A Modal Logic for Joint Abilities under Strategy Commitments

Zhaoshuai Liu[1], **Liping Xiong**[2], **Yongmei Liu**[1*],
Yves Lespérance[3], **Ronghai Xu**[1] and **Hongyi Shi**[1]
[1]Dept. of Computer Science, Sun Yat-sen University, Guangzhou 510006, China
[2]Dept. of Computer Science, South China Normal University, Guangzhou 510631, China
[3]Dept. of Electrical Engineering and Computer Science, York University, Toronto, ON M3J 1P3, Canada
liuzhsh6@mail2.sysu.edu.cn, xiongliping@scnu.edu.cn, ymliu@mail.sysu.edu.cn,
lesperan@eecs.yorku.ca, {xurh6, shihy9}@mail2.sysu.edu.cn

Abstract

Representation and reasoning about strategic abilities has been an active research area in AI and multi-agent systems. Many variations and extensions of alternating-time temporal logic ATL have been proposed. However, most of the logical frameworks ignore the issue of coordination within a coalition, and are unable to specify the internal structure of strategies. In this paper, we propose JAADL, a modal logic for joint abilities under strategy commitments, which is an extension of ATL. Firstly, we introduce an operator of elimination of (strictly) dominated strategies, with which we can represent joint abilities of coalitions. Secondly, our logic is based on linear dynamic logic (LDL), an extension of linear temporal logic (LTL), so that we can use regular expressions to represent commitments to structured strategies. We analyze valid formulas in JAADL, give sufficient/necessary conditions for joint abilities, and show that model checking memoryless JAADL is in EXPTIME.

1 Introduction

Representation and reasoning about strategic abilities has been an active research area in AI and multi-agent systems. The foundational work is Alternating-time Temporal Logic ATL/ATL* [Alur *et al.*, 2002] where formula $\langle\langle A \rangle\rangle \phi$ expresses that coalition A has a group strategy to ensure temporal goal ϕ holds no matter what the other agents do. The formulas are interpreted over concurrent game structures where multiple agents act concurrently and the system transition is determined by the collective behavior of agents.

Many variations and extensions of ATL/ATL* have been proposed, *e.g.*, semantic variants about different abilities of agents under perfect vs imperfect information and perfect vs imperfect recall [Schobbens, 2004], explicit quantification over strategies [Chatterjee *et al.*, 2010; Mogavero *et al.*, 2014], probabilistic extensions [Huang *et al.*, 2012], epistemic extensions or variations [van der Hoek and Wooldridge, 2003; Jamroga and van der Hoek, 2004; Naumov and Tao, 2018], resource bounded variations [Alechina *et al.*, 2010], etc.

However, most extensions ignore the *coordination problem*. As discussed in [Ghaderi *et al.*, 2007], a coalition may have many group strategies to ensure a goal, yet a player may not know other players' choices, hence the coalition may end up with a group strategy which may not ensure the goal. They studied the coordination problem and present a formalization of joint ability of coalitions based on the idea of iterated elimination of (strictly) dominated strategies [Osborne and Rubinstein, 1999]. Essentially, a coalition has joint ability to achieve a goal if after iterated elimination of dominated strategies, any remaining joint strategy achieves the goal. Further, [Xiong and Liu, 2016] presented a formalization of joint ability of coalitions under commitments to strategy programs. However, both works use the very expressive situation calculus [Reiter, 2001], making it difficult to analyze properties of the logics. Recently, based on coalition logic [Pauly, 2002], [Hawke, 2017] proposed a logic of joint ability in two-player tacit games with a joint ability modality $((A))\varphi$: two players have joint ability to achieve a goal if after iterated elimination of punishment strategies, *i.e.*, those strategies that fail to achieve the goal no matter what other agents do, any remaining joint strategy achieves the goal. So they only eliminate punishment strategies, hence their concept of joint ability is much weaker than that of [Ghaderi *et al.*, 2007].

Social rules [Shoham and Tennenholtz, 1995], such as traffic rules, play an important role in coordination. [Ågotnes *et al.*, 2007] used normative systems to describe norms/social rules semantically. As pointed out by [Ramanujam and Simon, 2008; Eijck, 2013], most extensions of ATL treat strategies as abstract objects rather than considering the internal structure of strategies. They proposed to use regular expressions to represent structured strategies. Existing strategic logics are based on various temporal logics such as LTL [Pnueli, 1977] and CTL [Clarke and Emerson, 1981]. [De Giacomo and Vardi, 2013] proposed linear dynamic logic LDL, an extension of LTL with regular expressions. Thus it is valuable to explore strategic logics based on LDL so that structured strategies can be represented.

Moreover, there have been works on specifying strategic abilities under constraints of rationality, by using various solution concepts in game theory. [Huang and Ruan, 2017]

*Corresponding Author

integrated correlated equilibrium (CE) into ATL, taking CE as agents' joint strategy and optimizing over the utilitarian value. [Gutierrez *et al.*, 2014] proposed a logic containing operator [NE]ψ, meaning ψ holds on all Nash equilibrium computations. [Gutierrez *et al.*, 2017] investigated techniques for the characterization and verification of equilibria in multi-play games with goals expressed using logics based on LDL.

In this paper, based on the idea of [Ghaderi *et al.*, 2007], we propose JAADL (meaning Alternating-time Dynamic Logic with Joint Abilities), a modal logic for joint abilities under strategy commitments, which is an extension of ATL*. Firstly, we introduce an operator $(A)^{\infty}_{\psi}\varphi$, meaning φ holds after iterated elimination of (strictly) dominated strategies w.r.t. group A and goal ψ. Then coalition A has joint ability to achieve ψ can be represented as the formula $(A)^{\infty}_{\psi}\langle\!\langle\varnothing\rangle\!\rangle\psi$, where $\varnothing$ is the empty set, which means that after iterated elimination of (strictly) dominated strategies, any remaining joint strategy achieves ψ. Secondly, our logic is based on LDL, so that we can use regular expressions to represent commitments to structured strategies such as traffic rules, and this can really help eliminate lack of coordination and achieve joint abilities in many cases. For example, the commitment that Car 1 cannot drive in Lane 4 is represented by the regular expression $(\neg move_1(4))^*$. This is similar to Xiong and Liu's work where strategy commitments are described by Golog programs. We present the syntax and semantics of JAADL, analyze valid formulas, give four sufficient/necessary conditions for joint abilities, and apply these conditions to analyze some interesting examples. Finally, we show that model checking memoryless JAADL is in EXPTIME.

2 Preliminaries

In this section, we introduce the concepts of concurrent game structures and strategies.

Let AP be a finite non-empty set of atoms, AC a finite non-empty set of actions, and let $AG = \{1, \dots, n\}$ be a finite non-empty set of agents. We use $\varnothing$ to denote the empty set.

