Formal Verification of Bayesian Mechanisms

Munyque Mittelmann[1], **Bastien Maubert**[1], **Aniello Murano**[1], **Laurent Perrussel**[2]

[1] Università degli Studi di Napoli “Federico II”, Italy
[2] IRIT - Université Toulouse Capitole, France
{munyque.mittelmann, laurent.perrussel}@ut-capitole.fr, {bastien.maubert, nello.murano}@gmail.com

Abstract

In this paper, for the first time, we study the formal verification of Bayesian mechanisms through strategic reasoning. We rely on the framework of Probabilistic Strategy Logic (PSL), which is well-suited for representing and verifying multi-agent systems with incomplete information. We take advantage of the recent results on the decidability of PSL model checking under memoryless strategies, and reduce the problem of formally verifying Bayesian mechanisms to PSL model checking. We show how to encode Bayesian-Nash equilibrium and economical properties, and illustrate our approach with different kinds of mechanisms.

Introduction

The design of mechanisms for aggregating preferences while achieving a socially desirable outcome is a central problem in Multi-Agent Systems (MAS). For example, an auction is a mechanism that combines bids into a choice of allocation and payments (Klemperer 1999). In recent years, there has been a growing effort at designing novel mechanisms for a wide variety of problems and settings, including peer selection (Aziz et al. 2019), hedonic coalition formation games (Bilò et al. 2018; Flammini, Kodric, and Varricchio 2022), sponsored search auctions (Gatti et al. 2015), and diffusion auctions (Li et al. 2022). The advantages of automating the development and/or analysis of mechanisms are numerous (Okada, Todo, and Yokoo 2019), as these tasks require deep knowledge of mathematics and game theory. Furthermore, the automated approach can yield better mechanisms (in terms of the designing criteria) because it capitalizes on the particulars of the setting, e.g. the probabilistic (or other) information that the designer has about the agents' preferences (Sandholm 2003).

Automated Mechanism Design (AMD) was introduced by Conitzer and Sandholm (2002). In this field, designing mechanisms is usually modelled as a computational optimization problem. Different techniques may be used such as neural networks (Shen, Tang, and Zuo 2019), statistical machine learning (Narasimhan, Agarwal, and Parkes 2016), black-box optimization algorithms (Vorobeychik, Reeves, and Wellman 2007), as well as evolutionary search methods (Niu et al. 2012). All these techniques treat AMD by solving it for a specific setting: no general perspective is considered and consequently typical properties are always defined in terms of the specific problem.

In line with the well established logical approach to system verification (Clarke et al. 2018), the work presented in (Wooldridge et al. 2007) advocates the use of Alternating-time Temporal Logic (ATL) (Alur, Henzinger, and Kupferman 2002) to reason about mechanism design. Due to ATL's limitations regarding the expression of solution concepts (such as Nash equilibria) as well as handling quantitative aspects, recent works proposed the use of variants of Quantitative Strategy Logic ($\mathsf{SL}[\mathcal{F}]$) (Bouyer et al. 2019) for reasoning about mechanisms. In (Maubert et al. 2021; Mittelmann 2022), the authors demonstrate how to represent and verify knowledge-based benchmarks and properties such as efficiency and strategyproofness in Epistemic $\mathsf{SL}[\mathcal{F}]$. Similarly, $\mathsf{SL}[\mathcal{F}]$ with natural strategies have been considered for reasoning with bounded recall (Belardinelli et al. 2022). Finally, the automated design of deterministic mechanisms was reduced to $\mathsf{SL}[\mathcal{F}]$-synthesis in (Mittelmann et al. 2022). However, $\mathsf{SL}[\mathcal{F}]$ semantics is deterministic and thus the logic is unable to express probabilistic features, which are essential when considering Bayesian and randomized mechanisms.

This paper shows the potential of using the probabilistic version of Strategy Logic (Probabilistic Strategy Logic, PSL) (Aminof et al. 2019) for AMD. Model-Checking PSL-formulas is decidable in 3-EXPSPACE in the context of memoryless strategy, a common assumption in Mechanism Design. As illustrated by recent work on the subject (Feldman et al. 2022; Varloot and Laraki 2022), randomness and imperfect information are foundational and must be addressed by any formal verification technique for Bayesian mechanisms. Generalizing from the deterministic to the probabilistic setting is challenging due to several aspects. First, the wide and heterogeneous range of settings considered in the literature obscures the path for a general and formal approach to verification. The setting may consider deterministic or randomized mechanisms, incomplete information about agents' types (Bayesian mechanisms), mixed or pure strategies, and direct or indirect mechanisms (iterative protocols). Second, considering Bayesian mechanisms brings out different methods for evaluating a mechanism according to the time-line for revealing the incomplete information as the mechanism is executed. We consider a very general Bayesian framework for mechanism design and show how to capture it with PSL. This allows for automatic verification of a wide class of Bayesian mechanisms through PSL model checking, and motivates further research on applications of logic-based approaches for AMD.

Related Work Logic-based approaches have been widely and successfully applied for probabilistic verification of MAS. For instance, probabilistic model-checking techniques have been used for verifying and analysing a drone swarm system (Lomuscio and Pirovano 2020), negotiation games (Ballarini, Fisher, and Wooldridge 2009), team formation protocols (Chen et al. 2011), and stochastic behaviours in dispersion games (Hao et al. 2012), to name a few. (Gutierrez et al. 2021) investigates rational verification for both non-cooperative games and cooperative games in the qualitative probabilistic setting. (Kwiatkowska et al. 2022) details how verification techniques can be developed for concurrent stochastic games and next implemented in PRISM-games model checker.

Outline The paper is organized as follows; first, we recall PSL. Next, we show how to represent a general notion of Bayesian mechanism as a stochastic transition system. Then, we show how to represent and verify different concepts of equilibrium. Next, we show how classical properties of Bayesian mechanism can be encoded in PSL and verified via model checking.

Preliminaries

Throughout this paper, we fix finite non-empty sets of agents Ag, atomic propositions AP and strategy variables Var. We write $\boldsymbol{o}$ for a tuple of objects $(o_a)_{a\in \mathrm{Ag}}$, one for each agent, and such tuples are called *profiles*. Given a profile $\boldsymbol{o}$ and $a \in \mathrm{Ag}$, we let o_a be agent a's component, and $\boldsymbol{o}_{-a}$ is $(o_b)_{b\neq a}$. Similarly, we let $\mathrm{Ag}_{-a} = \mathrm{Ag} \setminus \{a\}$.

Distributions Let X be a finite non-empty set. A *distribution* over X is a function $d : X \to [0, 1]$ such that $\sum_{x\in X} d(x) = 1$, and $\mathrm{Dist}(X)$ is the set of distributions over X. We write $x \in d$ for $d(x) > 0$. If $d(x) = 1$ for some element $x \in X$, then d is a *point distribution*. If, for $i \in I$, d_i is a distribution over X_i, then, writing $X = \prod_{i\in I} X_i$, the *product distribution* of the d_i is the distribution $d : X \to [0, 1]$ defined by $d(x) = \prod_{i\in I} d_i(x_i)$.

Markov Chains A *Markov chain* M is a tuple (Q, p) where Q is a set of states and $p \in \mathrm{Dist}(Q \times Q)$ is a distribution. The values $p(s, t)$ are called *transition probabilities* of M. A *path* is an infinite sequence of states.

Systems We now introduce the formal models we use to represent Bayesian mechanisms.

Definition 1. A *multi-agent stochastic transition system* (or simply *system*) $\mathcal{G}$ is a tuple $(\mathrm{Ac}, V, v_\emptyset, \delta, \ell, \mathrm{L})$ where (i) Ac is a finite non-empty set of *actions*, (ii) V is a finite non-empty set of *states*, (iii) $v_\emptyset \in V$ is an *initial state*, (iv) $\delta : V \times \mathrm{Ac}^{\mathrm{Ag}} \to \mathrm{Dist}(V)$ is a *transition function*, (v) $\ell : V \to 2^{\mathrm{AP}}$ is a *labelling function*, and (vi) $\mathrm{L} : V \to 2^{\mathrm{Ac}}$ is a *legality function* defining the available actions in each state.

A *joint-action* c is an element of $\mathrm{Ac}^{\mathrm{Ag}}$. We say that $\mathcal{G}$ is *deterministic* (instead of stochastic) if every $\delta(v, c)$ is a point distribution.

