Sylvain Bouveret
Université Grenoble Alpes - LIG
Grenoble, France
sylvain.bouveret@imag.fr

Laurent Perrussel
Université de Toulouse - IRIT
Toulouse, France
laurent.perrussel@irit.fr

ABSTRACT

The goal of this work is to propose a framework for representing and reasoning about the rules governing a combinatorial exchange. Such a framework is at first interest as long as we want to build up digital marketplaces based on auction, a widely used mechanism for automated transactions. Hence the framework should fulfill two requirements: (i) it should enable bidders to express their bids on combinations of goods and (ii) it should allow describing the rules governing some market, namely the legal bids, the allocation and payment rules. To do so, we define a logical language in the spirit of the *Game Description Language*: the *Combinatorial Exchange Description Language* is the first language for describing combinatorial exchange in a logical framework. The contribution is two-fold: first, we illustrate the general dimension by representing different kinds of protocols, and second, we show how to reason about auction properties in this machine-processable language.

KEYWORDS

Logics for Multi-agents, Game Description Language, Auction-based Markets

ACM Reference Format:
Munyque Mittelmann, Sylvain Bouveret, and Laurent Perrussel. 2021. A General Framework for the Logical Representation of Combinatorial Exchange Protocols : Extended Abstract. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 3 pages.

1 INTRODUCTION

Auction-based markets are widely used for automated business transactions. There are numerous variants whether we consider single or multiple goods, single or multiple units, single or double-side [3, 4]. For a fixed set of parameters, the auction protocol, i.e., the bidding, allocation, and payment rules, may also differ. Building intelligent agents that can switch between different auctions and process their rules is a key issue for building automated auction-based marketplaces. To do so, auctioneers should at first describe the rules governing an auction and second allow bidders to express complex bids. The aim of this work is to propose such language with clear semantics enabling us to derive properties. Hereafter, we consider *combinatorial exchanges* which are the most general case for auctions, mixing double and combinatorial variants [6].

In the spirit of the *General Game Playing* [2] where games are described with the help of a logical language, namely the *Game Description Language* (GDL), we propose the *Combinatorial Exchange Description Language* (CEDL) which is based on the Auction Description Language (ADL) [8]. CEDL includes a bidding language that can represent a wide range of auctions from a single-side, single-unit and good auction (as a single-unit Vickrey Auction) to Iterative Combinatorial Exchange [10]. As for GDL and ADL, we propose a precise semantics based on state-transition models, that gives a clear meaning to the properties describing an auction. CEDL embeds the Tree-Based Bidding Language (TBBL) [10], which generalizes known languages such as XOR/OR [9] to combinatorial exchange, where agents should be able to express preferences for both buying and selling goods. To the best of our knowledge, CEDL is the first framework offering a unified perspective on an auction mechanism: (i) representation on how to bid and (ii) representation of the protocol including allocation and payment. Such a framework offers two benefits: (i) with this language, one can represent many kinds of auctions in a compact way and (ii) the precise state-transition semantics can be used to derive key properties.

To the best of our knowledge, almost all contributions on the computational representation of auction-based markets focus on the implementation of the winner determination problem. For instance, Baral and Ulyan (2001) show how a specific auction, namely combinatorial auctions, can be encoded in a logic program. A hybrid approach mixing linear programming and logic programming has been proposed by Lee and Lee (1997): they focus on sealed-bid auctions and show how qualitative reasoning helps to refine the optimal quantitative solutions. The closest contributions to ours are the *Market Specification Language* (MSL) [14] and ADL [8], also based on GDL. They both focus on representing single good auctions through a set of rules and then interpreting an auction-instance with the help of a state-based semantics. MSL is limited to single agent perspective while ADL is not. However, the main limit of both approaches is the absence of a bidding language.

2 COMBINATORIAL EXCHANGE DESCRIPTION LANGUAGE

The *Combinatorial Exchange Description Language* (CEDL) is a framework for specification of auction-based markets and it is composed by a logical language for describing a protocol, denoted $\mathcal{L}_{CEDL}$, the TBBL language [10] for encoding bids and the winner determination, and a state-transition (ST) model. In TBBL, bids are

*The full version of the paper is available on https://arxiv.org/abs/2102.02061

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, Online.

represented as trees, with the leaf nodes representing bids over a single good type. A non-leaf node is a logical operator defining how many of its children nodes should be satisfied.

An ST-Model allows us to represent the key aspects of an auction, at first the initial and terminal states, the legality of bids, and the transitions between states. A path is a sequence of states and joint bids, starting from the initial state and defined according to the transition function. The truth value of a formula $\varphi \in \mathcal{L}_{CEDL}$ is evaluated at a stage of a path under an ST-Model M. We say φ globally true in M, denoted $M \models \varphi$, if φ is true in every stage of every path in M.

