

ATL Strategic Reasoning Meets Correlated Equilibrium

Xiaowei Huang¹, Ji Ruan²

¹School of Computing and Digital Technology, Birmingham City University, United Kingdom

²Department of Computer Science, Auckland University of Technology, New Zealand
 {xiaowei.huang@bcu.ac.uk, jiruan@aut.ac.nz}

Abstract

This paper is motivated by analysing a Google self-driving car accident, i.e., the car hit a bus, with the framework and the tools of strategic reasoning by model checking. First of all, we find that existing ATL model checking may find a solution to the accident with *irrational* joint strategy of the bus and the car. This leads to a restriction of treating both the bus and the car as rational agents, by which their joint strategy is an equilibrium of certain solution concepts. Second, we find that a randomly-selected joint strategy from the set of equilibria may result in the collision of the two agents, i.e., the accident. Based on these, we suggest taking Correlated Equilibrium (CE) as agents' joint strategy and optimising over the utilitarian value which is the expected sum of the agents' total rewards. The language ATL is extended with two new modalities to express the existence of a CE and a unique CE, respectively. We implement the extension into a software model checker and use the tool to analyse the examples in the paper. We also study the complexity of the model checking problem.

1 Introduction

On March 2016, Google revealed that its self-driving car hit a bus because it made an incorrect assumption about where it would go [Goo, 2016; TheVerge, 2016]. A simple illustration of the accident is in Figure 1. After its pathway being blocked by sand bags (i.e., scenario (a)), the car on autonomous driving mode intends to merge into the traffic, within which there is a bus coming from the behind. There exist four possibilities that could happen next: the bus lets the car merge into the traffic first (scenario (b)), which is the one assumed by the car, the car lets the bus pass first (scenario (c)), which is the one assumed by the bus driver, both of them stay still which literally repeats the current scenario (a), and they crash into one another (scenario (d)), which is what actually happened.

The accident provides an example for the study of strategic reasoning by model checking. Model checking [Clarke *et al.*, 1999] is a collection of techniques to automatically verify the correctness of a system against its specification. If described as a multiagent system M and reasoned with existing strategic

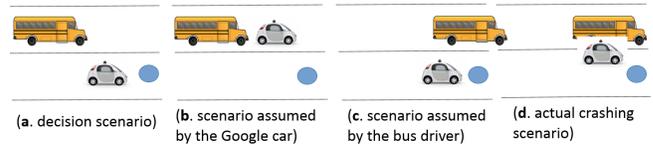


Figure 1: An analysis of the Google car accident

reasoning frameworks based on ATL model checking [Alur *et al.*, 2002], software model checkers such as MOCHA [Alur *et al.*, 1998], MCK [Huang and van der Meyden, 2014a; Huang, 2015], and MCMAS [Čermák *et al.*, 2014] will be able to claim, by certifying the satisfiability of the relation $M, s_{(a)} \models \langle\langle car, bus \rangle\rangle \Box \neg crash$ on the state $s_{(a)}$ representing the scenario (a), that there exists a joint strategy for the agents to avoid the accident. However, we notice that *one of the satisfiable strategies, in which both agents refrain from moving forward, is irrational*, so that agents have incentive to deviate from it by committing to other strategies. This observation leads to the restriction of considered strategies to rational strategies, or rational strategic reasoning [Wooldridge *et al.*, 2016]. Furthermore, we notice that *randomly selecting an equilibrium from the set of equilibria of certain solution concepts as the joint strategy may lead to the collision of the two agents*, including both Nash Equilibria (NEs) and Correlated Equilibria (CEs) [Aumann, 1974].

The main contribution of this paper is to complement the ATL strategic reasoning with new strategic modalities $CE_G^{psw} \psi$ and $UCE_G^{psw} \psi$ for the expressiveness of taking CEs as agents' joint strategies to achieve goal ψ . The semantics of the new modalities optimises the selection of equilibrium over objectives such as the utilitarian value [Greenwald and Hall, 2003]. It is known that deciding the existence of a CE is a computationally easier problem than for NE [Gilboa and Zemel, 1989], and a CE can have a higher utilitarian value than any NE (including both pure and mixed NEs). Beyond cooperative competition¹ through equilibrium, our CE-based model checking can reason about the existence of a CE for a group of agents to *compete with* the other group of agents in a zero-sum way. More specifically, a CE will consider not only the

¹In this paper, to simply notation, we may call cooperative competition as the collaboration. This is to distinguish with the pure competition in the zero-sum games.

rewards gained for the agents in the group but also the possible behaviour of the other agents which may disable the achievement of some rewards.

We implement a prototype model checker for CE-based strategic reasoning (which also includes usual ATL strategic reasoning), and thus all the results in our running examples have been formally verified. To the best of our knowledge, this is the first model checker which can reason about CE. Moreover, we study the computational complexity of the model checking problem.

2 Model Checking Multiagent Systems

We work with the notion of multiagent systems, which comprise of a set of agents running in an environment.

Definition 1 Let Agt be a finite set of agents, each of which has a finite set Act_i of actions for $i \in Agt$. A multiagent system is a tuple $M = (S, I, \{N_i\}_{i \in Agt}, T, L)$, where S is a finite set of (environment) states, $I \subseteq S$ is a finite set of initial states, $N_i : S \rightarrow \mathcal{P}(Act_i)$ provides agent i with a set of legal actions on every state such that $N_i(s) \subseteq Act_i$, $T : S \times Act \rightarrow \mathcal{P}(S)$ is a (partial) transition function such that $Act = \times_{i \in Agt} Act_i$, and $L : S \rightarrow \mathcal{P}(Prop)$ is a labelling function.