Plays A *play* in a system $\mathcal{G}$ is an infinite sequence $\pi = v_0v_1\cdots$ of states such that there exists a sequence $c_0c_1\cdots$ of joint-actions such that $c_i \in \mathrm{L}(v_i)^{\mathrm{Ag}}$ and $v_{i+1} \in \delta(v_i, c_i)$ for every i. We write π_i for v_i.

Strategies A (memoryless) *strategy* is a function $\sigma : V \to \mathrm{Dist}(\mathrm{Ac})$. Let *Str* denote the set of all strategies. A *strategy profile* is a tuple $\boldsymbol{\sigma}$ of strategies, one for each agent. We write σ_a for the strategy of agent a in the strategy profile $\boldsymbol{\sigma}$.

Next we recall *Probabilistic Strategy Logic* (PSL), introduced by Aminof *et al.* (2019).

PSL Syntax

The syntax of PSL is defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists s.\varphi \mid \tau \leq \tau$$
$$\tau ::= c \mid \tau^{-1} \mid \tau - \tau \mid \tau + \tau \mid \tau \times \tau \mid \mathbb{P}_{\boldsymbol{s}}(\psi)$$
$$\psi ::= \varphi \mid \neg\psi \mid \psi \vee \psi \mid \mathbf{X}\psi \mid \psi\mathbf{U}\psi$$

where $p \in \mathrm{AP}$, $s \in \mathrm{Var}$, $c \in \mathbb{Q}$ and $\boldsymbol{s} \in \mathrm{Var}^{\mathrm{Ag}}$.

As usual we write $\varphi \to \varphi'$ for $\neg\varphi \vee \varphi'$, and $\tau = \tau'$ for $\tau \geq \tau' \wedge \tau \leq \tau'$.

PSL Semantics

Formulas of PSL are interpreted over multi-agent stochastic transition systems. Temporal operators next ($\mathbf{X}$) and until ($\mathbf{U}$) have their usual meaning, and strategy quantification $\exists x.\varphi$ means that there exists a strategy such that φ holds, as usual in Strategy Logic (Mogavero et al. 2014). The characteristic feature of PSL consists in terms of the form $\mathbb{P}_{\boldsymbol{s}}(\psi)$, whose intuitive meaning is that if each agent a uses the strategy assigned to s_a then property ψ holds with probability $\mathbb{P}_{\boldsymbol{s}}(\psi)$ in the resulting behaviour of the system.

Probability Space on Outcomes An *outcome* of a strategy profile $\boldsymbol{\sigma}$ and a state v is a play π that starts with v and is extended by $\boldsymbol{\sigma}$, i.e., $\pi_0 = v$, and for every $k \geq 0$ there exists $c_k \in \boldsymbol{\sigma}(\pi_k)$ such that $\pi_{k+1} \in \delta(\pi_k, c_k)$. The set of outcomes of a strategy profile $\boldsymbol{\sigma}$ and state v is denoted $out(\boldsymbol{\sigma}, v)$. A given system $\mathcal{G}$, strategy profile $\boldsymbol{\sigma}$, and state v induce an infinite-state Markov chain $M_{\boldsymbol{\sigma},v}$ whose states are the finite prefixes of plays in $out(\boldsymbol{\sigma}, v)$. Such finite prefixes of plays are called *histories* and written h, and we let $\mathrm{last}(h)$ denote the last state in h. Transition probabilities in $M_{\boldsymbol{\sigma},v}$ are defined as $p(h, hv') = \sum_{c\in \mathrm{Ac}^{\mathrm{Ag}}} \boldsymbol{\sigma}(h)(c) \times \delta(\mathrm{last}(h), c)(v')$. The Markov chain $M_{\boldsymbol{\sigma},v}$ induces a canonical probability space on its set of infinite paths (Kemeny, Snell, and Knapp 1976), which can be identified with the set of plays in $out(\boldsymbol{\sigma}, v)$. The corresponding measure is denoted $\mu_{\boldsymbol{\sigma},v}$.[1]

Assignments An *assignment* is a partial function $\mathcal{A} : \mathrm{Var} \to$ *Str*. A binding is a function $\beta : \mathrm{Ag} \to \mathrm{Var}$ that maps agents to strategies. If the image of a binding β is contained in the domain of $\mathcal{A}$, then $\mathcal{A} \circ \beta : \mathrm{Ag} \to$ *Str* is a joint strategy. For an assignment $\mathcal{A}$, variable s and strategy σ, $\mathcal{A}[s \mapsto \sigma]$ is

[1] This is a classic construction, see (Berthon et al. 2020).

the assignment that maps s to σ and is otherwise equal to $\mathcal{A}$. Notation $\mathcal{A}[\boldsymbol{s} \mapsto \boldsymbol{\sigma}]$ is defined similarly for a variable profile $\boldsymbol{s} = (s_a)_{a \in \text{Ag}}$ and a strategy profile $\boldsymbol{\sigma} = (\sigma_a)_{a \in \text{Ag}}$.

A variable s is free in a formula if it has a subformula of the form $\mathbb{P}_{\boldsymbol{s}}(\psi)$ where s appears in $\boldsymbol{s}$ but $\mathbb{P}_{\boldsymbol{s}}(\psi)$ is not in the scope of any $\exists s$. A state formula with no free variables is called a *sentence*.

PSL Semantics PSL state formulas are interpreted in a transition system $\mathcal{G}$, an assignment $\mathcal{A} : \text{Var} \to \textit{Str}$ whose domain contains the free variables of the formula, and a state $v \in \mathcal{G}$. We only give the inductive cases for strategy quantification and comparison of expected outcomes, and refer to (Aminof et al. 2019) for the other standard cases.

$$\begin{array}{ll} \mathcal{G}, \mathcal{A}, v \models p & \text{if } p \in \ell(v) \\ \mathcal{G}, \mathcal{A}, v \models \exists s.\varphi & \text{if } \exists \sigma \in \textit{Str}.\ \mathcal{G}, \mathcal{A}[s \mapsto \sigma], v \models \varphi \\ \mathcal{G}, \mathcal{A}, v \models \tau_1 \leq \tau_2 & \text{if } val_{\mathcal{A},v}(\tau_1) \leq val_{\mathcal{A},v}(\tau_2) \end{array}$$

where

- $val_{\mathcal{A},v}(c) = c$ and $val_{\mathcal{A},v}(\tau^{-1}) = (val_{\mathcal{A},v}(\tau))^{-1}$
- $val_{\mathcal{A},v}(\tau \oplus \tau') = val_{\mathcal{A},v}(\tau) \oplus val_{\mathcal{A},v}(\tau')$ for $\oplus \in \{-, +, \times\}$
- $val_{\mathcal{A},v}(\mathbb{P}_{\boldsymbol{s}}(\psi)) = \mu_{\mathcal{A}\circ\beta,v}(\{\pi : \mathcal{G}, \mathcal{A}, \pi, 0 \models \psi\})$, where β is the binding such that $\beta(a) = s_a$ for each $a \in \text{Ag}$[2].

If φ is a sentence, the truth of $\mathcal{G}, \mathcal{A}, v \models \varphi$ does not depend on $\mathcal{A}$. Thus, as usual, we write $\mathcal{G}, v \models \varphi$ to mean that $\mathcal{G}, \mathcal{A}, v \models \varphi$ for some (equivalently all) $\mathcal{A}$.

Model Checking Model Checking problem consists of deciding whether $\mathcal{G}, \mathcal{A}, v \models \varphi$. It is decidable in the memoryless context (Aminof et al. 2019); as usual, perfect recall strategy is not decidable. More precisely, computational complexity is 3-EXPSPACE, w.r.t. the size of the φ and EXPSPACE w.r.t. the size of $\mathcal{G}$.

Bayesian Mechanism Design with PSL

Next, we recall mechanisms and we show how to model them as stochastic systems with the purpose of enabling the verification of properties under incomplete information.