Definition 1 (Concurrent Game Structures)**.** A concurrent game structure (CGS) is a tuple $\mathcal{G} = \langle W, L, P, \tau, w^0\rangle$, where

- W is a finite non-empty set of states; $w^0 \in W$ is a designated initial state; L is a labeling function mapping each state to a subset of AP; τ is a transition function mapping a state w and a decision at w to a new state;
- for each agent i, P_i is a possible action function mapping each state to a subset of AC; a decision at state w is a function mapping each agent i to an action from $P_i(w)$; we use $D(w)$ to denote the set of decisions at w;

Example 1 (Three-player Collaborative PRS (Paper, Rock, and Scissors))**.** As shown in Figure 1, three players simultaneously play the actions of Paper, Rock, and Scissors. The group wins if Player a is not beaten by Player b and Player b is not beaten by Player c. We formalize the game as a CGS $\mathcal{G}$ as follows:

- $AG = \{a, b, c\}$, $AP = \{win\}$, $AC = \{R, P, S\}$;
- $W = \{s_i, s_w, s_l\}$, $w^0 = s_i$;
- $L(s_i) = L(s_l) = \varnothing$, $L(s_w) = \{win\}$;

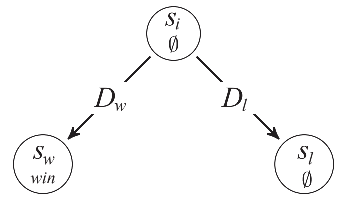

Figure 1: Three-player Collaborative PRS

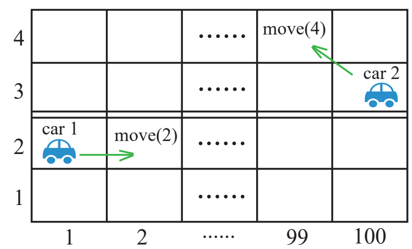

Figure 2: The Traffic Rule Example

- $P_a(s_i) = \{R, S\}$, $P_b(s_i) = \{P, S\}$, $P_c(s_i) = \{R, P\}$;
- $\tau(s_i, d) = s_w$ for $d \in D_w = \{RSP, SPR, SPP, SSP\}$, $\tau(s_i, d) = s_l$ for $d \in D_l = \{RPR, RPP, RSR, SSR\}$.

Example 2 (The Traffic Rule Example)**.** As shown in Figure 2, there are 4 lanes, and two cars that drive in opposite directions. We model this example as a CGS where

- $AG = \{1, 2\}$, $AP = \{crash\}$, $AC = \{move(k) \mid k \in \{1, 2, 3, 4\}\}$, where $move(k)$ means moving forward one unit while switching to Lane k.
- $W = \{(x_1, y_1, x_2, y_2) \mid 1 \le x_1, x_2 \le 100, 1 \le y_1, y_2 \le 4\}$, where (x_i, y_i) represents the position of car i.
- $w^0 = (1, 2, 100, 3)$.
- For any $w = (x_1, y_1, x_2, y_2) \in W$, $crash \in L(w)$ iff $(x_1, y_1) = (x_2, y_2)$.
- For any $i = 1, 2$, $w \in W$, if $crash \notin L(w)$, $P_i(w) = AC$, otherwise, $P_i(w) = \varnothing$.
- τ can be easily given. For example, $\tau((5, 3, 2, 4), (move(4), move(2))) = (6, 4, 1, 2)$.

We now define the concepts of tracks and paths. Tracks are finite state-decision sequences, they are used to define strategies: a strategy is a mapping from tracks to actions. Paths are infinite state-decision sequences, they are used to interpret JAADL path formulas.

Definition 2. A track h in a CGS $\mathcal{G}$ is a finite state-decision sequence $w_0 d_0 w_1 d_1 ... w_k$ s.t. for all i $(0 \le i < k)$, $d_i \in D(w_i)$, and $w_{i+1} = \tau(w_i, d_i)$. We use $last(h)$ to denote w_k.

Definition 3. A path λ in a CGS $\mathcal{G}$ is an infinite state-decision sequence $w_0 d_0 w_1 d_1 ...$ s.t. for all $i \ge 0$, $d_i \in D(w_i)$, and $w_{i+1} = \tau(w_i, d_i)$.

Definition 4. A strategy for agent i starting from state w is a function mapping each track h beginning from w to an action from $P_i(last(h))$. We let $Str_i(w)$ denote the set of all strategies for agent i starting from w.

Since we will handle elimination of strategies, we introduce the concept of strategy spaces. A strategy space specifies the set of possible strategies for each agent.

Definition 5. A strategy space s starting from state w is a function mapping each agent i to a subset of $Str_i(w)$. The full strategy space $fs(w)$ starting from state w maps each agent i to $Str_i(w)$.

Definition 6. A memoryless strategy for agent i is a function mapping each state w to an action from $P_i(w)$. The full memoryless strategy space, denoted by fms, maps each agent i to the set of all memoryless strategies for i.

We use σ to range over strategies. A group strategy of $A \subseteq AG$ is a mapping from A to strategies. We use σ_A to range over group strategies of A. As a special case, we use σ_i to range over strategies for agent i. A joint strategy is a group strategy of AG. We use σ_{all} to range over joint strategies. For $i \in AG$, we use $-i$ to denote $AG - \{i\}$.

Definition 7. A state w and a joint strategy σ_{all} determine a unique path $w_0 d_0 w_1 d_1 w_2 d_2 \ldots$ as follows: $w_0 = w^0$, and for each $j \geq 0$, d_j is the decision associated to the track $w_0 \ldots w_j$, *i.e.*, for each agent i, $d_j(i) = \sigma_i(w_0 \ldots w_j)$, and $w_{j+1} = \tau(w_j, d_j)$. We use $out(w, \sigma_{all})$ to denote this path.

3 Syntax and Semantics of JAADL

In this section, we introduce the syntax and semantics of JAADL. The logic is an extension of ATL* in that it is based on LDL and introduces operators of elimination of dominated strategies (EDS).

We begin with the syntax of JAADL. We use φ to denote state formulas, ψ path formulas, ϕ propositional formulas, and ρ path expressions, which are regular expressions over propositional formulas and tests of path formulas. Other than atomic propositions from AP, we introduce atomic propositions of the form a_i where $a \in$ AC and $i \in AG$, meaning agent i does action a. We use $\top$ to denote $true$.

Definition 8. JAADL formulas are built as follows:

$$\varphi ::= p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \langle\!\langle A \rangle\!\rangle \psi \mid (A)_\psi \varphi \mid (A)^\infty_\psi \varphi$$

$$\psi ::= \varphi \mid \neg\psi \mid \psi_1 \wedge \psi_2 \mid \langle \rho \rangle \psi$$

$$\rho ::= \phi \mid \psi? \mid \rho_1 + \rho_2 \mid \rho_1 ; \rho_2 \mid \rho^*$$

$$\phi ::= p \mid a_i \mid \neg\phi \mid \phi_1 \wedge \phi_2$$

where $p \in AP$, and $A \subseteq AG$.