Let a_i be agent i 's action in the joint action $a \in Act$. Agents take their actions independently: for any state s and joint action a such that $a_i \in N_i(s)$ for all $i \in Agt$, we have $T(s, a) \neq \emptyset$. We define a finite, resp. infinite, path ρ as a sequence of states $s_0 s_1 s_2 \dots$ such that $s_{i+1} \in T(s_i, -)$ for all $i \geq 0$, and denote the set of finite and infinite paths of M starting in s , respectively, by $FPath_M(s)$ and $IPath_M(s)$, and sets of paths starting from any state by $FPath_M$ and $IPath_M$, and omit M if clear from context. For any path ρ we write $\rho(n)$ for its $(n+1)$ -th state, i.e., s_n . For a finite path ρ we write $last(\rho)$ for its last state. For an infinite path $\rho = s_0 s_1 \dots$, we write $\rho[k..\infty]$ for its suffix $s_k s_{k+1} \dots$.

For an agent i , a (memoryless and deterministic) strategy $\sigma_i : S \rightarrow Act_i$ is such that $\sigma_i(s) \in N_i(s)$ for all $s \in S$. Let Γ_i be a set of strategies for agent i . A joint (pure) strategy σ_G for a set $G \subseteq Agt$ of agents is a vector of strategies $(\sigma_i)_{i \in G}$ such that $\sigma_i \in \Gamma_i$. Let $\Gamma_G = \times_{i \in G} \Gamma_i$ be a set of joint strategies for a set G of agents. When $G = Agt$, a joint strategy σ_G is called a strategy profile. The application of a strategy σ_i on M can be represented as $M\sigma_i = (S, I, \{N_i\}_{i \in Agt \setminus \{i\}} \cup \{\sigma_i\}, T, L)$ which replace the legal action function N_i of agent i with the strategy σ_i . Such application can be generalised to a joint strategy σ_G and obtain $M\sigma_G$.

ATL [Alur *et al.*, 2002] is an expressive language to describe the specifications of a multi-agent system M .

$$\phi ::= p \mid \neg\phi \mid \phi_1 \vee \phi_2 \mid \langle\langle G \rangle\rangle \phi \mid \langle\langle G \rangle\rangle (\phi_1 \cup \phi_2) \mid \langle\langle G \rangle\rangle \Box \phi$$

where $p \in Prop$ and $G \subseteq Agt$. ATL has the expressiveness of CTL [Clarke *et al.*, 1999] as $\langle\langle Agt \rangle\rangle \phi \equiv \exists \phi$. We may also use other standard operators such as $\phi_1 \wedge \phi_2$, $\Diamond \phi$, $\forall \phi$, etc.

The semantics of ATL over the system M and a state $s \in S$ is given as the relation $M, s \models \phi$, which is defined recursively over the structure of the ATL formula ϕ . Let M_0 be the original system without the application of any strategy.

- Propositional cases are omitted.

- $M, s \models \langle\langle G \rangle\rangle \psi$ if there exists some joint strategy $\sigma_G \in \Gamma_G$ such that $M_0 \sigma_G, \rho \models \psi$ for all $\rho \in IPath_{M_0 \sigma_G}(s)$.

The relation $M, \rho \models \psi$ is defined recursively for $\rho \in IPath_M$.

- $M, \rho \models \phi$ if $M, \rho(0) \models \phi$, for ϕ a state formula
- $M, \rho \models \Box \phi$ if $M, \rho[1..\infty] \models \phi$,
- $M, \rho \models \phi_1 \cup \phi_2$ if there exists a number $m \geq 0$ such that $M, \rho[k..\infty] \models \phi_1$ for $0 \leq k \leq m-1$ and $M, \rho[m..\infty] \models \phi_2$,
- $M, \rho \models \Box \phi$ if $M, \rho[k..\infty] \models \phi$ for all $k \geq 0$.

Definition 2 Given a multiagent system M and an ATL formula ϕ , the model checking problem, written as $M \models \phi$, is to determine whether $M, s \models \phi$ for all initial states $s \in I$.

3 The Need of Rational Strategy

Consider as a running example a multiagent system in which the decision making scenario of the Google car accident occurs repeatedly.

Example 1 Two autonomous cars, moving in the same direction, repeatedly need to share a common lane as that of scenario (a) in Figure 1 due to the obstacles on the road. In our modelling, each car has three local states $\{s_a, s_d, s_o\}$, where s_a represents that it is approaching the next sharing lane, s_d represents the state on which it decides how to handle the situation (give way or occupy), and s_o represents that it is occupying the lane. If decided to give way, a car will stay in the state s_d so that the other car can proceed to occupy the lane. Every car has to occupy the shared lane once to move ahead.

Each car has two actions $\{m, w\}$ representing the behaviour of moving and waiting, respectively. The environment states are $S = \{s_a, s_d, s_o\} \times \{s_a, s_d, s_o\}$. Initially, we let $I = \{(s_a, s_a)\}$, representing that cars are approaching the first shared lane. The legal action function N_i is $N_i(s_a) = m$, $N_i(s_d) = \{m, w\}$, and $N_i(s_o) = m$. Intuitively, a car may choose to either move to occupy the lane or give way if it is in state s_d , and can only move otherwise. The transition function T is deterministic as follows: $T((s_1, s_2), (a_1, a_2)) = \{(s'_1, s'_2)\}$ is such that, for all $i \in \{1, 2\}$, if $a_i = m$ then $(s_i, s'_i) \in \{(s_a, s_d), (s_d, s_o), (s_o, s_a)\}$, and if $a_i = w$ then $s_i = s'_i$.

Let crash be an atomic proposition that will be labeled on those states on which both cars are in state s_o . The formula

$$\phi_1 \equiv \langle\langle Agt \rangle\rangle \Box \neg \text{crash}$$

expresses the existence of strategy profile σ_{Agt} to avoid a crash. Assume the following two strategies for agents²:

- $\sigma_x((s_d, s_d)) = w$ and $\sigma_x(s) = m$ for $s \neq (s_d, s_d)$, and
- $\sigma_y(s) = m$ for $s \in S$.