Bayesian Mechanism Design

A mechanism is a game that maps agents' strategies to a collective choice from a finite set of alternatives Alt (Nisan et al. 2007). In the case of randomized mechanisms, strategies are mapped to probability distributions over Alt. As monetary transfers are often a central feature of the decision, we follow (Parkes 2001) and write each alternative as a tuple $(x, \boldsymbol{p})$ where x is a *choice* and $p_a \in \mathbb{Z}$ is the payment for agent a[3]. Each agent a has a *type* $\theta_a \in \Theta_a$ that specifies how she values each possible choice, where Θ_a is a (finite) set of possible types for agent a. The value an agent a with type θ_a attributes to a choice x is defined by the valuation function $\mathsf{v}_a(x, \theta_a) \in \mathbb{Z}$. We assume agents have quasi-linear utility, so that agent a's utility for an alternative $\alpha = (x, \boldsymbol{p})$ is $u_a(\theta_a, \alpha) = \mathsf{v}_a(x, \theta_a) - p_a$. We let $\boldsymbol{\Theta} = \prod_{a \in \text{Ag}} \Theta_a$, and we write $\boldsymbol{\theta} = (\theta_a)_{a \in \text{Ag}} \in \boldsymbol{\Theta}$ for a type profile, which assigns a type θ_a to each agent a.

In Bayesian mechanisms we consider incomplete information over the agents' types (Hartline 2012). The type θ_a of agent a is drawn from a publicly known distribution $d \in \text{Dist}(\Theta_a)$, $d(\theta_a)$ is the a priori probability that agent a has type θ_a.

Agents' strategies They depend on their type, we let $\sigma_a^\theta \in \textit{Str}_a$ denote a strategy of agent a with type θ, and we also write $\sigma_a = (\sigma_a^{\theta^1}, ..., \sigma_a^{\theta^n})$ for a tuple containing one strategy for each possible type of a, where $\Theta_a = \{\theta^1, ..., \theta^n\}$. A strategy σ_a for an agent a is thus a tuple $(\sigma_a^{\theta^1}, ..., \sigma_a^{\theta^n})$. Given a strategy profile $\boldsymbol{\sigma} = (\sigma_a)_{a \in \text{Ag}}$, we let $\boldsymbol{\sigma}(\boldsymbol{\theta}) = (\sigma_a^{\theta_a})_{a \in \text{Ag}}$ denote strategies for all agents when they have types $\boldsymbol{\theta}$, and $d(\boldsymbol{\theta})$ (resp. $d(\boldsymbol{\theta}_{-a}|\theta_a)$) denote the probability that $\boldsymbol{\theta}$ is drawn (resp. $\boldsymbol{\theta}_{-a}$ is drawn given that a has type θ_a).

Social Choice Function In the probabilistic setting, a social choice function maps each possible agents' type profile to a probability distribution over the alternatives.

Definition 2. A (randomized) *social choice function* (SCF) $\mathsf{f} : \boldsymbol{\Theta} \to \text{Dist}(\text{Alt})$ is a function that maps type profiles to probability distributions over the set of alternatives.

Given a type profile $\boldsymbol{\theta}$, we may write $d_{\mathsf{f},\boldsymbol{\theta}}$ instead of $\mathsf{f}(\boldsymbol{\theta})$ for the probability distribution assigned by f to $\boldsymbol{\theta}$.

Mechanism They are similar in spirit to SCFs except that instead of type profiles they map strategy profiles to probability distributions over the set of alternatives.

Definition 3. A (randomized) *mechanism* is a function $\mathcal{M} : \prod_{a \in \text{Ag}} \textit{Str}_a \to \text{Dist}(\text{Alt})$.

Given a strategy profile $\boldsymbol{\sigma}$, we may write $d_{\mathcal{M},\boldsymbol{\sigma}}$ instead of $\mathcal{M}(\boldsymbol{\sigma})$. A *direct mechanism* is a mechanism where the strategy sets are the agents' sets of possible types. A direct mechanism is thus also an SCF (Narahari 2014). In those mechanisms, a strategy of particular interest is the *truth-revelation strategy*, in which the agent reports its real type: given type θ_a for each agent a, the truth-revelation strategy $\hat{\theta}_a$ for agent a is the strategy such that $\hat{\theta}_a(v) = \theta_a$, for any position v.

A *deterministic mechanism* chooses an alternative for each strategy profile. This is the case when, for every strategy profile $\boldsymbol{\sigma}$, the mechanism assigns a point distribution, e.g., $d_{\mathcal{M},\boldsymbol{\sigma}}(\alpha) = 1$ for some alternative α. Deterministic SCFs are captured similarly.

Example 1 (BIN-TAC auction). We give an informal description of the "Buy-It-Now or Take-a-Chance" (BIN-TAC) auction (Celis et al. 2014), a randomized mechanism. In this auction, a good is auctioned with a buy-it-now price r, set relatively high. If a single bidder chooses buy-it now (BIN), she gets the good for price r. If more than one bidder takes the BIN option, a new bidding round is held between those bidders with reserve price[4] r. Then, the winner is the highest bidder and her payment is the second-highest bid. If no one

[2] $\mathcal{G}, \mathcal{A}, \pi, 0 \models \psi$ evaluates path formula ψ in π_0.

[3] Notice that mechanisms without monetary transfers can be handled by setting all agents' payments to zero.

[4] The reserve price denotes the minimum price that the seller is willing to accept.

chooses to BIN, a take-a-chance (TAC) auction is held between all bidders, with a TAC-reserve price r' ($0 \leq r' \leq r$). The TAC auction proceeds by choosing the top h bidders uniformly at random, and if her bid exceeds the reserve r', she wins the auction and pay the maximum of the reserve and the $(\mathrm{h}+1)$-th bid (if it exists). Ties among the h-highest bidders are broken randomly prior to the random allocation.

In our approach, mechanisms are represented by stochastic transition systems which define the possible actions and strategies of the agents and their outcomes: such transition systems should encode agents' choice and payment.

Definition 4 (Mechanism as a system)**.** Assume that AP contains $\{\mathrm{ter}, \mathrm{choice}^x, \mathrm{pay}_a^{p_a} \mid (x, \boldsymbol{p}) \in \mathrm{Alt}, a \in \mathrm{Ag}\}$ where ter specifies whether a state is terminal, choice^x denotes that the choice x was selected, and $\mathrm{pay}_a^{p_a}$ denotes that p_a is the payment chosen for agent a.

Given a system $\mathcal{G} = (\mathrm{Ac}, V, v_\emptyset, \delta, \ell, \mathrm{L})$ over the atomic propositions AP, we say that $\mathcal{G}$ is the system representation of the mechanism $\mathcal{M} : \prod_{a \in \mathrm{Ag}} Str_a \to \mathrm{Dist}(\mathrm{Alt})$ if it satisfies the following:

i every play eventually reaches a *terminal position*, i.e., a sink[5] where proposition ter has value true and al^α holds for exactly one $\alpha \in \mathrm{Alt}$;

ii in all non-terminal positions, ter has value false;

iii for every $\alpha \in \mathrm{Alt}$ and $\boldsymbol{\sigma} \in \prod_{a \in \mathrm{Ag}} Str_a$,

$$\mathcal{G}, \mathcal{A}[\boldsymbol{s} \mapsto \boldsymbol{\sigma}], v_\emptyset \models \mathbb{P}_{\boldsymbol{s}}(\mathbf{F}(\mathrm{ter} \wedge \mathrm{al}^\alpha)) = d_{\mathcal{M},\boldsymbol{\sigma}}(\alpha)$$

where $\mathrm{al}^\alpha := \mathrm{choice}^x \wedge \bigwedge_{a \in \mathrm{Ag}} \mathrm{pay}_a^{p_a}$ is a shortcut denoting that an alternative $\alpha = (x, \boldsymbol{p})$ was chosen.