Intuitively, $\langle \rho \rangle \psi$ means that from the current state in the path there exists an execution satisfying the path expression ρ such that its last state satisfies ψ. We use $[\rho]\psi$ as abbreviation for $\neg\langle\rho\rangle\neg\psi$. It is easy to encode LTL into path formulas as follows: $X\varphi$ translates to $\langle\top\rangle\varphi$; $G\varphi$ translates to $[\top^*]\varphi$; $\varphi_1 U \varphi_2$ translates to $\langle(\varphi_1?;\top)^*\rangle\varphi_2$.

Intuitively, $\langle\!\langle A \rangle\!\rangle\psi$ means group A has a strategy to achieve ψ. A special case of $\langle\!\langle A \rangle\!\rangle\psi$ is $\langle\!\langle \varnothing \rangle\!\rangle\psi$, which means ψ always holds no matter how the agents play. We usually write $\langle\!\langle i_1, \ldots, i_k \rangle\!\rangle$ instead of $\langle\!\langle \{i_1, \ldots, i_k\} \rangle\!\rangle$ where $i_1, \ldots, i_k \in AG$. $(A)_\psi\varphi$ means φ holds after one step of elimination of dominated strategies w.r.t. group A and the goal ψ. $(A)^\infty_\psi\varphi$ means φ holds after iterated elimination of dominated strategies w.r.t. group A and the goal ψ. We use $(A)^2_\psi\varphi$ to denote $(A)_\psi(A)_\psi\varphi$, and similarly for $(A)^k_\psi\varphi$, where $k \in \mathbb{N}$.

We use $((A))^k\psi$ to abbreviate for $(A)^k_\psi\langle\!\langle\varnothing\rangle\!\rangle\psi$. When $k = 1$, we simply write $((A))\psi$. Intuitively, $((A))^k\psi$ means after k-round elimination of dominated strategies, ψ holds no matter how the agents play, and we say group A has stage k joint ability to achieve ψ. We use $((A))^\infty\psi$ to abbreviate for $(A)^\infty_\psi(\langle\!\langle A \rangle\!\rangle\psi \wedge \langle\!\langle\varnothing\rangle\!\rangle\psi)$. The reason we conjoin $\langle\!\langle A \rangle\!\rangle\psi$ to $\langle\!\langle\varnothing\rangle\!\rangle\psi$ is that as we will show at the end of this section, the strategy space might become empty after iterated elimination of dominated strategies. We omitted the $\langle\!\langle A \rangle\!\rangle\psi$ part in the introduction to avoid confusion. Intuitively, $((A))^\infty\psi$ means group A has joint ability to ensure ψ.

Example 2 cont'd. We are concerned about whether the two cars have joint ability to ensure no crash will ever happen (resp. under the commitment that Car 1 cannot drive in Lane 4). This can be represented by the formula $((1,2))^\infty[\top^*]\neg crash$ (resp. $((1,2))^\infty[(\neg move_1(4))^*]\neg crash$).

We now provide the semantics of JAADL. We begin with the semantics of propositional formulas, which are interpreted over state-decision pairs.

Definition 9. Given a CGS $\mathcal{G}$, a state w, and a decision d at w, we interpret propositional formulas inductively:

- $w, d \models p$ if $p \in L(w)$;
- $w, d \models a_i$ if $d(i) = a$;
- $w, d \models \neg\phi$ if $w, d \not\models \phi$;
- $w, d \models \phi_1 \wedge \phi_2$ if $w, d \models \phi_1$ and $w, d \models \phi_2$.

We interpret state formulas and path formulas inductively. The interpretation is wrt a strategy space. When interpreting state formulas, we make use of two operators on strategy spaces: $R_{A,\psi,w}(s)$ and $R^\infty_{A,\psi,w}(s)$. Intuitively, $R_{A,\psi,w}(s)$ means the reduction of s via elimination of dominated strategies, and $R^\infty_{A,\psi,w}(s)$ means the reduction of s via iterated elimination of dominated strategies.

Definition 10 (JAADL Semantics)**.** Given a CGS $\mathcal{G}$, a state w, a strategy space s, and a path λ, we interpret state formulas and path formulas (we omit the cases of $\neg$ and $\wedge$) and define the operators $R_{A,\psi,w}(s)$ and $R^\infty_{A,\psi,w}(s)$ inductively:

- $w, s \models p$ if $p \in L(w)$.
- $w, s \models \langle\!\langle A \rangle\!\rangle\psi$ if there exists a group strategy $\sigma_A \in s_A$ such that for all strategies $\sigma_{-A} \in s_{-A}$, we have $out(w, (\sigma_A, \sigma_{-A})), s \models \psi$.
- $w, s \models (A)_\psi\varphi$ if $w, R_{A,\psi,w}(s) \models \varphi$.
- $w, s \models (A)^\infty_\psi\varphi$ if $w, R^\infty_{A,\psi,w}(s) \models \varphi$.
- $\lambda, s \models \varphi$ if $w_0, s \models \varphi$, where $\lambda = w_0 d_0 w_1 \ldots$.
- $\lambda, s \models \langle\phi\rangle\psi$ if $w_0, d_0 \models \phi$ and $\lambda', s \models \psi$, where $\lambda = w_0 d_0 w_1 \ldots$ and $\lambda' = w_1 d_1 \ldots$.
- $\lambda, s \models \langle\psi_1?\rangle\psi_2$ if $\lambda, s \models \psi_1$ and $\lambda, s \models \psi_2$.
- $\lambda, s \models \langle\rho_1 + \rho_2\rangle\psi$ if $\lambda, s \models \langle\rho_1\rangle\psi$ or $\lambda, s \models \langle\rho_2\rangle\psi$.
- $\lambda, s \models \langle\rho_1;\rho_2\rangle\psi$ if $\lambda, s \models \langle\rho_1\rangle\langle\rho_2\rangle\psi$.

STAGE 1		
agent i	σ_i	$M(\sigma_i)$
a	~~R~~	~~SP~~
	S	PR,PP,SP
b	P	SP,SR
	S	RP,SP
c	~~R~~	~~SP~~
	P	SP,RS,SS

Figure 3: Iterative EDS for coalition $\{a, b, c\}$

STAGE 1		
agent i	σ_i	$M(\sigma_i)$
a	~~R~~	~~SP~~
	S	PR,PP,SP
b	P	SR,SP
	S	RP,SP
c	P	
	R	

STAGE 2		
agent i	σ_i	$M(\sigma_i)$
a	S	PR,PP,SP
b	P	SR,SP
	~~S~~	~~SP~~
c	P	
	R	

Figure 4: Iterative EDS for coalition $\{a, b\}$

- $\lambda, s \models \langle\rho^0\rangle\psi$ if $\lambda, s \models \psi$.
- $\lambda, s \models \langle\rho^{k+1}\rangle\psi$ if $\lambda, s \models \langle\rho^k; \rho\rangle\psi$ for $k \in \mathbb{N}$.
- $\lambda, s \models \langle\rho^*\rangle\psi$ if there exists $k \in \mathbb{N}$ such that $\lambda, s \models \langle\rho^k\rangle\psi$.