Let agent 1 be the bus and agent 2 be the car in the Google car accident. We can see that, the joint strategy $\sigma_{xy} = (\sigma_x, \sigma_y)$ results in scenario (b), $\sigma_{yx} = (\sigma_y, \sigma_x)$ results in scenario (c), $\sigma_{xx} = (\sigma_x, \sigma_x)$ results in an infinite loop on the decision scenario, and $\sigma_{yy} = (\sigma_y, \sigma_y)$ results in the accident. It can be concluded that the relation $M, s \models \phi_1$ for any $s \in S$ is satisfiable with any joint strategy other than σ_{yy} .

²As there are 6 states, the number of strategies is $2^6 = 64$. But to simplify the discussion, we assume that agents always move when they are not in the state (s_d, s_d) ; hence this reduces to two strategies.

Note that, every agent has a number $|\Gamma_i| = 2$ of strategies, which results in a number $|\Gamma_{\{1,2\}}| = 4$ of joint strategies.

It is reasonable to assume that in the Google car accident, both the bus and the car are rational agents. However, when reasoning with $\langle\langle G \rangle\rangle\psi$, the synthesised strategy can be irrational for the agents (as will be explained in Example 2). If so, a rational agent may refuse to adhere to the strategy, and therefore the strategic reasoning becomes useless.

Following the solution concepts studied in game theory, we define the rationality of a set of agents as the fact that they follow a joint strategy which is an equilibrium with respect to certain solution concept. In this paper, we consider two notable concepts: Nash Equilibrium (NE) [Nash, 1950] and Correlated Equilibrium (CE) [Aumann, 1974]. Intuitively, NE means that every agent knows the strategies of the other agents, and no agent has anything to gain (i.e., the incentive) by changing only his or her own strategy. CE has an intuition that each agent chooses their action according to their observation of the value of the same public signal.

It is known that CE is more general than NE in the sense that all NEs are CEs in a game. More notably, comparing with NE which is PPAD-complete [Daskalakis et al., 2009], CE has a computational advantage that it is polynomial time computable for matrix games [Gilboa and Zemel, 1989]. Moreover, CEs may achieve better performance than NEs: any convex combination of NEs is a CE, but the payoff of a CE can be outside of the convex hull of the NEs' payoffs [Aumann, 1987].

Example 2 (Continue with Example 1) The joint strategy σ_{xx} is a strategy for the relation $M, s \models \phi_1$. However, it can be verified that σ_{xx} is neither NE nor CE, and agents find that it is irrational. Intuitively, the joint strategy of both agents refraining from moving forward is undesirable, because it traps the system into an infinite self-loop, and moreover both agents want to deviate from it, because the deviation will bring obvious benefit for them with respect to either NE or CE.

A possible fix to this is to lift formula ϕ_1 with components describing the rejection of undesirable situations. However, such a fix means that the formula ϕ_1 has to be problem specific by enumerating all undesirable situations, which may not be practical. In this paper, we follow the approach of handling this problem by considering restricting the focus of strategic reasoning on those strategies that are equilibria of certain solution concepts. Each equilibrium corresponds to a collective rational decision of the agents involved.

To formally define equilibrium notions in particular CE³, we associate each agent with a set of goals and rewards.

Definition 3 Let $Goal_i$ be a set of goal-reward pairs (ϕ, w) of agent i , where ϕ is a CTL formula to express a goal that i wants to achieve and $w \geq 0$ is a real-valued number representing the reward that i will receive if ϕ is achieved.

We take CTL as the specification language to describe goals because it is sufficiently expressive for many cases and has relatively lower computational complexity on model

³For space limit, we do not present the detail of NE. CE is formally given in Definition 4.

checking problem comparing to other logics such as LTL. The framework can be easily generalised to other logics.

Example 3 (Continue with Example 1) Let $p_{i@x}$ be an atomic proposition that is labelled on those states s such that agent i is on state s_x . For example, $p_{2@a}$ means that agent 2 is on state s_a . We use formula $\phi_2 \equiv \forall \square \neg \text{crash}$ to denote that there will never be a crash, and $\phi_3(i) \equiv \forall \square (p_{i@d} \Rightarrow \exists \bigcirc p_{i@o})$ to denote that agent i can move first on the shared lane.

For both agents $i \in \{1, 2\}$, we let $Goal_i$ include the following goal-reward pairs, where $j \neq i$ denotes the other agent:

- $(\phi_2 \wedge \phi_3(i), 5)$ (i.e., selfish)
- $(\phi_2 \wedge \bigwedge_{j \neq i, j \in \text{Agt}} (\neg \phi_3(i) \Rightarrow \neg \phi_3(j)), 4)$ (i.e., competitive)
- $(\phi_2 \wedge \bigvee_{j \neq i, j \in \text{Agt}} \phi_3(j), 2)$ (i.e., selfless)
- $(\neg \phi_2, 0)$ (i.e., reckless)

Intuitively, a selfish agent can achieve a reward 5 by ensuring that it can move ahead of the other agent on shared lane. A competitive agent can achieve a reward 4 by ensuring that if it cannot move ahead of the other agent then the other agent cannot either. A selfless agent gets a reward 2 by allowing the other agent to move first. All these three goals are required to avoid crashes. Finally, a reckless agent is discouraged and can only receive a reward 0.

Let $M^\dagger = (M, \{Goal_i\}_{i \in \text{Agt}})$ be the system M enhanced with an additional structure $\{Goal_i\}_{i \in \text{Agt}}$. For simplicity, in the following we still write M for M^\dagger .