Example 2 (BIN-TAC auction (cont.)). We now resume the BIN-TAC auction example and show how to represent it as a system. Let $\bar{w}$ be an upper bound for values used in the mechanism. For simplicity, we consider the lower bound to be 0. Assume the type of each bidder is drawn identically and independently from a distribution over $\Theta \subset [0, \bar{w}]$. We let $r \in (0, \bar{w})$ denote the BIN price, $r' \in [0, r]$ be the TAC reserve price, and $\mathrm{h} \in [1, n]$ be the randomization parameter. The set of alternatives is $\mathrm{Alt}_{\text{BIN-TAC}} = \{(x_a, (p_a)_{a \in \mathrm{Ag}}) : a \in \mathrm{Ag} \cup \{\mathrm{none}\}$ & $p_a \in [0, \bar{w}]\}$, where p_a is the payment for a and x_a denotes the winner of the item (or "none"). Define the mechanism $\mathcal{G}_{\text{BIN-TAC}} = (\mathrm{Ac}, V, v_\emptyset, \delta, \ell, \mathrm{L})$ over AP $= \{\mathrm{price}, \mathrm{choice}^x, \mathrm{pay}_a^{p_a}, \mathrm{ter} : a \in \mathrm{Ag}$ & $(x, \boldsymbol{p}) \in \mathrm{Alt}\}$, where:

- $\mathrm{Ac} = \{\mathrm{BIN}, \mathrm{TAC}, \mathrm{bid}_x : x \in \Theta\}$;
- $v \in V$ for each $v = \langle \mathrm{win}, \mathrm{p}, \mathrm{mode}, (\mathrm{act}_a)_{a \in \mathrm{Ag}} \rangle$, where win $\in$ Ag $\cup$ {none} denotes the winner, mode $\in \{\mathrm{init}, \mathrm{BIN}, \mathrm{TAC}, \mathrm{t}\}$, $\mathrm{p} \in [0, \bar{w}]$, and $\mathrm{act}_a \in \mathrm{Ac} \cup \{\mathrm{noop}\}$ denotes agent a's last action;
- The initial position is $v_\emptyset = \langle \mathrm{none}, r, \mathrm{init}, (\mathrm{noop})_{a \in \mathrm{Ag}} \rangle$.
- For each position $v = \langle \mathrm{win}, \mathrm{p}, \mathrm{mode}, (\mathrm{act}_a)_{a \in \mathrm{Ag}} \rangle$, the legality function is as follows: $c \in \mathrm{L}(v)$ if either (i) mode $=$ init and $c \in \{\mathrm{TAC}, \mathrm{BIN}\}$, or (ii) mode $\in \{\mathrm{BIN}, \mathrm{TAC}\}$ and $c \in \mathrm{Ac} \setminus \{\mathrm{BIN}, \mathrm{TAC}\}$, or (iii) mode $=$ t.
- For each position $v = \langle \mathrm{win}, \mathrm{p}, \mathrm{mode}, (\mathrm{act}_a)_{a \in \mathrm{Ag}} \rangle$ and joint action $\boldsymbol{c} = (c_a)_{a \in \mathrm{Ag}}$, transition $\delta(v, \boldsymbol{c})$ is as follows:

[5] A sink is a position that loops for all action profiles.

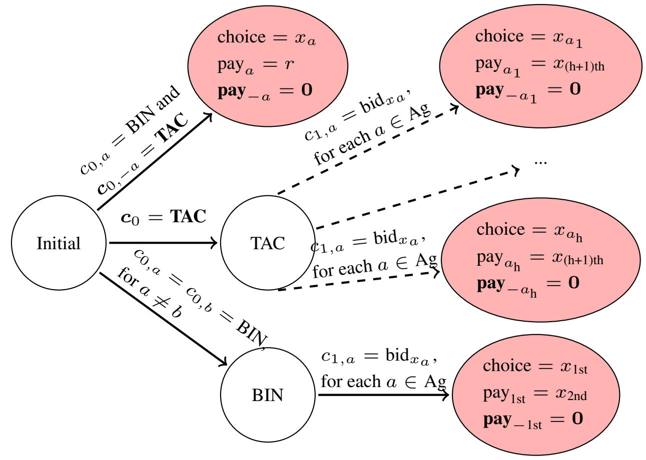

Figure 1: System representation of the BIN-TAC Auction. Red nodes represent terminal positions and specify the alternative chosen. Edges are transitions, with the label specifying the joint actions $\boldsymbol{c}_0$ and $\boldsymbol{c}_1$ performed in each stage. Continuous lines are transitions with probability 1 and dashed lines are transitions with probability $\frac{1}{\mathrm{h}}$. For simplicity, we omit the nodes in which there is no winner (that is, when the highest bid is below the reserve price).

- Let $\mathrm{Ag}_{\mathrm{BIN}} = \{a : a \in \mathrm{Ag}$ & $\mathrm{act}_a = \mathrm{BIN}\}$. If mode $=$ BIN, c_a is in the form bid_{x_a} for each $a \in \mathrm{Ag}_{\mathrm{BIN}}$, and $x_b \geq r$ for some b then, $\delta(v, \boldsymbol{c}) = \langle \mathrm{win}, \mathrm{p}', \mathrm{t}, (c_a)_{a \in \mathrm{Ag}} \rangle$ with probability 1, where win $= a$ if $x_a = \max_{b \in \mathrm{Ag}}(x_b)$ (ties are broken according to a predefined order) and $\mathrm{p}' = \max(r, \max_{b \in \mathrm{Ag}_{\mathrm{BIN}} \setminus \{a : x_a = \max_{a' \in \mathrm{Ag}}(x_{a'})\}}(x_b))$;
- Let $\mathrm{Ag}_{\mathrm{TAC}} = \{a : a \in \mathrm{Ag}\,\&\, \mathrm{act}_a = \mathrm{TAC}\}$. If mode $=$ TAC, c_a is in the form bid_{x_a} for each $a \in \mathrm{Ag}_{\mathrm{TAC}}$ and $x_b \geq r$ for some b, let $\mathrm{Ag}_{\mathrm{TAC}_\mathrm{h}} = \{a :$ x_a is one of the h-highest values in $\{x_a : a \in \mathrm{Ag}_{\mathrm{TAC}}\}$. Then, let $\delta(v, \boldsymbol{c}) = \langle a, \mathrm{p}', \mathrm{t}, (c_a)_{a \in \mathrm{Ag}} \rangle$ with probability $\frac{1}{\mathrm{h}}$ for each $a \in \mathrm{Ag}_{\mathrm{TAC}_\mathrm{h}}$, where $\mathrm{p}' = \max(r', x)$ and x is the $(\mathrm{h}+1)$-highest bid in $\{x_a : a \in \mathrm{Ag}_{\mathrm{TAC}_\mathrm{h}}\}$.
- If mode $=$ BIN (resp., mode $=$ TAC), c_a is in the form bid_{x_a} with $x_a < r$ (resp., $x_a < r'$) for each $a \in \mathrm{Ag}_{\mathrm{BIN}}$, then, $\delta(v, \boldsymbol{c}) = \langle \mathrm{none}, 0, \mathrm{t}, (c_a)_{a \in \mathrm{Ag}} \rangle$ with probability 1;
- The transitions when mode $=$ init are shown in Figure 1. Other cases are sinks, i.e., $\delta(v, \boldsymbol{c}) = v$ with prob. 1.
- For each $v = \langle \mathrm{win}, \mathrm{p}, \mathrm{mode}, (\mathrm{act}_a)_{a \in \mathrm{Ag}} \rangle$ and each $a \in \mathrm{Ag}$, the labeling is defined as follows: $\mathrm{choice}^x \in \ell(v)$ if $x =$ win, $\mathrm{pay}_a^{\mathrm{p}} \in \ell(v)$ if $a =$ win, $\mathrm{pay}_a^0 \in \ell(v)$ if $a \neq$ win, and $\mathrm{ter} \in \ell(v)$ if mode $=$ t.

Bayesian Mechanism Time-Line

The concepts of *ex ante*, *interim*, and *ex post* properties provide a way to consider the nuances between various mechanisms. These terms refer to the time-line of the game (see Figure 2), in relation to the incomplete information about types (Chawla and Sivan 2014). Initially, agents have incomplete information about all agents' type (*ex ante*). Then, each agent realizes her own type, while remaining uncertain

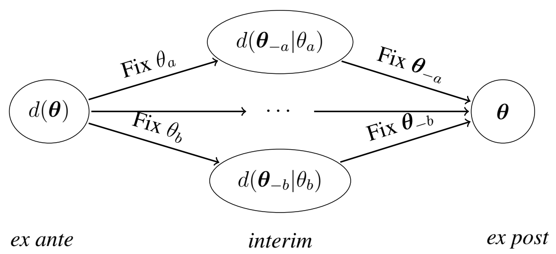

Figure 2: The information disclosure time-line of a mechanism. *Ex ante* properties are evaluated based on the type profile distribution, *interim* properties are calculated using the distributions of type profiles given one agent's type, and *ex post* properties are evaluated given a fixed type profile.

about others' (*interim*). Finally, all types are known to all (*ex post*). In other words, in *ex ante* no type is known, and we thus consider the expectation over all type profiles; in *interim* each agent knows her own type, and we consider the expectation over other agents' types; finally in *ex post*, all types are known, and we only consider the actual types[6].