For $\sigma_i \in s_i$, we define the set of strategies of $-i$ that work with σ_i to ensure ψ wrt state w and strategy space s as follows: $M_{\psi,w,s}(\sigma_i) = \{\sigma_{-i} \in s_{-i} \mid out(w, (\sigma_i, \sigma_{-i})), s \models \psi\}$. Sometimes when ψ, w and s are clear from the context, we omit them and write $M(\sigma_i)$. For $\sigma_i, \sigma_i' \in s_i$, we write $\sigma_i \geq_{\psi,w,s} \sigma_i'$ if $M_{\psi,w,s}(\sigma_i) \supseteq M_{\psi,w,s}(\sigma_i')$; we write $\sigma_i >_{\psi,w,s} \sigma_i'$ if $M_{\psi,w,s}(\sigma_i) \supset M_{\psi,w,s}(\sigma_i')$, and we say σ_i dominates σ_i'.

For a strategy space s, we define the reduction of s wrt group A, goal ψ and state w as follows: $R_{A,\psi,w}(s) = s'$ s.t. if $i \notin A$, $s_i' = s_i$; otherwise, $s_i' = \{\sigma_i \in s_i \mid \neg\exists\sigma_i' \in s_i. \sigma_i' >_{\psi,w,s} \sigma_i\}$. For $k \geq 2$, we define $R^k_{A,\psi,w}(s) = R_{A,\psi,w}(R^{k-1}_{A,\psi,w}(s))$. Finally, we define the iterative reduction of s: $R^\infty_{A,\psi,w}(s) = s'$ s.t. for $i \in AG$, $s_i' = \bigcap_{k=0}^{\infty} R^k_{A,\psi,w}(s)_i$.

Definition 11. We say a state formula φ is valid if for all CGS $\mathcal{G}$, we have $\mathcal{G} \models \varphi$, meaning $w^0, fs(w^0) \models \varphi$, where w^0 is the initial state of $\mathcal{G}$.

Recall $fs(w)$ is the full strategy space starting from state w.

Note that when we write $(\!(A)\!)^k\psi$, it means that the formula holds from a third-person point of view. Thus the third-person imitates the process of elimination of dominated strategies for each agent in A.

Example 1 cont'd. We have $\mathcal{G} \models \langle\!\langle a, b\rangle\!\rangle\langle\top\rangle win$, since agents a and b have a group strategy SP to achieve win no matter c plays R or P.

The procedures of iterative EDS for coalitions $\{a, b, c\}$ and $\{a, b\}$ are shown in Figures 3 and 4, respectively.

During the procedures, for each agent i in AG, for each available strategies σ_i of i, we list all the group strategies in $M(\sigma_i)$, which is just the set of strategies of $-i$ that work with σ_i to ensure the goal $\langle\top\rangle win$ w.r.t. the initial state and the full strategy space. Then we eliminate those strategies σ_i of agent i, whose $M(\sigma_i)$ is a strict subset of some $M(\sigma_j)$. Thus

$\mathcal{G} \not\models \langle\!\langle\varnothing\rangle\!\rangle\langle\top\rangle win$, $\quad \mathcal{G} \models (\!(a, b, c)\!)\langle\top\rangle win$,
$\mathcal{G} \not\models (\!(a, b)\!)\langle\top\rangle win$, $\quad \mathcal{G} \models (\!(a, b)\!)^2\langle\top\rangle win$.

We now introduce some terminology about strategies, which will be used in analyzing properties of our logic.

Definition 12. We say σ is a winning strategy for i wrt ψ, w, s if $M_{\psi,w,s}(\sigma)$ is s_{-i}. We say σ is a punishment strategy for i wrt ψ, w, s if $M_{\psi,w,s}(\sigma) = \varnothing$. We say σ is an optimal strategy for i wrt ψ, w, s, if for any $\sigma' \in s_i$, $\sigma \geq_{\psi,w,s} \sigma'$.

$$\begin{bmatrix} 1 & 1 & & & \\ & 1 & 1 & & \\ & & 1 & 1 & \\ & & & \ddots & \ddots \end{bmatrix}$$

Figure 5: Each strategy will eventually be eliminated

Thus σ is a winning strategy for i if σ works with any strategy of $-i$. The formula $\langle\!\langle i\rangle\!\rangle\psi$ represents that i has a winning strategy wrt goal ψ. Similarly, σ is a punishment strategy for i if σ works with no strategy of $-i$. The formula $\langle\!\langle i\rangle\!\rangle\neg\psi$ expresses that i has a punishment strategy wrt goal ψ. An optimal strategy for i is one that weakly dominates all others. Unfortunately, we cannot represent in JAADL that i has an optimal strategy wrt goal ψ.

Definition 13. We say that two strategies σ and σ' are equivalent wrt ψ, w, s if $M_{\psi,w,s}(\sigma) = M_{\psi,w,s}(\sigma')$. We say that two strategies σ and σ' are incomparable wrt ψ, w, s if $M_{\psi,w,s}(\sigma) \not\subseteq M_{\psi,w,s}(\sigma')$ and $M_{\psi,w,s}(\sigma') \not\subseteq M_{\psi,w,s}(\sigma)$.

Definition 14. Given a CGS $\mathcal{G}$, a state w, a strategy space s, and a goal ψ, the payoff matrix for ψ, denoted C_ψ is a 0-1 matrix defined as follows: for each $\sigma_{all} \in s$, $C_\psi(\sigma_{all}) = 1$ iff $out(w, \sigma_{all}), s \models \psi$.

Intuitively, in the 0-1 matrix, the payoff for a joint strategy is 1 if it achieves the goal, and 0 otherwise.

The definition of payoff matrices clarifies that our definition of elimination of dominated strategies is an instance of the one from game theory [Osborne and Rubinstein, 1999].

Note that different from game theory, when the strategy space is infinite, it might become empty after iterative elimination of dominated strategies. We illustrate with an example.

Example 3. There are two agents 1 and 2. Each agent i has infinitely many strategies $\sigma_i^1, \sigma_i^2, \ldots$. Figure 5 shows the payoff matrix for the goal ψ: The infinitely many rows and columns show agents 1 and 2's strategies, respectively; for each row j, there are only 1s in columns j and $j+1$. This means: for each σ_1^j, the set of strategies of 2 that work with σ_1^j to achieve the goal ψ is $\{\sigma_2^j, \sigma_2^{j+1}\}$. At stage 1, Column 1 is eliminated since it is dominated by Column 2; then at stage 2, Row 1 is eliminated since it is dominated by Row 2. Then Column 2 and Row 2 will be eliminated. Eventually, each strategy will be eliminated. By our definitions, neither $(\!(1, 2)\!)^k\psi$ where $k \in \mathbb{N}$ nor $(\!(1, 2)\!)^\infty\psi$ holds.

4 Properties of JAADL

In this section, we analyze valid formulas in JAADL, and give sufficient/necessary conditions for joint abilities, which we use to analyze some interesting examples.