4 CE-based Strategic Logic

We extend ATL with two new operators CE and UCE to express correlated equilibrium and call the resulting language ATL_{UCE} . Formally, ATL_{UCE} has the syntax as follows.

$$\begin{aligned} \phi & ::= p \mid \neg \phi \mid \phi_1 \vee \phi_2 \mid \langle\langle G \rangle\rangle\psi \mid CE_G^{w} \psi \mid UCE_G^{w} \psi \\ \psi & ::= \bigcirc \phi \mid \phi_1 U \phi_2 \mid \square \phi \end{aligned}$$

where $p \in Prop$, $G \subseteq \text{Agt}$, $\triangleright \in \{>, \geq, \leq, <\}$, $w \in \mathbb{R}$, and $w \geq 0$.

We define the semantics for the new operators below. CE, in its original definition [Aumann, 1974], addresses the cooperative competition (abbrv. collaboration) within a group. On the other hand, ATL, with formula $\langle\langle G \rangle\rangle\psi$, mainly focuses on the pure competition between groups. The general idea of our new formula $CE_G^{w} \psi$ is to reason about the collaboration of agents in a group G in the form of CE to play against the other agents in achieving a group objective ψ and at the same time optimise the utilitarian value of the group. Based on this, $UCE_G^{w} \psi$ checks moreover whether the obtained CE is unique.

Collaboration and Competition Formula $CE_G^{w} \psi$ includes a set G of agents as one of its parameters. Given a pure joint strategy σ_G , the system $M_0 \sigma_G$ may have other agents $\text{Agt} \setminus G$ whose strategies have not been fixed. We follow the intuition used in the original ATL model checking that, for the joint strategy σ_G to be *valid*, the system $M_0 \sigma_G$ needs to satisfy the formula $\forall \psi$, which intuitively means that the strategy σ_G is able to counter all possible behaviour of agents in $\text{Agt} \setminus G$ to ensure the satisfaction of formula ψ . The formula ψ can be seen as a group objective of G .

Meanwhile, whether an agent $i \in G$ can get a reward w under joint strategy σ_G and state s is dependent on the satisfaction of both its own goals $Goal_i$ and the group objective ψ . Formally, we use $r_{s,i}(\sigma_G, \psi)$ to denote the highest reward agent i can get with respect to s, σ_G, ψ , i.e., $r_{s,i}(\sigma_G, \psi) = w$ if the following three conditions hold

- $M_0\sigma_G, s \models \forall\psi$,
- there exists a $(\phi, w) \in Goal_i$ such that $M_0\sigma_G, s \models \phi$ and
- there does not exist a $(\phi', w') \in Goal_i$ such that $w' > w$ and $M_0\sigma_G, s \models \phi'$.

We let $r_{s,i}(\sigma_G, \psi) = 0$ when either of the first two conditions is not satisfiable. The last two conditions ensure that w is the highest reward agent i can get. Moreover, we let $r_{s,i}(\sigma_G, \psi) = 1$ if $Goal_i = \emptyset$ and $M_0\sigma_G, s \models \forall\psi$. Now we can define CE in our context.

Definition 4 (Correlated Equilibrium) *Given a system M^\dagger and a set G of agents with a group objective ψ , a correlated equilibrium on a state s is a probabilistic distribution $\delta_G : \Gamma_G \rightarrow [0, 1]$ such that*

$$\sum_{\sigma_G \in \Gamma_G} \delta_G(\sigma_G) = 1 \quad (1)$$

$$\sum_{\sigma_{G \setminus \{i\}} \in \Gamma_{G \setminus \{i\}}} (r_{s,i}(\sigma_{G \setminus \{i\}}\sigma_i^1, \psi) - r_{s,i}(\sigma_{G \setminus \{i\}}\sigma_i^2, \psi)) \cdot \delta_G(\sigma_{G \setminus \{i\}}\sigma_i^1) \geq 0$$

for all $i \in G$ and all strategies $\sigma_i^1, \sigma_i^2 \in \Gamma_i$ (2)

Intuitively, Equation (1) says that a CE δ_G is a probabilistic distribution over the set of pure joint strategies of G , and Equation (2) says that, on a CE δ_G , if agent i is informed (by e.g., a signal) to play strategy σ_i^1 then it has no incentive to deviate from this for another strategy σ_i^2 . The deviation is defined as the difference between achievable rewards (i.e., $r_{s,i}(\sigma_{G \setminus \{i\}}\sigma_i^1, \psi) - r_{s,i}(\sigma_{G \setminus \{i\}}\sigma_i^2, \psi)$), normalised over the distribution δ_G on those joint strategies in which agent i takes strategy σ_i^1 . It is important to note that, when informing agent i to play its strategy σ_i^1 , the signal does not tell i the strategies for the other agents in G .

Example 4 (Continue with Example 3) *Simply taking Equation (1) and (2) with formula $\psi = \mathbf{T}$, our model checker can find a CE δ_G such that $\delta_G(\sigma_{xy}) = \delta_G(\sigma_{yx}) = 0.4$, $\delta_G(\sigma_{xx}) = 0$ and $\delta_G(\sigma_{yy}) = 0.2$. However, with this CE, there is a non-zero probability that both agents are informed by the signal to play the strategy σ_y , which will lead to an accident.*

This is similar for NE. There are three Nash equilibria, σ_{xy} , σ_{yx} , and a mixed strategy equilibrium. The mixed strategy NE has a non-zero probability for the accident to occur.

Therefore, in addition to the notorious equilibrium selection problem that agents may follow different equilibria and crash, the above example explains that the accident may still occur even when agents follow the same equilibrium: though both the car and the bus act rationally, there may exist a non-zero probability for a disastrous result to occur. That is, agents' rational behaviour with respect to a random equilibrium may be insufficient. Moreover, we note that the other two non-mixed NEs have lower utilitarian values comparing to the optimal CE, which will be explained below.