We now show how to use PSL for evaluating a mechanism in relation to *ex ante*, *interim* and *ex post* properties. First we define the corresponding notions of *expected utility*.

Expected utility We recall the definition of expected utilities for Bayesian games, adapted from (Leyton-Brown and Shoham 2008) to our setting. The expected utility for agent a induced by the mechanism $\mathcal{M}$ given the strategies $\boldsymbol{\sigma}$ and type profile $\boldsymbol{\theta}$ in *ex post* is computed as follows: $\mathsf{E}^{e.p.}_a(\mathcal{M}, \boldsymbol{\sigma}, \boldsymbol{\theta}) := \sum_{\alpha \in \text{Alt}} d_{\mathcal{M}, \boldsymbol{\sigma}(\boldsymbol{\theta})}(\alpha) \times u_a(\theta_a, \alpha)$. Given a type θ_a for agent a, her *interim* expected utility is $\mathsf{E}^{e.i.}_a(\mathcal{M}, \boldsymbol{\sigma}, \theta_a) := \sum_{\boldsymbol{\theta}_{-a} \in \boldsymbol{\Theta}_{-a}} d(\boldsymbol{\theta}_{-a}|\theta_a) \times \mathsf{E}^{e.p.}_a(\mathcal{M}, \boldsymbol{\sigma}, (\boldsymbol{\theta}_{-a}, \theta_a))$. Finally, the *ex ante* expected utility for a is $\mathsf{E}^{e.a.}_a(\mathcal{M}, \boldsymbol{\sigma}) := \sum_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} d(\boldsymbol{\theta}) \times \mathsf{E}^{e.p.}_a(\mathcal{M}, \boldsymbol{\sigma}, \boldsymbol{\theta})$. The *ex post*, *interim* and *ex ante* expected utilities induced by a social choice function are defined analogously and denoted $\mathsf{E}^{e.p.}_a(\mathsf{f}, \boldsymbol{\theta})$, $\mathsf{E}^{e.i.}_a(\mathsf{f}, \theta_a)$, and $\mathsf{E}^{e.a.}_a(\mathsf{f})$, resp.

Let $s(\boldsymbol{\theta}) := (s^{\theta}_a)_{a \in \text{Ag}}$. We write the following PSL-arithmetic terms to capture the *ex ante*, *interim* and *ex post* expected utilities of agent a, respectively, where $\boldsymbol{s}$ is a tuple of strategy variables for each agent:

$$\mathbb{E}^{e.p.}_a(\boldsymbol{s}, \boldsymbol{\theta}) := \sum_{\alpha \in \text{Alt}} u_a(\theta_a, \alpha) \times \mathbb{P}_{\boldsymbol{s}(\boldsymbol{\theta})}(\mathbf{F}(\text{ter} \wedge \text{al}^{\alpha}))$$

$$\mathbb{E}^{e.i.}_a(\boldsymbol{s}, \theta_a) := \sum_{\boldsymbol{\theta}_{-a} \in \boldsymbol{\Theta}_{-a}} d(\boldsymbol{\theta}_{-a}|\theta_a) \times \mathbb{E}^{e.p.}_a(\boldsymbol{s}, (\boldsymbol{\theta}_{-a}, \theta_a))$$

$$\mathbb{E}^{e.a.}_a(\boldsymbol{s}) := \sum_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} d(\boldsymbol{\theta}) \times \mathbb{E}^{e.p.}_a(\boldsymbol{s}, \boldsymbol{\theta})$$

The following result is important as it will lead to define equilibrium in terms of PSL-formulas.

[6]Note that in our setting with probabilistic transitions and strategies, fixing types does not determine a unique outcome, as it does in deterministic mechanisms with pure strategies.

Theorem 1. *Let $\mathcal{G}$ be a system representing a mechanism $\mathcal{M}$, $\boldsymbol{\theta}$ a type profile, $\boldsymbol{\sigma}$ a strategy profile, $\boldsymbol{s}$ a variable profile, and $\mathcal{A}$ an assignment such that for all a, $\mathcal{A}(\boldsymbol{s}_a) = \boldsymbol{\sigma}_a$. Then for all agents a it holds that*

$$\mathcal{G}, \mathcal{A}, v_{\emptyset} \models \mathbb{E}^{e.a.}_a(\boldsymbol{s}, \boldsymbol{\theta}) = \mathsf{E}^{e.a.}_a(\mathcal{M}, \boldsymbol{\sigma}, \boldsymbol{\theta}) \wedge$$
$$\mathbb{E}^{e.i.}_a(\boldsymbol{s}, \theta_a) = \mathsf{E}^{e.i.}_a(\mathcal{M}, \boldsymbol{\sigma}, \theta_a) \wedge \mathbb{E}^{e.p.}_a(\boldsymbol{s}) = \mathsf{E}^{e.p.}_a(\mathcal{M}, \boldsymbol{\sigma})$$

Best Response Equilibrium

A strategy profile $\boldsymbol{\sigma}$ is an ***ex ante*** *best response equilibrium* (BRE$^{\text{e.a.}}$) if for every type profile $\boldsymbol{\theta}$ no agent can increase her expected utility with a unilateral change of strategy given the distributional information about ***all agents' types***. That is, for each agent a, playing her strategy σ_a is the ex-ante best response to the others playing $\boldsymbol{\sigma}_{-a}$.

Let $\boldsymbol{s} = (s_a)_{a \in \text{Ag}}$ be a profile of strategy variables. First, the fact that the strategies represented by $\boldsymbol{s}$ form an *ex ante* best response equilibrium (BRE$^{\text{e.a.}}$) can be expressed in PSL with the formula

$$\text{BRE}^{\text{e.a.}}(\boldsymbol{s}) := \bigwedge_{a \in \text{Ag}} \forall t_a. \, \mathbb{E}^{\text{e.a.}}_a((\boldsymbol{s}_{-a}, t_a)) \leq \mathbb{E}^{\text{e.a.}}_a(\boldsymbol{s})$$

A strategy profile $\boldsymbol{\sigma}$ is a Nash equilibrium (NE) if for every $\boldsymbol{\theta}$ no agent can increase her expected utility with a unilateral change of strategy (Nisan et al. 2007). We let

$$\text{NE}(\boldsymbol{s}) := \bigwedge_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \bigwedge_{a \in \text{Ag}} \forall t_a. \, \mathbb{E}^{e.p.}_a((\boldsymbol{s}_{-a}, t_a), \boldsymbol{\theta}) \leq \mathbb{E}^{e.p.}_a(\boldsymbol{s}, \boldsymbol{\theta})$$

Finally, a strategy profile $\boldsymbol{\sigma}$ is a Bayesian-Nash equilibrium (BNE) if for every player a and every θ_a, σ is the best response that a has to $\boldsymbol{\sigma}_{-a}$ when her type is θ_a, in expectation over the other types $\boldsymbol{\theta}_{-a}$ (Nisan et al. 2007). The following formula expresses that strategy profile $\boldsymbol{s}$ is a BNE:

$$\text{BNE}(\boldsymbol{s}) := \bigwedge_{a \in \text{Ag}, \theta_a \in \Theta_a} \forall t_a. \, \mathbb{E}^{e.i.}_a((\boldsymbol{s}_{-a}, t_a), \theta_a) \leq \mathbb{E}^{e.i.}_a(\boldsymbol{s}, \theta_a)$$

The key difference between BNE and NE is that in BNE an agent's strategy must be a best-response given distributional information about the preferences of other agents, while in NE an agent's strategy must be a best-response to the actual strategies of the other agents (Parkes 2001). Through the rest of this paper, we let $\text{E} \in \{\text{BRE}^{\text{e.a.}}, \text{BNE}, \text{NE}\}$ denote an equilibrium concept. The following theorem shows that checking the existence of such complex equilibria can be rephrased in terms of model checking of PSL-formulas.

Theorem 2. *Let $\mathcal{G}$ be a system representing a mechanism $\mathcal{M}$, $\boldsymbol{\sigma}$ a strategy profile, $\boldsymbol{s}$ a variable profile, and $\mathcal{A}$ an assignment such that for all a, $\mathcal{A}(\boldsymbol{s}_a) = \boldsymbol{\sigma}_a$. Then it holds that*

$$\mathcal{G}, \mathcal{A}, v_{\emptyset} \models E(\boldsymbol{s}) \textit{ iff } \boldsymbol{\sigma} \textit{ is a } E \textit{ in } \mathcal{M}$$

Example 3 (BIN-TAC auction (cont.)). Now we resume Example 2 and we focus on characterizing equilibrium strategies under BIN-TAC, based on the results of (Celis et al. 2014). If multiple agents choose to BIN, it is weakly dominant for agents to bid their valuations. Truth-telling is also weakly dominant in the TAC auction. Taking these auction strategies as given, we turn to the decision whether to buy-it-now or take-a-chance. Following (Celis et al. 2014), in a symmetric equilibrium, the BIN decision takes a threshold form: for some threshold $\bar{\theta}$, agent a with type $\theta_a > \bar{\theta}$ elect to BIN, and the rest do not.