First of all, it is easy to prove the following result. Item 1 says the elimination operators satisfy the K axiom. Item 2 says the negation operator can move inside elimination operators. Item 3 says the absence of stage k joint ability is equivalent to the existence of a joint strategy to achieve the negation of the goal after k round elimination of dominated strategies. Item 4 says stage k joint ability implies stage j joint ability, where $j \geq k$ or $j = \infty$. Item 5 says stage k joint ability implies the existence of a group strategy to achieve the goal. Item 6 says a coalition does not have joint ability if the other agents have a strategy to ensure the negation of the goal.

Proposition 1. *The following formulas are valid:*

1. $(A)^k_\psi(\varphi_1 \to \varphi_2) \to ((A)^k_\psi\varphi_1 \to (A)^k_\psi\varphi_2)$, $k \in \mathbb{N} \cup \{\infty\}$.
2. $\neg(A)^k_\psi\varphi \equiv (A)^k_\psi\neg\varphi$, $k \in \mathbb{N} \cup \{\infty\}$.
3. $\neg((A))^k\psi \equiv (A)^k_\psi\langle\!\langle AG\rangle\!\rangle\neg\psi$, $k \in \mathbb{N}$.
4. $((A))^k\psi \to ((A))^j\psi$, *where* $j \geq k$ *or* $j = \infty$.
5. $((A))^k\psi \to \langle\!\langle A\rangle\!\rangle\psi$, $k \in \mathbb{N} \cup \{\infty\}$.
6. $\langle\!\langle \overline{A}\rangle\!\rangle\neg\psi \to \neg((A))^\infty\psi$, *where* $\overline{A} = AG - A$.

Proof. Let $\mathcal{G}$ be a CGS, w its initial state, and $s = fs(w)$.

1. Suppose $w, s \vDash (A)^k_\psi(\varphi_1 \to \varphi_2)$ and $w, s \vDash (A)^k_\psi\varphi_1$. Then $w, R^k_{A,\psi,w}(s) \vDash \varphi_1 \to \varphi_2$ and $w, R^k_{A,\psi,w}(s) \vDash \varphi_1$. Thus $w, R^k_{A,\psi,w}(s) \vDash \varphi_2$, so $w, s \vDash (A)^k_\psi\varphi_2$.
2. $w, s \vDash \neg(A)^k_\psi\varphi$ iff $w, s \nvDash (A)^k_\psi\varphi$ iff $w, R^k_{A,\psi,w}(s) \nvDash \varphi$ iff $w, R^k_{A,\psi,w}(s) \vDash \neg\varphi$ iff $w, s \vDash (A)^k_\psi\neg\varphi$.
3. $w, s \vDash \neg((A))^k\psi$ iff $w, s \nvDash (A)^k_\psi\langle\!\langle\varnothing\rangle\!\rangle\psi$ iff $w, R^k_{A,\psi,w}(s) \nvDash \langle\!\langle\varnothing\rangle\!\rangle\psi$ iff $w, R^k_{A,\psi,w}(s) \vDash \langle\!\langle AG\rangle\!\rangle\neg\psi$ iff $w, s \vDash (A)^k_\psi\langle\!\langle AG\rangle\!\rangle\neg\psi$.
4. Suppose $w, s \vDash ((A))^k\psi$, *i.e.*, $w, R^k_{A,\psi,w}(s) \vDash \psi$. Since $R^j_{A,\psi,w}(s) \subseteq R^k_{A,\psi,w}(s)$, where $j \geq k$ or $j = \infty$, we have $w, R^j_{A,\psi,w}(s) \vDash \psi$, so $w, s \vDash ((A))^j\psi$.
5. Suppose $w, s \vDash ((A))^k\psi$. Then $R^k_{A,\psi,w}(s) \neq \varnothing$ and each $\sigma_A \in R^k_{A,\psi,w}(s)_A$ ensures ψ. Thus $w, s \vDash \langle\!\langle A\rangle\!\rangle\psi$.
6. Suppose $w, s \vDash \langle\!\langle \overline{A}\rangle\!\rangle\neg\psi$. Then there exists $\sigma_{-A} \in s_{-A}$ such that for all $\sigma_A \in s_A$, (σ_A, σ_{-A}) cannot achieve ψ. Thus $w, s \nvDash ((A))^\infty\psi$.

□

The following result says that if coalition A has a strategy to achieve ψ, then after elimination of dominated strategies, no agent in A has a punishment strategy. This is because such a strategy is already eliminated.

Proposition 2. $\langle\!\langle A\rangle\!\rangle\psi \to (A)_\psi \bigwedge_{a\in A} \neg\langle\!\langle a\rangle\!\rangle\neg\psi$ *is valid.*

Proof. Let $\mathcal{G}$ be a CGS, w its initial state, and $s = fs(w)$. Suppose $w, s \vDash \neg(A)_\psi \bigwedge_{a\in A} \neg\langle\!\langle a\rangle\!\rangle\neg\psi$. By Proposition 1(2), $w, s \vDash (A)_\psi \bigvee_{a\in A}\langle\!\langle a\rangle\!\rangle\neg\psi$. So $w, R_{A,\psi,w}(s) \vDash \bigvee_{a\in A}\langle\!\langle a\rangle\!\rangle\neg\psi$. Then there exists $a \in A$ such that $w, R_{A,\psi,w}(s) \vDash \langle\!\langle a\rangle\!\rangle\neg\psi$. Thus $w, s \vDash \neg\langle\!\langle A\rangle\!\rangle\psi$, otherwise each $\sigma_a \in s_a$ that ensures $\neg\psi$ will be eliminated and not be in $R_{A,\psi,w}(s)$. □

The following result says that coalition A has joint ability iff after iterative EDS, some agent a in A has a winning strategy. This is because if a has a winning strategy, then every remaining strategy is a winning strategy, otherwise, it is already eliminated.

Proposition 3. $((A))^\infty\psi \equiv \bigvee_{a\in A}(A)^\infty_\psi\langle\!\langle a\rangle\!\rangle\psi$ *is valid.*

Proof. Let $\mathcal{G}$ be a CGS, w its initial state, and $s = fs(w)$. Suppose $w, s \vDash ((A))^\infty\psi$. Then $w, R^\infty_{A,\psi,w}(s) \vDash \langle\!\langle A\rangle\!\rangle\psi \wedge \langle\!\langle\varnothing\rangle\!\rangle\psi$. Then for all $a \in A$, $w, R^\infty_{A,\psi,w}(s) \vDash \langle\!\langle a\rangle\!\rangle\psi$. So $w, s \vDash (A)^\infty_\psi\langle\!\langle a\rangle\!\rangle\psi$. Now suppose $w, s \vDash (A)^\infty_\psi\langle\!\langle a\rangle\!\rangle\psi$ for some $a \in A$. Then some $\sigma_a \in R^\infty_{A,\psi,w}(s)_a$ is a winning strategy wrt ψ. Thus any strategy in $R^\infty_{A,\psi,w}(s)_a$ is a winning strategy. So $w, R^\infty_{A,\psi,w}(s) \vDash \langle\!\langle A\rangle\!\rangle\psi \wedge \langle\!\langle\varnothing\rangle\!\rangle\psi$. Hence $w, s \vDash ((A))^\infty\psi$. □