Optimal Equilibrium w.r.t. Utilitarian Value The set of correlated equilibria forms a polytope in the space $[0, 1]^{|\Gamma_G|}$ and different equilibria can have very different performance on selection criteria such as utilitarian (i.e., expected sum of the players' total rewards), egalitarian (i.e., expected sum of the weakest player's reward) and plutocratic (i.e., expected sum of the strongest player's reward), in which the expectation is based on the CE which is a distribution.

In this paper, we adopt utilitarian criterion⁴ [Greenwald and Hall, 2003] and choose a maximal point from the polytope by maximising the expected sum of the players' total rewards. Formally, to find a maximal CE is to maximise over the following objective function, under the constraints in Equation (1) and (2):

$$\text{maximise } \sum_{\sigma_G \in \Gamma_G} \delta_G(\sigma_G) \cdot \sum_{i \in G} r_{s,i}(\sigma_G, \psi) \quad (3)$$

Proposition 1 *The size of the linear program (1–3) is polynomial with respect to $|\Gamma_G|$.*

Let $\text{sat}_G(M, s, \psi) \in \{\mathbf{T}, \mathbf{F}\}$ be the result of whether there exists a solution to the defined optimisation problem, where \mathbf{T} and \mathbf{F} stand for the propositional true and false, respectively. If $\text{sat}_G(M, s, \psi) = \mathbf{T}$, we let $u_G(M, s, \psi)$ be the maximal utilitarian value and $\delta_G(M, s, \psi)$ be one of its associated CEs. It is noted that, there may exist multiple CEs whose utilitarian values are optimal.

Semantics The semantics of the formula $\text{CE}_G^{\text{pw}}\psi$ is defined as follows.

- $M, s \models \text{CE}_G^{\text{pw}}\psi$ if $\text{sat}_G(M, s, \psi) = \mathbf{T}$ and $u_G(M, s, \psi) \bowtie w$.

It is noted that, $M, s \models \text{CE}_G^{>0}\psi$ implies $M, s \models \langle\langle G \rangle\rangle\psi$, but not vice versa. The implication can be seen from the fact that $M, s \models \text{CE}_G^{>0}\psi$ implies that there must exist at least one pure joint strategy σ_G such that $M\sigma_G, s \models \psi$. The latter means that $M, s \models \langle\langle G \rangle\rangle\psi$. The unsatisfiability of the other direction can be seen from the possibility that all pure joint strategies satisfying $M\sigma_G, s \models \psi$ achieve a reward 0, depending on the definition of goals.

Example 5 (Continue with Example 3) *Consider the formula*

$$\phi_4 \equiv \text{CE}_{\{1,2\}}^{>7}\mathbf{T}$$

which asks for the optimal utilitarian value for the agents $\{1, 2\}$. By setting group objective as \mathbf{T} , the agents do not care about the behaviour of the environment (when no other agents in the system). By checking the relation $M, s \models \phi_4$ on the initial state $s = (s_a, s_a)$, we are able to find a CE δ_G such that $\delta_G(\sigma_{xy}) = \delta_G(\sigma_{yx}) = 0.25$, $\delta_G(\sigma_{xx}) = 0.5$ and $\delta_G(\sigma_{yy}) = 0$, whose associated utilitarian value $u_{\{1,2\}}(M, s, \mathbf{T})$ is 7.5. It is noted that, the undesirable joint strategy σ_{yy} will never occur. The two NEs σ_{xy} and σ_{yx} have utilitarian value 7.

We show that agents do not have an incentive to deviate from the strategy suggested by the signal. Assume that agent 1 (i.e., the bus) is suggested to follow the strategy σ_x and wait in front of a shared lane. If the other agent 2 (i.e., the

⁴The approach can be generalised to work with other mentioned criteria by updating Equation (3).

car) sticks with the strategy being suggested by the signal then, because agent 1 does not know which strategy agent 2 is asked to follow, it learns from the CE that agent 2 is asked to follow σ_x with probability $2/3$ and follow σ_y with probability $1/3$. Therefore, the expected sum of its own reward is $1/3 * r_{s,1}(\sigma_{xy}, T) + 2/3 * r_{s,1}(\sigma_{xx}, T) = 1/3 * 2 + 2/3 * 4 = 10/3$. If it changes its strategy into σ_y , then the expected sum of its own reward is $1/3 * r_{s,1}(\sigma_{yy}, T) + 2/3 * r_{s,1}(\sigma_{yx}, T) = 1/3 * 0 + 2/3 * 5 = 10/3 \neq 10/3$. Other cases follow the same reasoning.

The following example illustrates the group objective.

Example 6 (Continue with Example 5) Let $\phi_5 = \forall \square ((p_1 @ d \wedge p_2 @ d) \Rightarrow \forall \bigcirc \neg (p_1 @ d \wedge p_2 @ d))$ expresses that agents will not loop over the state in which both are at a decision phase. Consider

$$\phi_6 \equiv \text{CE}_{(1,2)}^{\geq 7} \phi_5$$

which asks for the optimal utilitarian value when agents are pursuing the group objective ϕ_5 . By checking the relation $M, s \models \phi_6$ on the initial state $s = (s_a, s_a)$, we are able to find a CE δ_G such that $\delta_G(\sigma_{xy}) = 1$ and $\delta_G(\sigma_{yx}) = \delta_G(\sigma_{xx}) = \delta_G(\sigma_{yy}) = 0$, whose associated utilitarian value is $7.0 < 7.5$. To see why $\delta_G(\sigma_{xx}) = 0$, it is noted that for the group objective ϕ_5 , the joint strategy σ_{xx} will disable agents to achieve rewards because it will make $r_{s,1}(\sigma_{xx}, \phi_5) = r_{s,2}(\sigma_{xx}, \phi_5) = 0$. So, only joint strategies σ_{xy} and σ_{yx} are valid strategies.

