The Alternating-Time μ-Calculus with Disjunctive Explicit Strategies

Merlin Göttlinger

Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany
merlin.goettlinger@fau.de

Lutz Schröder

Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany
lutz.schroeder@fau.de

Dirk Pattinson

The Australian National University, Canberra, Australia
dirk.pattinson@anu.edu.au

Abstract

Alternating-time temporal logic (ATL) and its extensions, including the alternating-time μ-calculus (AMC), serve the specification of the strategic abilities of coalitions of agents in concurrent game structures. The key ingredient of the logic are path quantifiers specifying that some coalition of agents has a joint strategy to enforce a given goal. This basic setup has been extended to let some of the agents (revocably) commit to using certain named strategies, as in *ATL with explicit strategies (ATLES)*. In the present work, we extend ATLES with fixpoint operators and strategy disjunction, arriving at the *alternating-time μ-calculus with disjunctive explicit strategies (AMCDES)*, which allows for a more flexible formulation of temporal properties (e.g. fairness) and, through strategy disjunction, a form of controlled non-determinism in commitments. Our main result is an ExpTime upper bound for satisfiability checking (which is thus ExpTime-complete). We also prove upper bounds QP (quasipolynomial time) and NP $\cap$ coNP for model checking under fixed interpretations of explicit strategies, and NP under open interpretation. Our key technical tool is a treatment of the AMCDES within the generic framework of coalgebraic logic, which in particular reduces the analysis of most reasoning tasks to the treatment of a very simple *one-step logic* featuring only propositional operators and next-step operators without nesting; we give a new model construction principle for this one-step logic that relies on a set-valued variant of first-order resolution.

2012 ACM Subject Classification Theory of computation → Modal and temporal logics; Computing methodologies → Multi-agent systems

Keywords and phrases Alternating-time logic, multi-agent systems, coalitional strength

Digital Object Identifier 10.4230/LIPIcs.CSL.2021.26

Funding *Merlin Göttlinger*: Work performed under the DFG Project *Reconstructing Arguments from Noisy Text (RANT)*, SCHR 1118/14-1 / SCHR 1118/14-2.
Lutz Schröder: Work performed under the DFG Project *RANT*, SCHR 1118/14-1 / SCHR 1118/14-2.

1 Introduction

Alternating-time temporal logic (ATL) [1] extends computation tree logic (CTL) with path quantifiers $\langle\langle A \rangle\rangle$ read "coalition A of agents has a (long-term) joint strategy to enforce". It is embedded into the *alternating-time μ-calculus (AMC)*, which instead of path quantifiers, features nested least and greatest fixpoints alongside the next-step coalition modalities $\langle\langle A \rangle\rangle \bigcirc$ ("A can enforce in the next step"). The AMC is strictly more expressive than ATL, e.g. supports fairness constraints.

Coalitional power in ATL and the AMC is measured without any restrictions on the moves chosen by the opponents. There has been interest in extensions of ATL where the power of the opponents can be constrained, e.g. by committing some of them to a particular strategy, allowing for statements such as "no matter what the other network actors do, Alice and Bob can collaborate to exchange keys via Server S provided that S adheres to the protocol". One such extension is provided in *ATL with explicit strategies (ATLES)* [29], which has path quantifiers $\langle\langle A\rangle\rangle_\rho$ additionally parametrized over a commitment ρ of some agents to given named strategies, read "provided that the commitments ρ are kept, A can enforce ...". This extension has substantial impact on expressiveness; e.g. unlike in basic ATL, the semantics of ATLES over history-free strategies differs from the one over history-dependent strategies.