Note that if $\psi_1 \to \psi_2$ is valid, $((A))^k\psi_1 \to ((A))^k\psi_2$, where $k \in \mathbb{N} \cup \{\infty\}$, may not be valid. In general, the processes of iterated elimination for two different goals might be different, even if there is a logical relationship between them. For example, there are two agents, and each agent has two strategies. The following shows the payoff matrices for the strategies wrt goals ψ_1 and ψ_2. Since $\psi_1 \to \psi_2$ is valid, when we have a 1 in C_{ψ_1}, we must have a 1 in the same position in C_{ψ_2}. It is clear that $((A))^\infty\psi_1$ holds, but $((A))^\infty\psi_2$ does not hold.

$$C_{\psi_1} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \quad C_{\psi_2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

It is easy to prove the following

Proposition 4. *When* $A \subseteq B$, $((A))\psi \to ((B))\psi$ *is valid.*

Proof. When $A \subseteq B$, for each agent a, $R_{B,\psi,w}(s)_a \subseteq R_{A,\psi,w}(s)_a$. Let $\mathcal{G}$ be a CGS, w its initial state, and $s = fs(w)$. Suppose $w, s \vDash ((A))\psi$, *i.e.*, $(A)_\psi\langle\!\langle\varnothing\rangle\!\rangle\psi$. Then $w, R_{A,\psi,w}(s) \vDash \langle\!\langle\varnothing\rangle\!\rangle\psi$. Thus $w, R_{B,\psi,w}(s) \vDash \langle\!\langle\varnothing\rangle\!\rangle\psi$. Hence $w, s \vDash ((B))\psi$. □

However, when $A \subseteq B$, $((A))^k\psi \to ((B))^k\psi$, where $k \in \mathbb{N} \cup \{\infty\}$, may not be valid.

We now give three sufficient conditions for joint abilities.

The following result shows that if at stage k, some agent in the coalition has a winning strategy, then there is joint ability at stage $k+1$.

Theorem 1. $(A)^k_\psi\langle\!\langle a\rangle\!\rangle\psi \to ((A))^{k+1}\psi$, $a \in A$, *is valid.*

Proof. Suppose a has a stage k winning strategy σ_a. Then σ_a remains at stage $k+1$. For every strategy σ'_a which remains at stage $k+1$, it must be a stage $k+1$ winning strategy, otherwise, $\sigma_a > \sigma'_a$, hence it is already eliminated. Thus there is joint ability at stage $k+1$. □

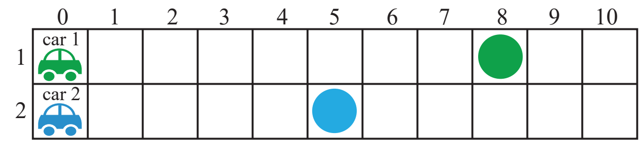

Figure 6: Autonomous Cars Example

Theorem 2. *If $\langle\langle A\rangle\rangle\psi$ holds, and at stage k, each agent in A has an optimal strategy, then there is joint ability at stage $k+1$.*

Proof. Suppose $\langle\langle A\rangle\rangle\psi$ holds, and at stage k, each agent in A has an optimal strategy. Then for each joint strategy σ_{all} where for each $a \in A$, σ_a is an optimal strategy, σ_{all} achieves ψ. At the next stage, each non-optimal strategy will be deleted. Hence, there is joint ability at stage $k+1$. □

Theorem 3. *If $\langle\langle A\rangle\rangle\psi$ holds, and at stage k, $|A|-1$ agents in A have optimal strategies, then there is joint ability at stage $k+2$.*

Proof. Suppose $\langle\langle A\rangle\rangle\psi$ holds, $a \in A$, and at stage k, each agent in $A-\{a\}$ have an optimal strategy. Then a has a strategy σ_a such that for each group strategy σ_{-a} where for each $b \in A-\{a\}$, σ_b is an optimal strategy, (σ_a, σ_{-a}) achieves the goal. At stage $k+1$, each non-optimal strategy of each agent in $A-\{a\}$ will be deleted. Thus a must have a remained strategy which is a winning strategy. By Theorem 1, there is joint ability at stage $k+2$. □

Below, we give a necessary condition for joint abilities.

Theorem 4. *Suppose at some stage, no agent in A has a winning strategy, and any two strategies are either equivalent or incomparable, then there is no joint ability.*

Proof. When any two strategies are either equivalent or incomparable, no elimination is possible. Since no agent in A has a winning strategy, there is no joint ability. □

Finally, we apply the above theorems to analyze the traffic rule example and two more interesting examples.

Example 2 cont'd. Clearly, $\mathcal{G} \not\models ((1,2))^{\infty}[\top^*]\neg crash$. This is because any two different strategies are incomparable, by Theorem 4, there is no joint ability.

However, if Car 1 cannot drive in Lane 4, then the two cars will have joint ability to ensure no crash will ever happen, *i.e.*, $\mathcal{G} \models ((1,2))[(\neg move_1(4))^*]\neg crash$. This is because Car 2 can choose to always drive in Lane 4, which is a winning strategy. By Theorem 1, there is stage 1 joint ability.

Example 4 (Autonomous Cars)**.** As shown in Figure 6, two cars move in the same direction. They need to switch lanes to avoid obstacles on the road. The two cars win if both get to the destination without crashing into each other or the obstacles. We can model it as a CGS $\mathcal{G}$ where $AP = \{win, crash\}$, and $AC = \{K, W, S\}$, where K means keeping moving, W means waiting, and S means switching lanes, *i.e.*, moving up or down.

We first consider the situation where there are no obstacles on the road. We model this as the CGS $\mathcal{G}'$. We use AK to denote the strategy of always keeping moving. Then $\mathcal{G}' \models ((1,2))^{\infty}[\top^*]\neg crash$. This is because AK won't make a crash. Thus AK is a winning strategy. By Theorem 1, there is stage 1 joint ability.

We also have $\mathcal{G}' \models ((1,2))\langle\top^*\rangle win$. We prove that for each car, AK is an optimal strategy: Let σ' be a strategy of the other car. Suppose that (σ, σ') achieves the goal, then there is no crash, and the other car gets to the destination. Since AK won't make a crash, (AK, σ') achieves the goal. By Theorem 2, there is stage 1 joint ability.

Now we consider the case with obstacles. We have $\mathcal{G} \models ((1,2))^2\langle\top^*\rangle win$. This is because Car 1 has an optimal strategy: keep going, switch lane at position 7, and keep going. By Theorem 3, there is stage 2 joint ability.