Moreover, because of the symmetry of the two strategies σ_{xy} and σ_{yx} , there are an infinite number of CEs δ_G such that $\delta_G(\sigma_{yx}) = p$, $\delta_G(\sigma_{xy}) = 1 - p$, and $\delta_G(\sigma_{xx}) = \delta_G(\sigma_{yy}) = 0$, for any $p \in [0, 1]$. Their optimal utilitarian value is the same.

Now consider the case in which the group of agents have opponents while the group has a trivial objective T.

Example 7 (Continue with Example 5) Assume that there is another agent 3 who can interfere with the group $\{1, 2\}$ of agents. Its interference behaviour is described as follows: prevent agent 2 from moving when both 1 and 2 are in the decision states. Intuitively, this can be explained as a traffic rule that a car has to give way to a bus, involuntarily or not. Let M' be the new system M by adding agent 3.

By checking $M', s \models \phi_4$ on the initial state $s = (s_a, s_a)$, we are able to find a CE δ_G which is the same as that in Example 6, i.e., $\delta_G(\sigma_{yx}) = 1$ and $\delta_G(\sigma_{xy}) = \delta_G(\sigma_{xx}) = \delta_G(\sigma_{yy}) = 0$. The difference exists in their generalisations. For this case, any δ_G is optimal if it is such that $\delta_G(\sigma_{yx}) = p$, $\delta_G(\sigma_{yy}) = 1 - p$, and $\delta_G(\sigma_{xx}) = \delta_G(\sigma_{xy}) = 0$, for some $p \in [0, 1]$. To see this, we note that no matter what agent 2's strategy is, after interfered by the new agent, it is the same as σ_x .

We remark that, by optimising over the utilitarian value, all the CEs obtained from the above three examples do not allow (with probability 0) the crash to occur. We argue that they are valid theoretical solutions to avoid the Google car accident.

Uniqueness of Optimal Equilibrium After having an optimal utilitarian value and a CE, one might still be interested to learn whether the obtained CE is unique, i.e., whether there exists another CE whose associated utilitarian value is also optimal. A unique CE avoids the equilibrium selection problem, upon which equilibrium notions receive many critiques.

Formally, after obtaining $\delta'_G = \delta_G(M, s, \psi)$, the existence of another optimal CE is equivalent to the existence of an optimal solution to the linear programming problem whose constraints and maximisation objective are those as in Equations (1), (2), and (3), together with one of the following constraints for some $\sigma_G \in \Gamma_G$

$$\delta_G(\sigma_G) > \delta'_G(\sigma_G) \text{ or } \delta_G(\sigma_G) < \delta'_G(\sigma_G) \quad (4)$$

such that the utilitarian value is the same as $u_G(M, s, \psi)$. Intuitively, Equation (4) means that δ_G and δ'_G assign different probabilities to at least one of the joint strategies.

Proposition 2 There are a number $|2 \cdot \Gamma_G|$ of linear programs (1–4), two for each $\sigma_G \in \Gamma_G$.

Let $usat_G(M, s, \psi, \sigma_G, d) \in \{T, F\}$ be the existence of a solution to one of the linear programs corresponding to the joint strategy σ_G and the left or right inequation such that $d \in \{>, <\}$. When $usat_G(M, s, \psi, \sigma_G, d) = T$, we define $uu_G(M, s, \psi, \sigma_G, d)$ and $u\delta_G(M, s, \psi, \sigma_G, d)$ as the corresponding utilitarian value and the CE, respectively.

The semantics of the formula $\text{UCE}_G^{\text{pw}} \psi$ is defined as follows.

- $M, s \models \text{UCE}_G^{\text{pw}} \psi$ if $M, s \models \text{CE}_G^{\text{pw}} \psi$ and there exists some $\sigma_G \in \Gamma_G$ and $d \in \{>, <\}$ such that $usat_G(M, s, \psi, \sigma_G, d) = T$ and $uu_G(M, s, \psi, \sigma_G, d) = u_G(M, s, \psi)$.

Example 8 (Continue with Example 7) It can be checked that, the CE found in Example 5 is unique, i.e.,

$$M, (s_a, s_a) \models \text{UCE}_{(1,2)}^{>7.5} T \text{ and } M, (s_a, s_a) \not\models \text{UCE}_{(1,2)}^{>7.5} T$$

Moreover, as already discussed, the CEs in Example 6 and Example 7 are not unique. We can check that

$$M, (s_a, s_a) \models \text{CE}_{(1,2)}^{>7} \phi_5 \text{ and } M, (s_a, s_a) \not\models \text{UCE}_{(1,2)}^{>7} \phi_5$$

$$M', (s_a, s_a) \models \text{CE}_{(1,2)}^{>7} T \text{ and } M', (s_a, s_a) \not\models \text{UCE}_{(1,2)}^{>7} T$$

Implementation of Results in Real-World Scenario For the implementation of a CE-based strategy in real-world scenarios, agents need to agree on playing rationally (i.e., deviate from a strategy only when such deviation can lead to a better reward) according to a correlated equilibrium, before the system starts. Therefore, the computation of CEs can be done offline, which is similar with the strategic reasoning by ATL model checking. It is noted that such computation needs the goals $Goal_i$ of all agents $i \in \text{Agt}$ to be publicly known. This is a usual assumption made for complete information systems: for human agents these can be evaluated by rational choice theory [Anand, 1993]; for artificial agents these are pre-programmed before deployment.

During the system execution, agents do not communicate with each other. Instead, there needs to be a signal suggesting them to play specific strategies. As suggested by most CE literature, for each group, the signal is implemented by an extra passive agent changing its state according to the precomputed CE. The state of the passive agent is publicly observable to all agents in the group, and every state of the passive agent is associated with a valid joint strategy in the support of the CE. A notable example of this implementation is the traffic lights. [Solan and Vieille, 2002] also discusses the correlated devices.