Restricting opponents to fixed moves is, of course, quite drastic; as noted already in the conclusion of Walther [28, Chapter 4], it is desirable to allow for more permissive restrictions where the opponents can still pick among several designated moves, as in "Alice has a strategy to get her print job executed if Bob either cancels his large print job or splits it into several smaller ones". In the present paper, we introduce such an extension with disjunctive commitments. Additionally, we include full support for least and greatest fixpoint operators, with associated gains in expressivity analogous to the extension from ATL to the AMC. We thus arrive at the *alternating-time μ-calculus with disjunctive explicit strategies (AMCDES)*.

Our main result on this logic is that satisfiability checking remains only ExpTime-complete (i.e. no harder than the AMC, or in fact than basic ATL or even CTL). We note also that (following a distinction made also in work on ATLES [29]) model checking is in quasipolynomial time QP and in NP ∩ coNP under fixed interpretation of explicit strategies (matching the best known bounds for the AMC and in fact even the plain relational μ-calculus), and in NP under open interpretation; these results are obtained by fairly straightforward adaptation of results on the AMC [11], and therefore discussed in full only in the appendix. We obtain our results by casting the AMCDES as an instance of *coalgebraic logic* [5], a unifying framework for modal and temporal logics. The driving principle of coalgebraic logic is to reduce reasoning tasks to the analysis of a simple *one-step logic*, whose formulae employ only Boolean connectives and a single layer of next-step modalities [22, 4, 11]. In particular, the automata- and game-theoretic machinery needed for the treatment of fixpoint logics is entirely encapsulated in results on the coalgebraic μ-calculus [4, 11]. The actual technical work then lies in providing algorithms, axiomatizations, and model constructions for the one-step logic of AMCDES, still posing substantial challenges due to nested quantification over strategies. The model construction principle for the one-step logic that we employ is based on a set-valued variant of first-order resolution that we introduce here, along with an associated notion of equationally complete model that we use to move from (generally infinite) Herbrand universes to finite models; this principle is the key to supporting strategy disjunction.

Related Work. Many ATL extensions are concerned with commitments of agents to strategies. Besides ATL with explicit strategies (ATLES), this includes, e.g., counterfactual ATL [26], which differs from ATLES by making commitments irrevocable. ATL with actions (ATL-A) [30] has per-agent disjunctive commitments (while the AMCDES allows disjunctions over joint commitments). ATL-A admits polynomial-time model checking; satisfiability checking is not considered (it would be somewhat simpler than in the present setting, as in ATL-A all actions are named, and hence known in advance). ATL with explicit actions (ATLEA) [12] features commitments of agents to a given action at only the current world, and has a fairly straightforward satisfiability-preserving embedding into the AMCDES.

Various forms of *strategy logic* [3, 15, 16] possibly contain ATL* with disjunctive explicit strategies (but presumably not the AMCDES or even the AMC, as they lack fixpoint operators); they tend to be computationally much harder than the AMCDES. Goranko and Ju [7] discuss various forms of conditional strategic modalities, one of which (O_{dd}) is similar in spirit to our strategy disjunction in that it restricts the moves of the opposition, however not to given named moves but rather to moves enforcing a given goal; their main technical result is a Hennessy-Milner style expressiveness theorem. De Nicola and Vandraager [18] consider disjunction of named actions in labelled transition systems, which in that setting can be encoded into next-modalities for single actions using logical disjunction.

Organization. We introduce the syntax and the semantics of the alternating-time μ-calculus with disjunctive explicit strategies (AMCDES) in Section 2. After recalling the requisite principles of coalgebraic logic in Section 3 we introduce the method of set-valued first-order resolution in Section 4. We illustrate these methods on the basic AMC in Section 5, and establish our main results on satisfiability checking for the AMCDES in Section 6.