Let the random variable X^j be the j-th highest draw in an independent sample of size $\mathsf{n} - 1$ from a distribution d (the j-th highest rival valuation) and X^* be the maximum of X^{h} and the TAC reserve price r'.

Proposition 1 ((Celis et al. 2014)). *There exists a unique symmetric pure strategy Bayes-Nash equilibrium*[7]*, characterized by a threshold* $\bar{\theta} \in (0, \bar{w}]$.

Let σ_a *be a strategy for agent* a *such that* $\sigma_a(v) = bid_{\hat{\theta}_a(v)}$ *for each* $v \neq v_\emptyset$ *and* $h > 1$ *be a constant. Now, we characterize the action assigned in* $v_\emptyset$*. If* $\bar{w} - p < \frac{1}{h} \times \mathsf{E}[\bar{w} - X^* | X^1 < \bar{w}]$ *then* $\bar{\theta} = \bar{w}$ *and* $\sigma_a(v_\emptyset) = TAC$*. Otherwise,* $\bar{\theta}$ *is the solution of* $\bar{\theta} = p + \frac{1}{h} \times \mathsf{E}[\bar{w} - X^* | X^1 < \bar{\theta}]$*. Then,* $\sigma_a(v_\emptyset) = BIN$ *if* $\theta \geq \bar{\theta}$ *and* $\sigma_a(v_\emptyset) = TAC$ *otherwise. We then have:* $\mathcal{G}_{BIN\text{-}TAC}, \mathcal{A}[\boldsymbol{s} \rightarrow \boldsymbol{\sigma}], v_\emptyset \models BNE(\boldsymbol{\sigma})$.

The intuition is that since strategies have a threshold form, the choice is relevant only when there are no higher-valuation bidders, since these bidders will BIN and win.

Verifying Mechanism Properties

As we are now able to represent mechanisms as stochastic systems and complex Bayesian solution concepts as PSL-formulas, we are in a position for investigating classical Bayesian Mechanism Design properties: we show how evaluating such properties boils down to model checking.

Implementation of an SCF

Given an SCF f, the mechanism design problem consists in defining game rules (the mechanism) to implement f despite agents' individual interests. Game-theoretic concepts are used for analysing the outcome of the mechanism.

In the deterministic setting, a mechanism implements an SCF f if the alternative chosen when agents follow equilibrium strategies is the alternative chosen by f for all possible agent preferences. Different equilibrium concepts may be used to define implementation, including Bayesian-Nash and dominant strategies (Parkes 2001).

We generalize to our setting the definitions of implementation of SCFs from (Parkes 2001; Conitzer and Sandholm 2002). We center this definition on the equilibrium notions introduced above, but the implementation may be defined using some other concept (Parkes 2001).

Definition 5. A mechanism E-implements an SCF f if there exists a strategy profile $\boldsymbol{\sigma}(\boldsymbol{\theta})$ that is an E-equilibrium in the mechanism and $d_{\mathcal{M},\boldsymbol{\sigma}(\boldsymbol{\theta})}(\alpha) = d_{\mathsf{f},\boldsymbol{\theta}}(\alpha)$ for each alternative $\alpha \in \mathrm{Alt}$ and $\boldsymbol{\theta}$.

In the deterministic setting, Definition 5 corresponds to the mechanism implementation from (Parkes 2001). We generalize the definition to take into account randomized mechanisms inline with (Conitzer and Sandholm 2002). To do so, we first introduce the following macros:

$\exists s_a := \exists s_a^{\theta^1} ... \exists s_a^{\theta^n}$, $\forall s_a := \forall s_a^{\theta^1} ... \forall s_a^{\theta^n}$, $\exists \boldsymbol{s} := \exists s_{a_1} ... \exists s_{a_\mathsf{n}}$ where $\Theta_a = \{\theta^1, ..., \theta^n\}$.

Let $\varphi_{\mathsf{f},\boldsymbol{s}}$ be a formula denoting that f assigns the same probability distribution as the mechanism under strategies $\boldsymbol{s}$, for any types:

$$\varphi_{\mathsf{f},\boldsymbol{s}} := \bigwedge_{\boldsymbol{\theta} \in \boldsymbol{\Theta}, \alpha \in \mathrm{Alt}} \mathbb{P}_{\boldsymbol{s}(\boldsymbol{\theta})}(\mathbf{F}\mathrm{ter} \wedge \mathrm{al}_\alpha) = d_{\mathsf{f},\boldsymbol{\theta}}(\alpha)$$

Theorem 3. *Let* $\mathcal{G}$ *be a system representing a mechanism* $\mathcal{M}$*. Given an equilibrium concept* $\mathbf{E} \in \{BRE^{e.a.}, NE, BNE\}$*, assume that* $\mathcal{M}$ $\mathbf{E}$*-implements an SCF* f*. Then it holds that:*

$$\mathcal{G}, v_\emptyset \models \exists \boldsymbol{s}. \, \mathbf{E}(\boldsymbol{s}) \wedge \varphi_{\mathsf{f},\boldsymbol{s}}$$

Example 4 (Dutch auction). In a Dutch auction, the price of a good starts at a high value and is gradually lowered until a bidder accepts the going price. Then she gets the good and the auction ends. The choices in the α include the option of selecting either agent as the winner or not selling the item (in such case the winner is "none"). In each alternative, an agent's payment may be either 0 or the reserve price minus dec times the number of rounds and at most one agent may pay more than 0. The Dutch auction is represented by the mechanism $\mathcal{G}_{\mathrm{dut}}$, defined similarly to Example 2.

This auction is a BNE-implementation of the first price auction (Narahari 2014):

Proposition 2. *Let* $\mathsf{f}_{first} : \boldsymbol{\Theta} \rightarrow Alt_{dut}$ *be a deterministic SCF defined as follows:* $\mathsf{f}_{first}(\boldsymbol{\theta}) = (x, \boldsymbol{p})$ *where* $x = a$ *for the agent such that* $\mathsf{v}_a(a, \theta_a) = \max_{a' \in \mathrm{Ag}} \mathsf{v}_{a'}(a', \theta_{a'})$ *(ties are broken based on the order* $\prec$*),* $p_a = \mathsf{v}_a(a, \theta_a)$ *if* $x = a$*, and* $p_a = 0$ *otherwise. We have that:* $\mathcal{G}_{dut}$ *BNE-implements* f_{first}*, or, equivalently,* $\mathcal{G}_{dut}, v_\emptyset \models \exists \boldsymbol{s}. BNE(\boldsymbol{s}) \wedge \bigwedge_{\boldsymbol{\theta} \in \boldsymbol{\Theta}, \alpha \in Alt} \mathbb{P}_{\boldsymbol{s}(\boldsymbol{\theta})}(\mathbf{F}ter \wedge al_\alpha) = d_{\mathsf{f}_{first},\boldsymbol{\theta}}(\alpha)$.

That is, BNE implementation requires that there exists an (*interim*) best response equilibrium from which, with probability 1, the alternative assigned by the mechanism is the same as the one chosen by the SCF. Similarly, NE implementation requires that there exists an (*ex post*) best response equilibrium from which, with probability 1, the alternative assigned by the mechanism is the same as the one chosen by the SCF.

We are now able to evaluate mechanisms in relation to classical economic properties in *ex ante*, *interim* and *ex post* equilibrium.

Mechanism Properties

Mechanisms are designed with the goal of achieving desirable properties. We illustrate a number of those properties and use them for characterizing a classical mechanism. According to the modeled problem and target properties, different stages of information disclosure in the mechanism may be relevant. For instance, individual rationality (IR) captures the idea that an agent can decide whether or not to participate, in the sense that her expected utility induced by the

[7]A symmetric equilibrium is an equilibrium where all players use the same strategy.

game is no worse then her utility outside the mechanism. The most natural definition of IR is *interim* (Parkes 2001), as it expresses the case that the agent has incentive for participating when she considers her own preferences and has only distributional information about others.