A formula φ in $\mathcal{L}_{CEDL}$ is defined by the following BNF grammar:

$$\varphi ::= p \mid initial \mid terminal \mid legal_i(\beta) \mid does_i(\beta) \mid$$
$$\neg\varphi \mid \varphi \wedge \varphi \mid \bigcirc\varphi \mid z < z \mid z > z \mid z = z \mid rest_i(res, \beta)$$

where $p \in \Phi$ is an atomic proposition, $i \in N$ is an agent, $res \in \{buyer, seller, good, unit\}$ is a bid restriction, $\beta \in \mathcal{A}$ is a bid-tree and $z \in \mathcal{L}_z$ is a numerical term.

The numerical terms specified by $\mathcal{L}_z$ describe (i) classical mathematical operations and functions (e.g. sum and maximum), (ii) explicit links with TBBL, such as the value of a bid given a trade, the quantity of units of a good in a trade, and the winning trade given a list of bids. Intuitively, *initial* and *terminal* specify the initial terminal states, resp.; $legal_i(\beta)$ asserts that agent i is allowed to bid β at the current state and $does_i(\beta)$ asserts that agent i bids β at the current state. The formula $\bigcirc\varphi$ means "φ holds at the next state". The formula $rest_i(res, \beta)$ specifies whether the bid β from agent i respects the restriction res. The restriction *buyer* specifies that β cannot have negative quantities or prices. Similarly, the restriction *seller* specifies that β cannot have positive quantities or prices. The restriction *good* states that β should be a leaf node. Finally, the restriction *unit* says any leaf node in β can only demand a single unit from a good type.

We have that if a player bids at a stage in a path, then (i) she does not bid anything else at the same stage and (ii) the bid is legal.

Proposition 1. *Given an ST-Model M, for any agent $i \in N$ and any bid-tree $\beta \in \mathcal{A}$, we have that (i) $M \models does_i(\beta) \rightarrow \neg does_i(\beta')$, for any $\beta' \in \mathcal{A}$ such that $\beta' \neq \beta$; (ii) $M \models does_i(\beta) \rightarrow legal_i(\beta)$.*

3 REPRESENTING AUCTION-BASED PROTOCOLS

Let us illustrate how to represent a protocol in $\mathcal{L}_{CEDL}$, namely the One-Shot Combinatorial Exchange (Figure 1).

In the initial state, the trade and payment are zero for every agent and good (Rule 1). Any state that is not initial is terminal (Rule 2). The proposition *bidRound* helps to distinguish the initial state from the terminal state where no trade or payment were assigned to any agent (*e.g*, when all agents bid *noop*). Once in a terminal state, players can only do *noop*. Otherwise, they can bid any bid-tree $\beta \in \mathcal{A}_{ce}$ (Rules 3 and 4). If a list of bid-trees is the joint action performed in the initial sate, then in the next state each agent receives an individual trade, which is assigned by the WD over the initial allocations and the bid-trees (Rule 5). After performing a bid in the initial state, the payment for an agent will be the value of her trade given her bid (Rule 6). No numerical variable has its value changed after reaching a terminal state (Rule 7). The allocation for an agent is the quantity of goods she initially held plus her trade (Rule 8). Finally, the proposition *bidRound* is always false in the next state (Rule 9).

(1) $initial \rightarrow bidRound \wedge \bigwedge_{i \in N_{ce}}(payment_i = 0 \wedge \bigwedge_{j \in G_{ce}} trade_{i,j} = 0)$
(2) $terminal \leftrightarrow \neg initial$
(3) $terminal \rightarrow legal_i(noop)$, for each $i \in N_{ce}$
(4) $initial \rightarrow legal_i(\beta)$, for each $i \in N_{ce}$, $\beta \in \mathcal{A}_{ce}$
(5) $does(T_1, \ldots, T_n) \wedge initial \rightarrow \bigcirc(\bigwedge_{i \in N_{ce}} trade_i = win_i(T_1, \ldots, T_n, x_{1,1}, \ldots, x_{n,m}))$, for each $(T_1, \ldots, T_n) \in \mathcal{A}^n_{ce}$
(6) $does_i(\beta) \wedge initial \rightarrow \bigcirc payment_i = value_i(\beta, trade_i)$, for each $i \in N_{ce}$, $\beta \in \mathcal{A}_{ce}$
(7) $terminal \wedge y = x \rightarrow \bigcirc y = x$, for each $y \in Y_{ce}$, $x \in I_{ce}$
(8) $alloc_{i,j} = add(x_{i,j}, trade_{i,j})$, for each $i \in N_{ce}$, $j \in G_{ce}$
(9) $\bigcirc\neg bidRound$

Figure 1: A Combinatorial Exchange represented by Σ_{ce}

A mechanism where only the designer can earn revenue satisfies *no-deficit* [7], that is the cumulative payment among the bidders cannot be negative. The One-Shot Combinatorial Exchange, whose semantics is evaluated by the ST-model M_{ce}, satisfies no-deficit.