2 AMC With Disjunctive Explicit Strategies

We proceed to introduce the syntax and semantics of the alternating-time μ-calculus with disjunctive explicit strategies (AMCDES). As indicated in Section 1, the logic is inspired by ATL with explicit strategies (ATLES) [29]. We deviate from the ATLES syntax in that we express (disjunctive) commitments of agents by means of names for strategies in the syntax. Also, we shorten the ATL syntax for next-step operators from $\langle\langle C\rangle\rangle\bigcirc$ ("C can enforce in the next step that …") to $[C]$ as in coalition logic [20]. We thus arrive at modalities $[C, O]$ where O is a set of named joint strategies for agents in a further coalition D of agents restricted in their choice of strategies, disjoint from C, read "if the agents in D use one of the joint strategies in O, then C can enforce that …". The dual modality $\langle C, O\rangle$ is read "even if the agents in D are limited to the joint strategies in O, C cannot prevent that …". Formally, our syntax is defined as follows.

▶ **Definition 2.1.** The syntax of the AMCDES is parametrized over a set At of (propositional) *atoms*, V of *variables*, a finite set Σ of agents (for technical simplicity, assumed to be linearly ordered), and sets M_j of *explicit strategies* (i.e. names for strategies) per agent j; *we fix these data from now on.* A *coalition* is a subset of Σ. We also (and mainly) refer to explicit strategies as *explicit moves*. We write $M_D = \prod_{j\in D} M_j$ for the set of *joint explicit moves* of a coalition D. Formulae ϕ, ψ are then given by the grammar

$$\phi, \psi ::= p \mid \neg p \mid x \mid \top \mid \bot \mid \phi \wedge \psi \mid \phi \vee \psi \mid [C, O]\,\phi \mid \langle C, O\rangle\,\phi \mid \mu x.\,\phi \mid \nu x.\,\phi$$

where $x \in V$, $p \in \mathsf{At}$, and $C \subseteq \Sigma$, i.e. a *coalition*. We generally write $\overline{C} = \Sigma \setminus C$. Moreover, $O \subseteq M_D$ is a set of joint explicit moves, called a *disjunctive explicit strategy* (or *move*), for some coalition D, disjoint from C, that we denote by $Ag(O)$. We call a modality $[C, O]$ or $\langle C, O\rangle$ a *grand coalition modality* if $C \cup Ag(O) = \Sigma$, and *non-disjunctive* if $|O| = 1$, in which case we often omit set brackets and just write O as its single element. We restrict grand coalition modalities to be non-disjunctive (cf. Remark 2.8). As usual, μ and ν take least and greatest fixpoints, respectively. Negation $\neg$ is not included but can be defined in the standard way, taking negation normal forms. The AMC with explicit strategies (AMCES) is the fragment of the AMCDES allowing only non-disjunctive modalities.

The AMCDES thus subsumes both the standard AMC [1] (with $[C]$ corresponding to $[C, O]$ with $Ag(O) = \emptyset$) and the history-free variant of ATLES [29] (which as we will detail in Remark 2.7 is the variant to which previous technical results refer).

▶ **Example 2.2.** The formula indicated in the introduction,

$$[\mathsf{Alice}, (\mathsf{Bob}\colon \{\mathsf{cancelPrint}, \mathsf{splitPrint}\})]\ \mathsf{printed}$$

says (using hopefully self-explanatory human-readable syntax for disjunctive explicit moves) that "Alice has a strategy to have her print job executed, provided that Bob opts to either cancel his print job or to split it into smaller jobs". The fixpoint formula

$$\nu x.\, \neg\mathsf{corrupted} \wedge [\mathsf{ECC}, (\mathsf{Env}\colon \{\mathsf{0\text{-}flips}, \mathsf{1\text{-}flip}\})]\, x$$

expresses that ECC memory can ensure that the stored data is not corrupted provided that in each cycle the environment flips either one or zero bits. The formula

$$\nu x.\, \neg\mathsf{intrusion} \wedge \langle \mathsf{Attacker}, (\mathsf{IPS}\colon \{\mathsf{dropPackage}, \mathsf{blockIP}\})\rangle\, x$$

expresses that "No matter what an attacker tries, the intrusion prevention system can always drop suspicious packets or block his IP address to prevent illegitimate access to company resources".