Example 9 (Continue with Example 5) We can use a 4-way traffic light (0,1,2,3) to act as the signal. On the light 0, agent 1 is informed to follow the strategy σ_x and agent 2 is informed to follow σ_y ; On the light 1, agent 1 is informed to follow σ_y and agent 2 is informed to follow σ_x ; On the light 2 and 3, both agents are informed to follow σ_x . The switches between lights can be implemented with a time-sharing system.

The implementation for Example 6 and 7 can be done in a similar way according to the value p .

5 Software Implementation and Complexity

We implemented the proposed framework into a prototype software model checker⁵. A language based on reactive modules [Alur and Henzinger, 1996] is employed to describe multiagent systems. The prototype tool implements an explicit-state model checking algorithm, see standard textbook such as [Clarke *et al.*, 1999; Alur *et al.*, 2002], to verify temporal formulas. The ATL strategic formulas are checked with the help of a SAT solver. For new formulas $CE_G^{\text{sw}}\psi$ and $UCE_G^{\text{sw}}\psi$, a linear constraint solver in SciPy library is used to solve their corresponding linear programming problems (either (1–3) or (1–4)). For satisfiability results, a CE strategy (and its utilitarian value) can be obtained by processing the information returned from the solver. For $UCE_G^{\text{sw}}\psi$, in the worst case, the solver needs to be invoked for $2|\Gamma_G|$ times. All the results in the running example are computed automatically within seconds by the model checker.

In the following, we analyse the computational complexity of the mode checking problem on ATL_{UCE} . Let ATL_{CE} be the fragment in which the UCE operator is removed. Suppose that the number of agents is constant. The complexity is dependent on the measurement. First, assume that the problem is measured over the number of states $|S|$, the number of joint actions $|Act|$, and the number of operators in the formula $|\phi|$.

Theorem 1 *If the complexity is measured over $|S|$, $|Act|$ and $|\phi|$ then model checking ATL_{UCE} is in 2-EXPTIME and model checking ATL_{CE} is in EXPTIME.*

The high complexity is because the set $|\Gamma_{Agt}|$ is exponential with respect to $|S|$. The following theorem confirms that the complexity is lower if it is measured over the number $|\Gamma_{Agt}|$ of pure joint strategies and the size of the formula.

Theorem 2 *If the complexity is measured over $|\Gamma_{Agt}|$ and $|\phi|$ then model checking ATL_{UCE} and ATL_{CE} is in polynomial time.*

6 Related Work

It is *not* new that in strategic model checking the ATL does not have the expressiveness to describe equilibrium notions. There are mainly two lines of research trying to counter this weakness. The first is based on the observation that strategies are explicitly quantified in equilibrium notions but implicitly quantified in ATL syntax. The research is focused on introducing additional syntactic structures to directly name and quantify strategies. In [van der Hoek *et al.*, 2005], a formula $C_i(\sigma_i, \phi)$ means that ϕ holds if agent i chooses the strategy σ_i .

With this additional structure, it shows that NE is expressible as a formula. In [Chatterjee *et al.*, 2010], the formulas $\forall x \in \Gamma_i : \phi(x)$ and $\exists x \in \Gamma_i : \phi(x)$ explicitly quantify over the strategies. The logic SL can express a special NE called winning secure equilibrium. In [Mogavero *et al.*, 2014], the formula $(i, x)\phi$ is to bind agent i to the strategy associated with the variable x . It shows that NE is expressible with the logic. [Aminof *et al.*, 2016] extends [Mogavero *et al.*, 2014] to express and check for uniqueness of NE, with goals expressed in LTL. In [Huang and van der Meyden, 2014b], the formula $e_i(x)$ expresses that agent i has the same local state at the current point with the global state represented in x . NE and perfect cooperative equilibrium are shown to be expressible. The second line of research is to reason about Nash equilibrium without the syntax of strategic logics. In [Toumi *et al.*, 2015], an equilibrium checker is to check whether a given strategy profile is an NE. Some complexity results are reviewed in [Wooldridge *et al.*, 2016], where the multi-agent systems are described with iterated boolean games (IBG). IBG can be exponentially more succinct than state-based systems as we do, and therefore their complexities are higher than ours. It is unclear whether CE can be easier than NE for IBG.

Both the above research lines work with non-probabilistic systems and consider only pure (i.e., non-mixed) Nash equilibria. As noted, CE is a concept based on probabilistic distributions. [Koller and Pfeffer, 1995] presents a tool Gala which computes Nash equilibrium by taking sequence form which is linear in the size of the game tree. Progresses on stochastic games have been survey in [Chatterjee and Henzinger, 2011], including reasoning based on NEs. This thread of research does not work with temporal logics, except for [Huang *et al.*, 2012; Huang and Luo, 2013; Chen *et al.*, 2013], which are based on extending ATL to stochastic systems and cannot reason about equilibrium notions. Moreover, the extension of ATL on probabilistic dimension is also investigated in the probabilistic software model checker PRISM [Chen *et al.*, 2013] on turn-based games (instead of concurrent games as most literature on multiagent systems does). As most of the research reviewed in the above, PRISM cannot compute equilibria.

For the computation of CEs, except for the linear programming approach as we employed, multiagent learning, see e.g., [Shoham *et al.*, 2007; Busoniu *et al.*, 2008] for comprehensive reviews, focuses on extending the reinforcement learning approaches to a multiagent setting. In particular, [Greenwald and Hall, 2003] presents an extension of the Q-learning algorithm to find a CE. However, the algorithm cannot guarantee the convergence because a multiagent setting breaks an essential assumption for Q-learning to converge: the environment is non-stationary for a learner [Sandholm, 2007].

7 Conclusion

In this paper, we integrate correlated equilibrium into ATL strategic model checking. New CE operators are introduced to describe a group of agents collaborating to compete with their opponents and at the same time optimising their utilitarian value. A software model checker has been implemented to enable automated reasoning. The complexity of the model checking problem is also studied.

⁵<https://gitlab.com/xiaowei.huang/rationality>.