An SCF f is (interim) *individually rational* if for every $\boldsymbol{\theta} \in \boldsymbol{\Theta}$, $\mathsf{E}_a^{e.i.}(\mathsf{f}, \theta_a) \geq 0$ for each agent a. The following formula rephrases individual rationality with respect to some strategy variable $\boldsymbol{s}$ and type profile $\boldsymbol{\theta}$:

$$\mathrm{IR}(\boldsymbol{s}, \boldsymbol{\theta}) := \bigwedge_{a \in \mathrm{Ag}} 0 \leq \mathbb{E}_a^{e.i.}(\boldsymbol{s}, \theta_a)$$

Theorem 4. *Let $\mathcal{G}$ be a system representing a mechanism $\mathcal{M}$. The SCF f E-implemented by $\mathcal{M}$ is (interim) individual rational iff $\mathcal{G}, v_\emptyset \models \exists \boldsymbol{s}.\, \mathbf{E}(\boldsymbol{s}) \wedge \mathbf{F}(ter \wedge \varphi_{\mathsf{f},\boldsymbol{s}} \wedge \bigwedge_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} IR(\boldsymbol{s}, \boldsymbol{\theta}))$.*

In order to collect truthful preferences, the mechanism can be designed to incentivize the agents to tell the truth. In Dominant Strategy Incentive Compatible (DSIC), truth revelation is the best strategy for each agent irrespective of what is reported by the other agents. As DSIC is a strong requirement, a useful relaxation is when truth revelation is the agents' best response in expectation of the types of others (Bayesian Incentive Compatible, BIC) (Narahari 2014).

Let $\mathrm{Ac} = \cup_{a \in \mathrm{Ag}} \Theta_a$. A direct mechanism $\mathcal{M}$ is *Bayesian Incentive Compatible* if the truth-revealing strategy profile $(\hat{\theta}_a)_{a \in \mathrm{Ag}}$ is a BNE in $\mathcal{M}$ for any $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. In other words, BIC holds if the corresponding PSL definition of BNE holds:

Proposition 3. *Let $\mathcal{G}$ be a system representing a mechanism $\mathcal{M}$. $\mathcal{M}$ is Bayesian Incentive Compatible iff for any $\boldsymbol{\theta} \in \boldsymbol{\Theta}$: $\mathcal{G}, \mathcal{A}[\boldsymbol{s} \to (\hat{\theta}_a)_{a \in \mathrm{Ag}}], v_\emptyset \models BNE(\boldsymbol{s})$.*

We illustrate these properties with a classical example:

Example 5. In the dAGVA mechanism (d'Aspremont and Gérard-Varet 1979; Arrow 1979) the allocation rules assigns the choice that maximizes the cumulative of the agents valuations for the (possibly untruthful) reported types, which defines a discount for agent's payment (for her bid ϑ_a) based on the expected total value of the other agents when she bids ϑ_a and other are truthful.

Let $\mathrm{Alt}_{\mathrm{dAGVA}}$ be defined over a finite set of choices C and a finite set of payments in $[w, \bar{w}]$ analogously to Example 2.

Define the mechanism $\mathcal{G}_{\mathrm{dAGVA}} = (\mathrm{Ac}, V, v_\emptyset, \delta, \ell, \mathrm{L})$ over $\mathrm{AP} = \{\mathrm{price}, \mathrm{choice}^x, \mathrm{pay}_a^{p_a}, \mathrm{ter} : a \in \mathrm{Ag} \;\&\; (x, \boldsymbol{p}) \in \mathrm{Alt}_{\mathrm{dAGVA}}\}$, where:

- $\mathrm{Ac} = \{\theta \in \Theta\}$, where $\Theta = \cup_{a \in \mathrm{Ag}} \Theta_a$;
- $v \in V$ for each $v = \langle \mathrm{win}, \mathrm{p} \rangle$, where $\mathrm{win} \in \mathrm{C} \cup \{\mathrm{none}\}$ denotes the winning allocation, and $\mathrm{p}_a \in [w, \bar{w}] \cup \{0\}$ is the payment for a.
- The initial position is $v_\emptyset = \langle \mathrm{none}, (0, ..., 0) \rangle \in V$.
- $\mathrm{L}(v) = \mathrm{Ac}$ for any position $v \in V$.
- For each position $v = \langle \mathrm{win}, \mathrm{p} \rangle$ and joint action $\boldsymbol{c} = (c_a)_{a \in \mathrm{Ag}}$, let $x^*(\boldsymbol{c}) = \mathrm{argmax}_{x' \in \mathrm{C}} \mathsf{v}_b(x', c_b)$ denote the alternative maximizing the reported types [8]. Also let $\mathrm{SW}_{-a}(c_a) = \mathsf{E}_{\boldsymbol{\theta}_{-a}}(\sum_{b \neq a} \mathsf{v}_b(x^*(c_a, \boldsymbol{\theta}_{-a}), \theta_b))$ where the term $\mathsf{E}_{\boldsymbol{c}_{-a}}(.)$ is the expected total value for agents $b \neq a$ when agent a announces θ_a and others tell the truth.
- The transition $\delta(v, \boldsymbol{c})$ is defined as follows:
  - If $\mathrm{win} = \mathrm{none}$, define $\delta(v, \boldsymbol{c}) = \langle \mathrm{win}', \mathrm{p}' \rangle$ with probability 1, where, for each a,

$$\mathrm{win}' = \mathrm{argmax}_{x \in \mathrm{C}}(\sum_{b \in \mathrm{Ag}} \mathsf{v}_b(x, c_b))$$

$$\mathrm{p}'_a = \frac{\sum_{b \neq a} \mathrm{SW}_{-b}(c_b)}{\mathsf{n} - 1} - \mathrm{SW}_{-a}(c_a)$$

  - Other cases are sinks, i.e., $\delta(v, \boldsymbol{c}) = v$ with prob. 1.
- The labelling function ℓ is as for the other examples.

The dAGVA auction is BIC, however, it is not interim IR. Formally, let $\boldsymbol{\theta}$ be a type profile, we have that $\mathcal{G}_{\mathrm{dAGVA}}, \mathcal{A}[\boldsymbol{s} \to (\hat{\theta}_a)_{a \in \mathrm{Ag}}], v \models \mathrm{BNE}(\boldsymbol{s})$ and $\mathcal{G}_{\mathrm{dAGVA}}, \mathcal{A}, v \not\models \exists \boldsymbol{s}.\, \mathrm{BNE}(\boldsymbol{s}) \wedge \mathbf{F}(\mathrm{ter} \wedge \bigwedge_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \mathrm{IR}(\boldsymbol{s}, \boldsymbol{\theta}))$.

Notice that the dAGVA auction is a direct mechanism and, as pointed before, it is an SCF. Thus, the simplification in relation to the complete formula in Theorem 4.

Similarly to IR, we can represent the conditions for other properties from mechanism design, such as efficiency or budget-balance. The above dAGVA auction illustrates how expected social welfare may be encoded, paving the way for going further in checking such new properties.

Discussion

This works addresses the gap between the economics' approach to mechanism design and the well-established techniques for formal reasoning in MAS. We propose the automated verification of Bayesian mechanisms using Probabilistic Strategy Logic (PSL). Unlike previous proposals, we introduce a general approach for evaluating mechanisms, which is able to take into account a wide range of settings (e.g. randomized, indirect and Bayesian mechanisms). Furthermore, thanks to the great expressiveness of the specification language, PSL, the verification *ex ante*, *interim* and *ex post* of complex solution concepts and properties is fully-automated through model checking of logical formulas.

The possibilities for future work are many. Although memoryless strategies are sufficient for most mechanisms, memory can be considered, e.g. for modeling mechanism design with information acquisition (Bikhchandani and Obara 2017). Since model checking PSL with perfect recall is undecidable (Aminof et al. 2019), one possibility is to study the case with bounded memory. As quantitative aspects (such as payments and utilities) are key features of mechanisms, we intend to investigate an extension of PSL with weighted semantics and its application to AMD. A quantitative approach is particularly appealing for considering approximation in Mechanism Design and the assessment of expected optimality (e.g. in relation to efficiency and revenue). Yet another line of research is the automated construction (or *synthesis*) of Bayesian mechanisms from PSL-specifications.