▶ Remark 2.3. One can encode an extension ATLDES of ATL with disjunctive explicit strategies into the AMCDES, e.g. defining $\langle\langle C, O\rangle\rangle(G\,\phi)$ "C can enforce that ϕ always holds, provided that $Ag(O)$ are committed to play strategies in O" as

$$\langle\langle C, O\rangle\rangle(G\,\phi) := \nu x.\, \phi \wedge [C, O]\, x.$$

The AMCDES is more expressive than ATLDES in this sense; e.g. for $C = \{\mathsf{client}\}$ and $O = (\mathsf{server}\colon \{\mathsf{protocol}, \mathsf{recover}\})$ the formula $\nu x.\, \mu y.\, (\mathsf{granted} \wedge [C, O]\, x) \vee [C, O]\, y$ says that "client can enforce that his requests are granted infinitely often, provided that server always either keeps to the protocol or immediately recovers when failures occur" (a specification that may, of course, hold or fail in a given system).

Note that the definition of $\langle\langle C, O\rangle\rangle$ allows $Ag(O)$ to choose their joint move from O anew in each step, like in the fixpoint formulae of Example 2.2, which in fact belong to the ATLDES fragment of the AMCDES. To illustrate that this is really the reasonable choice of a semantics for ATLDES (as opposed to letting O choose only in the beginning of a play), consider a situation where players K (*Kangaroo*) and M (*Marc-Uwe*) [14] play rock-paper-scissors (R, P, S) for an indefinite number of rounds, say to determine daily who does the dishwashing, until someone quits. Let the model include memory for the moves in the previous round, and atoms p "at least two rounds have been played" and k "K won the previous round". Consider the ATL with disjunctive explicit strategies (ATLDES) formula

$$\mathsf{rigged} = \langle\langle K, (M\colon \{R, P, S\})\rangle\rangle\, G(p \to k)$$

"K wins all rounds after the first if M keeps playing". In ATLDES, rigged does not hold in the model, as one would expect. If M could make his choice of R, P, S only once (in reality, sadly, he does just that [13]), then rigged would in fact hold.

We proceed to define the semantics, which is based on concurrent game structures [1] extended with interpretations of explicit moves.

▶ **Notation 2.4.** For $k \in \mathbb{N}$, we write $[k] = \{1, \dots, k\}$. For $C \subseteq \Sigma$ and a tuple $(k_j)_{j \in C} \in \mathbb{N}^C$, we put $[k_C] = \prod_{j \in C}[k_j]$. Given $m \in [k_C]$ and $D \subseteq C$, we write $m|_D$ for the restriction of m to an element of $[k_D]$. We write $n \sqsubseteq m$ if $n = m|_{Ag(n)}$, and $n =_\sqcap m$ if $n|_{Ag(n) \cap Ag(m)} = m|_{Ag(n) \cap Ag(m)}$. We write $\mathcal{P}X$ for the powerset of a set X.

▶ **Definition 2.5.** A *concurrent game structure with explicit strategies (CGSES)* is a tuple (W, k, v, f, ι) consisting of

- a finite set W of *states*,
- for each agent j and each state w, a natural number $k_j^w \geq 1$ determining the set of *moves* available to agent j at state w to be $[k_j^w]$,
- for each state $w \in W$,
  - a set $v(w) \subseteq \mathsf{At}$ of propositional atoms true at w,
  - an *outcome function* $f^w : [k_\Sigma^w] \to W$, and
  - for each agent j, a *move interpretation* $\iota_j^w : M_j \to [k_j^w]$.