References

- [Alur and Henzinger, 1996] R. Alur and T.A. Henzinger. Reactive modules. In *LICS1996*, pages 207–218, 1996.
- [Alur *et al.*, 1998] Rajeev Alur, Thomas A. Henzinger, F.Y.C. Mang, Shaz Qadeer, Sriram K. Rajamani, and Sardar Tasiran. Mocha: Modularity in model checking. In *CAV 1998*, pages 521–525, 1998.
- [Alur *et al.*, 2002] Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. Alternating-Time Temporal Logic. *JACM*, 49(5):672–713, 2002.
- [Aminof *et al.*, 2016] Benjamin Aminof, Vadim Malvone, Aniello Murano, and Sasha Rubin. Graded strategy logic: Reasoning about uniqueness of nash equilibria. In *AAMAS2016*, pages 698–706, 2016.
- [Anand, 1993] Paul Anand. *Foundations of Rational Choice Under Risk*. Oxford University Press, 1993.
- [Aumann, 1974] Robert Aumann. Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1:67–96, 1974.
- [Aumann, 1987] Robert J. Aumann. Correlated equilibrium as an expression of bayesian rationality. *Econometrica*, 55(1):1–18, 1987.
- [Busoniu *et al.*, 2008] Lucian Busoniu, Robert Babuska, and Bart De Schutter. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 38(2):156–172, 2008.
- [Čermák *et al.*, 2014] Petr Čermák, Alessio Lomuscio, Fabio Mogavero, and Aniello Murano. Mcmsl-k: A model checker for the verification of strategy logic specifications. In *CAV2014*, pages 525–532, 2014.
- [Chatterjee and Henzinger, 2011] Krishnendu Chatterjee and Thomas A. Henzinger. A survey of stochastic -regular games. *Journal of Computer and System Sciences*, 78(2):394–413, 2011.
- [Chatterjee *et al.*, 2010] Krishnendu Chatterjee, Thomas A. Henzinger, and Nir Piterman. Strategy logic. *Information and Computation*, 208(6):677–693, 2010.
- [Chen *et al.*, 2013] Taolue Chen, Vojtech Forejt, Marta Z. Kwiatkowska, David Parker, and Aistis Simaitis. Automatic verification of competitive stochastic systems. *Formal Methods in System Design*, 43(1):61–92, 2013.
- [Clarke *et al.*, 1999] E. M. Clarke, O. Grumberg, and D. Peled. *Model Checking*. The MIT Press, 1999.
- [Daskalakis *et al.*, 2009] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a nash equilibrium. *SIAM Journal on Computing*, 39(1):195–259, 2009.
- [Gilboa and Zemel, 1989] Itzhak Gilboa and Eitan Zemel. Nash and correlated equilibria: Some complexity considerations. *Games and Economic Behavior*, 1:80–93, 1989.
- [Goo, 2016] Google car accident report, Feb 2016.
- [Greenwald and Hall, 2003] Amy Greenwald and Keith Hall. Correlated q-learning. In *ICML 2003*, pages 242–249, 2003.
- [Huang and Luo, 2013] Xiaowei Huang and Cheng Luo. A logic of probabilistic knowledge and strategy. In *AAMAS 2013*, pages 845–852, 2013.
- [Huang and van der Meyden, 2014a] Xiaowei Huang and Ron van der Meyden. Symbolic model checking epistemic strategy logic. In *AAAI 2014*, pages 1426–1432, 2014.
- [Huang and van der Meyden, 2014b] Xiaowei Huang and Ron van der Meyden. A temporal logic of strategic knowledge. In *KR 2014*, 2014.
- [Huang *et al.*, 2012] Xiaowei Huang, Kaile Su, and Chenyi Zhang. Probabilistic alternating-time temporal logic. In *AAAI 2012*, pages 765–771, 2012.
- [Huang, 2015] Xiaowei Huang. Bounded model checking of strategy ability with perfect recall. *Artificial Intelligence*, pages 182–200, 2015.
- [Koller and Pfeffer, 1995] Daphne Koller and Avi Pfeffer. Generating and solving imperfect information games. In *IJCAI1995*, pages 1185–1193, 1995.
- [Mogavero *et al.*, 2014] Fabio Mogavero, Aniello Murano, and Giuseppe Perelli. Reasoning about strategies: On the model-checking problem. *ACM transactions on Computational Logic*, 2014.
- [Nash, 1950] John Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950.
- [Sandholm, 2007] Tuomas Sandholm. Perspectives on multiagent learning. *Artificial Intelligence*, 171:382–391, 2007.
- [Shoham *et al.*, 2007] Yoav Shoham, Rob Powers, and Trond Grenager. If multi-agent learning is the answer, what is the question? *Artificial Intelligence*, 171(7):365–377, 2007.
- [Solan and Vieille, 2002] Eilon Solan and Nicolas Vieille. Correlated equilibrium in stochastic games. *Games and Economic Behavior*, pages 362–399, 2002.
- [TheVerge, 2016] TheVerge. A google self-driving car caused a crash for the first time – a bad assumption led to a minor fender-bender, 2016.
- [Toumi *et al.*, 2015] Alexis Toumi, Julian Gutierrez, and Michael Wooldridge. A tool for the automated verification of nash equilibria in concurrent games. In *ICTAC 2015*, pages 583–594, 2015.
- [van der Hoek *et al.*, 2005] Wiebe van der Hoek, Wojciech Jamroga, and Michael Wooldridge. A logic for strategic reasoning. In *AAMAS 2005*, pages 157–164, 2005.
- [Wooldridge *et al.*, 2016] Michael Wooldridge, Julian Gutierrez, Paul Harrenstein, Enrico Marchioni, Giuseppe Perelli, and Alexis Toumi. Rational verification: From model checking to equilibrium checking. In *AAAI 2016*, pages 4184–4191, 2016.