[8] We assume function argmax breaks ties according to a predefined order over the set of choices.

Acknowledgments

This research is partially supported by the ANR project AGAPE ANR-18-CE23-0013, the PRIN project RIPER (No. 20203FFYLK), the JPMorgan AI Faculty Research Award "Resilience-based Generalized Planning and Strategic Reasoning", and the EU ICT-48 2020 project TAILOR (No. 952215).

References

Alur, R.; Henzinger, T.; and Kupferman, O. 2002. Alternating-time temporal logic. *J. ACM*, 49(5): 672–713.

Aminof, B.; Kwiatkowska, M.; Maubert, B.; Murano, A.; and Rubin, S. 2019. Probabilistic Strategy Logic. In *Proc. of IJCAI-19*.

Arrow, K. J. 1979. The Property Rights Doctrine and Demand Revelation under Incomplete Information. In *Economics and Human Welfare*, 23–39. Academic Press.

Aziz, H.; Lev, O.; Mattei, N.; Rosenschein, J. S.; and Walsh, T. 2019. Strategyproof peer selection using randomization, partitioning, and apportionment. *Artificial Intelligence*, 275: 295–309.

Ballarini, P.; Fisher, M.; and Wooldridge, M. 2009. Uncertain Agent Verification through Probabilistic Model-Checking. In *Safety and Security in Multiagent Systems*.

Belardinelli, F.; Jamroga, W.; Malvone, V.; Mittelmann, M.; Murano, A.; and Perrussel, L. 2022. Reasoning about Human-Friendly Strategies in Repeated Keyword Auctions. In *AAMAS-22*.

Berthon, R.; Fijalkow, N.; Filiot, E.; Guha, S.; Maubert, B.; Murano, A.; Pinault, L.; Pinchinat, S.; Rubin, S.; and Serre, O. 2020. Alternating Tree Automata with Qualitative Semantics. *ACM TOCL*, 22(1): 1–24.

Bikhchandani, S.; and Obara, I. 2017. Mechanism design with information acquisition. *Econ. Theory*, 63(3): 783–812.

Bilò, V.; Fanelli, A.; Flammini, M.; Monaco, G.; and Moscardelli, L. 2018. Nash Stable Outcomes in Fractional Hedonic Games: Existence, Efficiency and Computation. *J. Artif. Int. Res.*, 62(1): 315–371.

Bouyer, P.; Kupferman, O.; Markey, N.; Maubert, B.; Murano, A.; and Perelli, G. 2019. Reasoning about Quality and Fuzziness of Strategic Behaviours. In *Proc. of IJCAI 2019*.

Celis, L. E.; Lewis, G.; Mobius, M. M.; and Nazerzadeh, H. 2014. Buy-It-Now or Take-a-Chance: Price Discrimination Through Randomized Auctions. *Manag. Sci.*, 60(12): 2927–2948.

Chawla, S.; and Sivan, B. 2014. Bayesian Algorithmic Mechanism Design. *SIGecom Exch.*, 13(1): 5–49.

Chen, T.; Kwiatkowska, M.; Parker, D.; and Simaitis, A. 2011. Verifying team formation protocols with probabilistic model checking. In *Int. Workshop on Comp. Logic in MAS*.

Clarke, E.; Grumberg, O.; Kroening, D.; Peled, D.; and Veith, H. 2018. *Model checking*. MIT press.

Conitzer, V.; and Sandholm, T. 2002. Complexity of mechanism design. In *Proc. of Uncertainty in AI - 02*.

d'Aspremont, C.; and Gérard-Varet, L.-A. 1979. Incentives and incomplete information. *Journal of Public Economics*, 11(1): 25–45.

Feldman, M.; Gkatzelis, V.; Gravin, N.; and Schoepflin, D. 2022. Bayesian and Randomized Clock Auctions. In *EC '22*, 820–845. ACM.

Flammini, M.; Kodric, B.; and Varricchio, G. 2022. Strategyproof mechanisms for Friends and Enemies Games. *Artificial Intelligence*, 302: 103610.

Gatti, N.; Lazaric, A.; Rocco, M.; and Trovò, F. 2015. Truthful learning mechanisms for multi-slot sponsored search auctions with externalities. *Artificial Intelligence*, 227: 93–139.

Gutierrez, J.; Hammond, L.; Lin, A. W.; Najib, M.; and Wooldridge, M. 2021. Rational Verification for Probabilistic Systems. In *Proc. of KR-21*, 312–322.

Hao, J.; Song, S.; Liu, Y.; Sun, J.; Gui, L.; Dong, J. S.; and Leung, H.-f. 2012. Probabilistic model checking multi-agent behaviors in dispersion games using counter abstraction. In *Proc. of PRIMA-12*.

Hartline, J. D. 2012. Bayesian mechanism design. *Theoretical Computer Science*, 8(3): 143–263.

Kemeny, J. G.; Snell, J. L.; and Knapp, A. W. 1976. Stochastic Processes. In *Denumerable Markov Chains*, 40–57. Springer.

Klemperer, P. 1999. Auction theory: A guide to the literature. *Journal of economic surveys*, 13(3).

Kwiatkowska, M.; Norman, G.; Parker, D.; Santos, G.; and Yan, R. 2022. Probabilistic Model Checking for Strategic Equilibria-based Decision Making: Advances and Challenges. In *Proc. of MFCS 2022*, 4–22.

Leyton-Brown, K.; and Shoham, Y. 2008. *Essentials of Game Theory: A Concise Multidisciplinary Introduction*. Synthesis Lectures on AI and Machine Learning. Morgan & Claypool Publishers.

Li, B.; Hao, D.; Gao, H.; and Zhao, D. 2022. Diffusion auction design. *Artificial Intelligence*, 303: 103631.

Lomuscio, A.; and Pirovano, E. 2020. Parameterised Verification of Strategic Properties in Probabilistic Multi-Agent Systems. In *Proc. of AAMAS-20*, 762–770.

Maubert, B.; Mittelmann, M.; Murano, A.; and Perrussel, L. 2021. Strategic Reasoning in Automated Mechanism Design. In *KR-21*.

Mittelmann, M. 2022. Formally Reasoning about Strategies in Mechanisms. In *OVERLAY@AI*IA*, volume 3311 of *CEUR Workshop Proceedings*, 21–26. CEUR-WS.org.

Mittelmann, M.; Maubert, B.; Murano, A.; and Perrussel, L. 2022. Automated Synthesis of Mechanisms. In *IJCAI-22*.

Mogavero, F.; Murano, A.; Perelli, G.; and Vardi, M. Y. 2014. Reasoning About Strategies: On the Model-Checking Problem. *ACM Trans. Comput. Log.*, 15(4).

Narahari, Y. 2014. *Game Theory and Mechanism Design*. World Scientific.

Narasimhan, H.; Agarwal, S. B.; and Parkes, D. C. 2016. Automated mechanism design without money via machine learning. In *IJCAI-16*.

Nisan, N.; Roughgarden, T.; Tardos, É.; and Vazirani, V. 2007. *Algorithmic Game Theory*. Cambridge Univ. Press.

Niu, J.; Cai, K.; Parsons, S.; Fasli, M.; and Yao, X. 2012. A grey-box approach to automated mechanism design. *Electronic Commerce Research and Applications*, 11(1): 24–35.

Okada, N.; Todo, T.; and Yokoo, M. 2019. SAT-Based Automated Mechanism Design for False-Name-Proof Facility Location. In *PRIMA 2019*.

Parkes, D. 2001. *Iterative combinatorial auctions: Achieving economic and computational efficiency*. Univ. of Pennsylvania Philadel.

Sandholm, T. 2003. Automated mechanism design: A new application area for search algorithms. In *Proc. of CP-03*.

Shen, W.; Tang, P.; and Zuo, S. 2019. Automated Mechanism Design via Neural Networks. In *AAMAS-19*.

Varloot, E. M.; and Laraki, R. 2022. Level-strategyproof Belief Aggregation Mechanisms. In *EC-22*, 335–369. ACM.

Vorobeychik, Y.; Reeves, D. M.; and Wellman, M. P. 2007. Constrained Automated Mechanism Design for Infinite Games of Incomplete Information. In *Conf. on Uncertainty in AI - 07*.

Wooldridge, M.; Agotnes, T.; Dunne, P.; and Van der Hoek, W. 2007. Logic for automated mechanism design-a progress report. In *AAAI-07*.