For a joint explicit move $m \in M_D$, we just write $\iota^w(m)$ for the joint move with components $\iota_j^w(m_j)$ for $j \in D$. We use function image notation $\iota^w[O]$ to denote the result of applying ι^w to each joint move in the set O. The semantics of the AMCDES is then defined by assigning to each formula ϕ an extension $[\![\phi]\!]_S^\sigma \subseteq Q$, which depends on a CGSES $S = (W, k, v, f, \iota)$ and a valuation $\sigma : V \to \mathcal{P}W$. The propositional cases are standard (e.g. $[\![p]\!]_S^\sigma = \{w \in W \mid p \in v(w)\}$, $[\![x]\!]_S^\sigma = \sigma(x)$, $[\![\top]\!]_S^\sigma = W$, and $[\![\phi \wedge \psi]\!]_S^\sigma = [\![\phi]\!]_S^\sigma \cap [\![\psi]\!]_S^\sigma$). The remaining clauses are

$$
\begin{aligned}
[\![[C, O]\,\phi]\!]_S^\sigma &= \{w \in W \mid \exists m_C \in [k_C^w].\, \forall m_\Sigma \in [k_\Sigma^w].\\
&\qquad\qquad (m_C \sqsubseteq m_\Sigma \wedge m_\Sigma|_{Ag(O)} \in \iota^w[O]) \Rightarrow f^w(m_\Sigma) \in [\![\phi]\!]_S^\sigma\}\\
[\![\langle C, O\rangle\,\phi]\!]_S^\sigma &= \{w \in W \mid \forall m_C \in [k_C^w].\, \exists m_\Sigma \in [k_\Sigma^w].\\
&\qquad\qquad m_C \sqsubseteq m_\Sigma \wedge m_\Sigma|_{Ag(O)} \in \iota^w[O] \wedge f^w(m_\Sigma) \in [\![\phi]\!]_S^\sigma\}\\
[\![\mu x.\, \phi(x)]\!]_S^\sigma &= \bigcap\{B \subseteq W \mid [\![\phi(x)]\!]_S^{\sigma[x \mapsto B]} \subseteq B\}\\
[\![\nu x.\, \phi(x)]\!]_S^\sigma &= \bigcup\{B \subseteq W \mid B \subseteq [\![\phi(x)]\!]_S^{\sigma[x \mapsto B]}\}
\end{aligned}
$$

where $\sigma[x \mapsto B]$ denotes σ updated to return B on input x; and $[\![\langle C, O\rangle\,\phi]\!]_S^\sigma = [\![\neg\,[C, O]\,\neg\phi]\!]_S^\sigma$. That is, μ and ν take least and greatest fixpoints according to the Knaster-Tarski fixpoint theorem. At a state w, $[C, O]\,\phi$ holds if the agents in C have a joint move such that a state satisfying ϕ is reached no matter what the other agents do, as long as the agents in $Ag(O)$ play one of the joint moves in O. Dually, $\langle C, O\rangle\,\phi$ holds at w if whatever the agents in C do, the other agents have a joint move that leads to an outcome in ϕ and in which the joint move of $Ag(O)$ is in O.

▶ Remark 2.6. In the modal operators $[C, O]$, $Ag(O)$ is in opposition to C. One may envision an alternative setup where $Ag(O)$ is instead made a part of C. However, then $[C, O]\,\phi$ would become equivalent to $\bigvee_{m \in O}[C, \{m\}]\,\phi$, hence expressible already in ATLES. We thus opt for our present more expressive version where $Ag(O)$ and C are disjoint. Note that $[C, O]\,\phi$ then is *not* equivalent to $\bigwedge_{m \in O}[C, \{m\}]\,\phi$: The latter formula allows C to use different moves against each $m \in O$, while in $[C, O]\,\phi$, the *same* joint move of C must work against every $m \in O$.

▶ Remark 2.7. The above semantics uses history-free strategies (i.e. ones that look only at the present state, not the history of previously visited states). While basic ATL is insensitive to whether it is interpreted over history-free or history-dependent strategies [1], ATLES does distinguish these semantics [29]. Although this may not be always apparent from the phrasing, all technical results on ATLES in Walther et al. [29] are meant to apply to the semantics over history-free strategies only[1] (in particular the fixpoint unfolding axioms [29, Figure 1] clearly hold only over the history-free semantics). Note that the basic AMC, which the AMCDES extends, similarly is interpreted over history-free strategies (and nevertheless includes ATL*, which is history-dependent [1]).