Theorem 1. $M_{ce} \models add(payment_1, \ldots, payment_n) \geq 0$.

A mechanism is individually rational if agents can always achieve at least as much utility as from participating as without participating [11]. We say a ST-model M is *Individual Rational* (IR_i) for each agent $i \in N$ if in each path δ in M and stages t in δ, there is a path δ' in M with the prefix $\delta[t] = \delta'[t]$, s.t. i's utility in $\delta'[t+1]$ is at least her utility in $\delta[t+1]$. We can show that this is the case for M_{ce}.

Theorem 2. *For each agent $i \in N_{ce}$ and some monotonic valuation ϑ_i over individual trades, $M_{ce} \models IR_i$.*

4 CONCLUSION

In this work, we have presented a unified framework for representing auction protocols. For going further, the first direction is computational complexity. Although the model-checking (MC) problem in ADL is PTIME [8], the winner determination in Combinatorial Auctions (and thus also in Combinatorial Exchange) is known to be NP-complete [12]. We aim to explore how the MC problem in CEDL is affected by these results.

CEDL definitely puts the emphasis on the auctioneer and auction designer. Our second direction is to design a CEDL-based *General Auction Player* (GAP) that can interpret and reason about the rules of an auction-based market. The key difference, when the players' perspective is considered, is the epistemic and strategic aspects: players have to reason about other players' behavior. The epistemic component will allow an agent to bid according to her beliefs about other agents' private values. Our future GAP should then be based on the epistemic extensions of GDL such as GDL-III [13].

ACKNOWLEDGMENTS

This research is supported by the ANR project AGAPE ANR-18-CE23-0013.

REFERENCES

- [1] Chitta Baral and Cenk Uyan. 2001. Declarative Specification and Solution of Combinatorial Auctions Using Logic Programming. In *Logic Programming and Nonmotonic Reasoning*, Thomas Eiter, Wolfgang Faber, and Miros law Truszczyński (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 186–199.
- [2] Michael Genesereth and Michael Thielscher. 2014. *General game playing*. Morgan & Claypool Publishers.
- [3] Paul Klemperer. 1999. Auction Theory: A Guide to the Literature. *Journal of Economic Surveys* 13, 3 (1999), 227–286. https://doi.org/10.1111/1467-6419.00083 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-6419.00083
- [4] Vijay Krishna. 2009. *Auction Theory*. Academic Press. https://doi.org/10.1192/bjp.111.479.1009-a
- [5] H. Geun Lee and R. Lee. 1997. A hybrid approach of linear programming and logic modeling for the market core of sealed bid auctions. *Annals of Operations Research* 75 (1997).
- [6] Benjamin Lubin, Adam I Juda, Ruggiero Cavallo, Sébastien Lahaie, Jeffrey Shneidman, and David C Parkes. 2008. ICE: An expressive iterative combinatorial exchange. *Journal of Artificial Intelligence Research* 33 (2008), 33–77.
- [7] Debasis Mishra and Tridib Sharma. 2018. A simple budget-balanced mechanism. *Social Choice and Welfare* 50, 1 (2018), 147–170.
- [8] Munyque Mittelmann and Laurent Perrussel. 2020. Auction Description Language (ADL): a General Framework for Representing Auction-based Markets. In *ECAI 2020*, G. de Giacomo (Ed.). IOS Press, Santiago de Compostela.
- [9] Noam Nisan. 2000. Bidding and Allocation in Combinatorial Auctions. In *ACM Conference on Electronic Commerce*. 1–12.
- [10] David C Parkes, Ruggiero Cavallo, Nick Elprin, Adam Juda, Sébastien Lahaie, Benjamin Lubin, Loizos Michael, Jeffrey Shneidman, and Hassan Sultan. 2005. ICE: An Iterative Combinatorial Exchange. In *Proceedings of the 6th ACM Conference on Electronic Commerce (EC '05)*. Association for Computing Machinery, New York, NY, USA, 249–258. https://doi.org/10.1145/1064009.1064036
- [11] David Christopher Parkes and Lyle H Ungar. 2001. *Iterative combinatorial auctions: Achieving economic and computational efficiency*. University of Pennsylvania Philadelphia, PA.
- [12] Michael H Rothkopf, Aleksandar Pekeč, and Ronald M Harstad. 1998. Computationally manageable combinational auctions. *Management science* 44, 8 (1998), 1131–1147.
- [13] Michael Thielscher. 2016. GDL-III: A proposal to extend the game description language to general epistemic games. In *Proc. the European Conference on Artificial Intelligence (ECAI)*, Vol. 285. Hague, 1630–1631. https://doi.org/10.3233/978-1-61499-672-9-1630
- [14] Michael Thielscher and Dongmo Zhang. 2010. *From General Game Descriptions to a Market Specification Language for General Trading Agents*. Springer Berlin Heidelberg, 259–274.