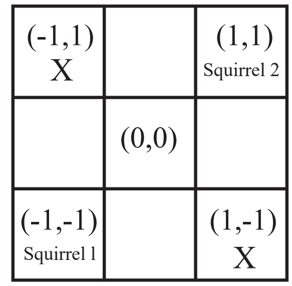

Figure 7: The Squirrels World

Example 5 (The Squirrels World)**.** There are two squirrels and two acorns living in a finite grid. Each squirrel can do actions below: $pick$ up an acorn if she is located at the same cell as this acorn and does not hold any acorn; and move up, $down$, $right$, and $left$ a cell. As shown in Figure 7, initially, squirrel 1 is located at the cell $(-1,-1)$ and 2 is located at the cell $(1,1)$; there is only one acorn in each of the cell $(-1,1)$ and $(1,-1)$, and in other cells there are no acorns.

We can model this squirrels world as a CGS $\mathcal{G}$, and we omit the details here. We use two atoms $hold_1$ and $hold_2$, where $hold_i$ means squirrel i holds an acorn.

We first consider the formula $\psi = \langle\top;\top;\top\rangle(hold_1 \wedge hold_2)$, which means each squirrel will hold an acorn in exactly 3 steps. Clearly, $\mathcal{G} \models \langle\langle 1,2\rangle\rangle\psi$. This is because the two squirrels have a strategy to achieve ψ, for example, Squirrel 1 does the action sequence $right; right; pick$ and Squirrel 2 does $left; left; pick$. However, $\mathcal{G} \not\models ((1,2))^{\infty}\psi_3$. It is easy to show that at stage 1, any two different strategies are incomparable. By Theorem 4, there is no joint ability.

If we require the first actions of the two squirrels to be $(right, left)$ or $(up, down)$, there is joint ability, *i.e.*, $\mathcal{G} \models ((1,2))\langle right_1 \wedge left_2; \top; \top\rangle(hold_1 \wedge hold_2)$ and $\mathcal{G} \models ((1,2))\langle up_1 \wedge down_2; \top; \top\rangle(hold_1 \wedge hold_2)$. This is because in each case, each agent has an optimal strategy. By Theorem 2, there is stage 1 joint ability.

5 Model-Checking Memoryless JAADL

In this section, we explore the computational complexity of model checking JAADL where we consider only memoryless strategies, which are functions from states to actions.

Algorithm 1 Labeling State Space

$Label(\mathcal{G}, s, \varphi)$
for φ' in $Sub(\varphi)$ **do**
  case $\varphi' = p$: $[\varphi']_s \leftarrow \{w \in W \mid p \in L(w)\}$
  case $\varphi' = \neg\varphi$: $[\varphi']_s \leftarrow W - [\varphi]_s$
  case $\varphi' = \varphi_1 \wedge \varphi_2$: $[\varphi']_s \leftarrow [\varphi_1]_s \cap [\varphi_2]_s$
  case $\varphi' = \langle\langle A\rangle\rangle\psi$:
    $[\varphi']_s \leftarrow \{w \mid \exists\sigma_A \in s_A \forall\sigma_{-A} \in s_{-A}.$ $PathF(\mathcal{G}, w, (\sigma_A, \sigma_{-A}), s, \psi) = true\}$
  case $\varphi' = (A)_\psi\varphi_1$: $[\varphi']_s \leftarrow [\varphi_1]_{StrS(A,\psi,w,s)}$
  case $\varphi' = (A)^\infty_\psi\varphi_1$: $[\varphi']_s \leftarrow [\varphi_1]_{StrS^\infty(A,\psi,w,s)}$
end for
return $[\varphi]_s$

Algorithm 2 Model-Checking Path Formulas

function $PathF(\mathcal{G}, w, \sigma_{all}, s, \psi)$
$Max(\psi) \leftarrow$ the set of maximal state-subformulas in ψ
for each $\varphi \in Max(\psi)$ **do**
  define a fresh atom p_φ,
  let $w \in L(p_\varphi)$ iff $w \in Label(\mathcal{G}, s, \varphi)$
end for
replace each occurrence in ψ of $\varphi \in Max(\psi)$ by p_φ to get a pure LDL formula ψ_{ldl}
return whether $Kripke(w, \sigma_{all}) \vDash_{ldl} \psi_{ldl}$

We first formally state the model checking problem for memoryless JAADL: Given a CGS $\mathcal{G}$, and a JAADL formula φ, decide if $w^0, \mathit{fms} \vDash \varphi$, where w^0 is the initial state of $\mathcal{G}$. Recall *fms* denotes the full memoryless strategy space of $\mathcal{G}$.

Now we give a labeling algorithm (Algorithm 1) which, given a CGS $\mathcal{G} = \langle W, L, P, \tau, w^0\rangle$, a strategy space s, and a state formula φ, returns $[\varphi]_s$, denoting the set of all states satisfying φ, *i.e.*, $[\varphi]_s = \{w \in W \mid w, s \vDash \varphi\}$. Note that different from ATL, JAADL formulas are evaluated w.r.t. a strategy space, thus the notation $[\varphi]_s$ has s as the subscript.

Our algorithm proceeds by cases. For case $\langle\langle A\rangle\rangle\psi$, we make use of function $PathF$ shown in Algorithm 2, which is used to decide whether $out(w, \sigma_{all}), s \vDash \psi$. A joint strategy σ_{all} and a state w in a CGS $\mathcal{G}$ uniquely determine an infinite path $out(w, \sigma_{all})$, which can be viewed as a finite Kripke model with initial state w, we denote it as $Kripke(w, \sigma_{all})$. Given a pure LDL formula ψ_{ldl}, we can call the LDL model-checking algorithm [Faymonville and Zimmermann, 2017] with minor change due to our action atoms, to verify whether $Kripke(w, \sigma_{all}) \vDash_{ldl} \psi_{ldl}$. Their algorithm is polynomial in the model size and exponential in the formula size.

For case $(A)_\psi\varphi_1$, we call function $StrS$ given in Algorithm 3, which is used to calculate $R_{A,\psi,w}(s)$. For case $\varphi' = (A)^\infty_\psi\varphi_1$, $R^\infty_{A,\psi,w}(s)$ can be computed as follows: repeat $s \leftarrow StrS(A, \psi, w, s)$ until there is no change to s.

Theorem 5. *Model-checking memoryless JAADL can be done in time exponential in the model size and formula size.*

Proof. Let n be the model size and l the formula size. Then the number of different memoryless strategies is $O(2^n)$. Calculating the reduction and iterative reduction of a strategy

Algorithm 3 Calculating Reduced Strategy Space

function $StrS(A, \psi, w, s)$
for each $i \in A$ **do**
  for each $\sigma_i \in s_i$, compute $M_{\psi,w,s}(\sigma_i)$, *i.e.*, $\{\sigma_{-i} \in s_{-i} \mid PathF(\mathcal{G}, w, (\sigma_i, \sigma_{-i}), s, \psi) = true\}$
  for each $\sigma_i, \sigma'_i \in s_i$, if $M_{\psi,w,s}(\sigma_i) \supset M_{\psi,w,s}(\sigma'_i)$ then $s_i \leftarrow s_i - \{\sigma'_i\}$
end for
return s

space takes time $O(2^n 2^l)$, and so does processing the case $\langle\langle A\rangle\rangle\psi$. Thus the whole algorithm takes time $O(2^n 2^l)$. □

The exponential complexity result is due to the fact that the concept of joint abilities is based on elimination of dominated strategies. Model checking strategic logics beyond ATL usually has high complexity. Even restricting to memoryless strategies does not help much. For example, model checking memoryless Strategy Logic (SL) is PSPACE-complete wrt both the model size and the formula size [Čermák *et al.*, 2018]. The paper presents a labeling algorithm for model checking memoryless SL which is exponential time wrt both the model size and the formula size, and gives a symbolic implementation of the labeling algorithm.