▶ Remark 2.8. The interdiction of proper strategy disjunction in grand coalition modalities is needed (only) for the upper bound on satisfiability checking (Section 6); our results on model checking (Section 2) would actually not need this restriction. The fragment we term AMCES in Definition 2.1 does include grand coalition modalities with (non-disjunctive) explicit strategies. It is hence more permissive on these modalities than the original version of ATLES [29], where the set of agents is made variable, which for purposes of satisfiability is equivalent to excluding grand coalition modalities.

We note that the axiomatization we present later and its completeness proof become much simpler if one excludes the grand coalition completely (like, effectively, in ATLES): E.g. in the rule (C) for basic coalition logic / ATL (Section 5), the literals $\langle\Sigma\rangle\, c_j$ disappear; and in the proof of one-step tableau completeness (Theorem 5.1), one can, in this simplified setting, just use a single move $\perp$ as witness for all $\langle C_j\rangle\, c_j$ in Ξ, using non-determinism to ensure satisfaction of the $\langle C_j\rangle\, c_j$. This is discussed in detail in Appendix B.

Model Checking

Walther et al. [29] consider two variants of the model checking problem that differ on whether the interpretation of explicit strategies is considered part of the model (*fixed*) or to be found by the model checking algorithm (*open*). They show for ATLES that if strategies are restricted to be history-free, then the problem is P-complete under fixed interpretation, and NP-complete under open interpretation, with the upper bound being by straightforward guessing of history-free strategies. The complexity for the history-dependent variant remains open.

We obtain upper bounds for model checking in the AMCDES using generic results on the coalgebraic μ-calculus [11]:

▶ **Theorem 2.9.** *Model checking for the full AMCDES is in NP $\cap$ coNP as well as in QP under fixed interpretation of explicit strategies, and in NP under open interpretation.*

We defer a summary of the requisite results in coalgebraic logic and the proof of Theorem 2.9 to Appendix A, as the details are mostly by simple adaptation from the AMC [11].

3 Preliminaries: Coalgebraic Logic

We will employ the machinery of coalgebraic logic to obtain our main complexity results; we recall basic definitions and tools, using the standard AMC as our running example.

Coalgebraic logic [5] is a uniform framework for modal and temporal logics interpreted over state-based systems. It parametrizes the **semantics** of logics over the type of such systems, encapsulated in a *functor* F on the category of sets. Such a functor assigns to each set X a set FX and to each map $f : X \to Y$ a map $Ff : FX \to FY$, preserving identities and composition. We think of the elements of FX as structured collections over X. Systems are then F-*coalgebras*, i.e. pairs (W, γ) consisting of a set W of *states* and a *transition map*

[1] Personal communication with the authors

$\gamma : W \to FW$, which thus assigns to each state a structured collection of successors. Our leading example is the functor G that maps a set X to the set

$$\mathsf{G}X = \{((k_j)_{j\in\Sigma}, f) \mid (k_j) \in \mathbb{N}^{\Sigma}_{\geq 1}, f : (\prod_{j\in\Sigma}[k_j]) \to X\}$$

of *one-step games* over X. G-Coalgebras are essentially concurrent game structures (CGSs) [1] without the interpretation of propositional atoms, as they assign to each state numbers k_j of available moves for the agents and an outcome function f. Propositional atoms are covered by extending G to $\mathsf{G_p}X = \mathcal{P}\mathsf{At} \times \mathsf{G}X$; although the logic becomes trivial without propositional atoms, we mostly elide their explicit treatment, which is straightforward and can be dealt with using fusion results in coalgebraic logic [23]. To obtain CGSESs, we extend G to the functor $\mathsf{G_{ES}}$ with $\mathsf{G_{ES}}X$ consisting of *one-step games with explicit strategies* $((k_j), f, \iota)$ over X, where $((k_j), f)$ is a one-step game over X and $\iota_j \colon M_j \to [k_j]$ (for $j \in \Sigma$) interprets explicit strategies; we use the same notation for ι as introduced for ι^w in Section 2.

The **syntax** of coalgebraic logics is then parametrized over the choice of a set Λ of (next-step) *modal operators* with assigned finite arities; nullary modalities are just propositional atoms. For readability, we assume in the technical treatment that all modalities are unary. We require that for every $\heartsuit \in \Lambda$ there is a *dual* operator $\overline{\heartsuit} \in \Lambda$. The *coalgebraic μ-calculus* [4] over Λ then has formulae ϕ, ψ given by the grammar

$$\phi, \psi ::= \top \mid \bot \mid x \mid \phi \wedge \psi \mid \phi \vee \psi \mid \heartsuit\, \phi \mid \mu x.\, \phi \mid \nu x.\, \phi$$

where x ranges over a reservoir V of *fixpoint variables*, and $\heartsuit$ over Λ. The operators μ and ν take least and greatest fixpoints, respectively. Again, negation is definable. We assume a representation of the modalities in Λ as strings over some alphabet, with an ensuing notion of *representation size* for formulae and modalities.

Over F-coalgebras, a modal operator $\heartsuit \in \Lambda$ is interpreted by assigning to it a *predicate lifting* $[\![\heartsuit]\!]$, which is a family of maps $[\![\heartsuit]\!]_X$, indexed over all sets X, that assign to each subset $Y \subseteq X$ a subset $[\![\heartsuit]\!]_X(Y) \subseteq FX$, subject to a naturality condition. To enable fixpoint formation, we require $[\![\heartsuit]\!]_X$ to be monotone w.r.t. subset inclusion. Moreover, we require predicate liftings to respect duals, i.e. $[\![\overline{\heartsuit}]\!]_X(Y) = FX \setminus [\![\heartsuit]\!]_X(X \setminus Y)$. Given an F-coalgebra $C = (W, \gamma)$ and a valuation $\sigma : V \to \mathcal{P}W$, the semantic clauses defining the extension $[\![\phi]\!]^\sigma_C \subseteq W$ of a formula ϕ are then the standard ones for the Boolean connectives; μ and ν take least and greatest fixpoints in the same way as made explicit for the AMCDES in Section 2; and

$$[\![\heartsuit\, \phi]\!]^\sigma_C = \gamma^{-1}[[\![\heartsuit]\!]_W([\![\phi]\!]^\sigma_C)].$$

We fix the data F, Λ, $[\![\heartsuit]\!]$ for the remainder of this section.

▶ **Example 3.1.** The AMC is cast as a coalgebraic μ-calculus by interpreting the modality $[C]$ over the functor G by the predicate lifting

$$[\![[C]]\!]_X(Y) = \{((k_j), f) \in \mathsf{G}X \mid \exists m_C \in [k_C].\, \forall m \in [k_\Sigma].\, m_C \sqsubseteq m \Rightarrow f(m) \in Y\}$$

(using notation introduced in Section 2). The more general modalities $[C, O]$ of AMCDES are interpreted by a predicate lifting that correspondingly lifts a predicate Y on X to the set of all one-step games with explicit strategies $((k_j), f, \iota) \in \mathsf{G_{ES}}X$ such that there exists a joint move $m_C \in [k_C]$ such that $f(m) \in Y$ for all $m \in [k_\Sigma]$ such that $m_C \sqsubseteq m$ and $\iota(n) \sqsubseteq m$ for some $n \in O$.