6 Conclusions

In this paper, we have proposed JAADL, a modal logic for joint abilities under strategy commitments, which is an extension of ATL*. Firstly, we introduce an explicit operator for elimination of dominated strategies so that joint abilities can be expressed. Secondly, the logic is based on LDL, so that regular expressions can be used to represent constraints of structured strategies such as norms/social laws. We analyze valid formulas in the logic, and identify three sufficient conditions and a necessary condition for joint abilities. These conditions make use of the concepts of winning strategies, optimal strategies, equivalent strategies, and incomparable strategies. We use examples to illustrate that we can conveniently apply these conditions to analyze whether there exist joint abilities. Finally, we prove that model checking memoryless JAADL is in EXPTIME.

In the future, we are interested in a thorough investigation of the computational complexity of model-checking JAADL, including the exact complexity of the memoryless case and the general case. Also, we are interested in implementing a symbolic model checker for memoryless JAADL.

Acknowledgments

We thank the anonymous reviewers for helpful comments. We acknowledge support from the Natural Science Foundation of China under Grant Nos. 61572535 and U1811263, and the National Science and Engineering Research Council of Canada.

References

[Ågotnes *et al.*, 2007] Thomas Ågotnes, Valentin Goranko, and Wojciech Jamroga. Alternating-time temporal logics with irrevocable strategies. In *TARK*, pages 15–24, 2007.

[Alechina *et al.*, 2010] Natasha Alechina, Brian Logan, Nguyen Hoang Nga, and Abdur Rakib. Resource-bounded alternating-time temporal logic. In *AAMAS*, pages 481–488, 2010.

[Alur *et al.*, 2002] Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. Alternating-time temporal logic. *J. ACM*, 49(5):672–713, 2002.

[Chatterjee *et al.*, 2010] Krishnendu Chatterjee, Thomas A. Henzinger, and Nir Piterman. Strategy logic. *Inf. Comput.*, 208(6):677–693, 2010.

[Clarke and Emerson, 1981] Edmund M. Clarke and E. Allen Emerson. Design and Synthesis of Synchronization Skeletons Using Branching-Time Temporal Logic. In *Logics of Programs, Workshop*, volume 131 of *Lecture Notes in Computer Science*, pages 52–71. Springer, 1981.

[De Giacomo and Vardi, 2013] Giuseppe De Giacomo and Moshe Y. Vardi. Linear temporal logic and linear dynamic logic on finite traces. In *IJCAI*, 2013.

[Eijck, 2013] Jan van Eijck. PDL as a multi-agent strategy logic. In *TARK*, 2013.

[Faymonville and Zimmermann, 2017] Peter Faymonville and Martin Zimmermann. Parametric linear dynamic logic. *Inf. Comput.*, 253:237–256, 2017.

[Ghaderi *et al.*, 2007] Hojjat Ghaderi, Hector J. Levesque, and Yves Lespérance. A logical theory of coordination and joint ability. In *AAAI*, pages 421–426, 2007.

[Gutierrez *et al.*, 2014] Julian Gutierrez, Paul Harrenstein, and Michael J. Wooldridge. Reasoning about equilibria in game-like concurrent systems. In *KR*, 2014.

[Gutierrez *et al.*, 2017] Julian Gutierrez, Giuseppe Perelli, and Michael Wooldridge. Iterated games with LDL goals over finite traces. In *AAMAS*, pages 696–704, 2017.

[Hawke, 2017] Peter Hawke. The logic of joint ability in two-player tacit games. *Rew. Symb. Logic*, 10(3):481–508, 2017.

[Huang and Ruan, 2017] Xiaowei Huang and Ji Ruan. ATL strategic reasoning meets correlated equilibrium. In *IJCAI*, pages 1102–1108, 2017.

[Huang *et al.*, 2012] Xiaowei Huang, Kaile Su, and Chenyi Zhang. Probabilistic alternating-time temporal logic of incomplete information and synchronous perfect recall. In *AAAI*, 2012.

[Jamroga and van der Hoek, 2004] Wojciech Jamroga and Wiebe van der Hoek. Agents that know how to play. *Fundam. Inform.*, 63(2-3):185–219, 2004.

[Mogavero *et al.*, 2014] Fabio Mogavero, Aniello Murano, Giuseppe Perelli, and Moshe Y. Vardi. Reasoning about strategies: On the model-checking problem. *ACM Trans. Comput. Log.*, 15(4):34:1–34:47, 2014.

[Naumov and Tao, 2018] Pavel Naumov and Jia Tao. Strategic coalitions with perfect recall. In *AAAI*, pages 4702–4709, 2018.

[Osborne and Rubinstein, 1999] Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*. The MIT Press, 1999.

[Pauly, 2002] Marc Pauly. A modal logic for coalitional power in games. *J. Log. Comput.*, 12(1):149–166, 2002.

[Pnueli, 1977] Amir Pnueli. The temporal logic of programs. In *18th Annual Symposium on Foundations of Computer Science*, pages 46–57. IEEE Computer Society, 1977.

[Ramanujam and Simon, 2008] Ramaswamy Ramanujam and Sunil Easaw Simon. Dynamic logic on games with structured strategies. In *KR*, pages 49–58, 2008.

[Reiter, 2001] Raymond Reiter. *Knowledge in action: logical foundations for specifying and implementing dynamical systems*. MIT press, 2001.

[Schobbens, 2004] Pierre-Yves Schobbens. Alternating-time logic with imperfect recall. *Electr. Notes Theor. Comput. Sci.*, 85(2):82–93, 2004.

[Shoham and Tennenholtz, 1995] Yoav Shoham and Moshe Tennenholtz. On social laws for artificial agent societies: Off-line design. *Artif. Intell.*, 73(1-2):231–252, 1995.

[van der Hoek and Wooldridge, 2003] Wiebe van der Hoek and Michael J. Wooldridge. Cooperation, knowledge, and time: Alternating-time temporal epistemic logic and its applications. *Studia Logica*, 75(1):125–157, 2003.

[Xiong and Liu, 2016] Liping Xiong and Yongmei Liu. Strategy representation and reasoning for incomplete information concurrent games in the situation calculus. In *IJCAI*, pages 1322–1329, 2016.

[Čermák *et al.*, 2018] Petr Čermák, Alessio Lomuscio, Fabio Mogavero, and Aniello Murano. Practical verification of multi-agent systems against Slk specifications. *Information and Computation*, 261:588 – 614, 2018.