Satisfiability checking in coalgebraic logics can be based on the provision of a complete set of tableau rules for the next-step modal operators [22, 4]. The basic example of such a rule is the tableau rule $\Box a_1, \dots, \Box a_n, \Diamond b / a_1, \dots, a_n, b$ for standard modal logic, which says essentially that in order to satisfy $\Box a_1 \wedge \dots \wedge \Box a_n \wedge \Diamond b$, we need to generate a successor state satisfying $a_1 \wedge \dots \wedge a_n \wedge b$. Formal definitions are as follows.

▶ **Definition 3.2** (One-step tableau rules)**.** Fix a supply V of *(propositional) variables*, serving as placeholders for formulae in rules. A *(monotone) one-step (tableau) rule* has the form

$$\frac{\Phi}{\Theta_1 \mid \dots \mid \Theta_n} \qquad (n \geq 0)$$

where the *conclusions* $\Theta_1, \dots, \Theta_n$ are finite subsets of V, read as finite conjunctions, and the *premiss* Φ is a finite subset of the set $\Lambda(\mathsf{V}) = \{\heartsuit a \mid \heartsuit \in \Lambda, a \in \mathsf{V}\}$ of *modal atoms*, also read conjunctively; additionally, we require that Φ mentions each variable at most once, and the Θ_i mention only variables occurring in Φ. Given a set X and a $\mathcal{P}X$-valuation $\tau : \mathsf{V} \to \mathcal{P}X$, we interpret such a Θ_i as $[\![\Theta_i]\!]\tau = \bigcap_{a \in \Theta_i} \tau(a)$, and Φ as $[\![\Phi]\!]\tau = \bigcap_{\heartsuit a \in \Phi} [\![\heartsuit]\!]_X(\tau(a)) \subseteq FX$.

The rule $\Phi/\Theta_1 \mid \dots \mid \Theta_n$ is *one-step tableau sound* if $[\![\Theta_i]\!]\tau \neq \emptyset$ for some i whenever $[\![\Phi]\!]\tau \neq \emptyset$. Let $\mathcal{R}$ be a set of one-step tableau rules, closed under injective renaming of variables. Then $\mathcal{R}$ is *one-step tableau complete* if the following condition holds: For all X, $\tau : \mathsf{V} \to \mathcal{P}X$, and $\Xi \subseteq \Lambda(\mathsf{V})$, whenever for each rule $\Phi/\Theta_1 \mid \dots \mid \Theta_n \in \mathcal{R}$ such that $\Phi \subseteq \Xi$, we have $[\![\Theta_i]\!]\tau \neq \emptyset$ for some i, then $[\![\Xi]\!]\tau \neq \emptyset$.

We will give one-step tableau sound and complete sets of rules for the AMCDES in Section 6. To obtain complexity results, rule sets formally need to be *ExpTime-tractable*, meaning that rule matches are encodable as strings over some alphabet such that all rule matches to a given set of formulae can be represented by polynomially sized codes and moreover basic operations on codes (well-formedness check, check for rule matching, access to conclusions) can be performed in exponential time [22, 4]; we refrain from elaborating details, as all rule sets we consider here will be clearly computationally harmless. The main benefit that we draw from these rule sets is the following generic upper complexity bound.

▶ **Theorem 3.3** (Satisfiability checking [4])**.** *If a coalgebraic μ-calculus admits an ExpTime-tractable one-step tableau complete set of one-step tableau sound rules, then its satisfiability problem is in ExpTime.*

In the algorithm underlying the above theorem, one-step rules combine with standard tableau rules for propositional and fixpoint operators. The arising tableaux need to be checked for bad branches (where least fixpoints are unfolded indefinitely) using dedicated parity automata, which combine with the tableau to form the *tableau game*, a parity game that is won by Eloise iff the target formula is satisfiable.

4 Set-Valued First-Order Resolution

For use in completeness proofs of modal rules, we next introduce *set-valued first-order resolution*, an adaptation of the standard first-order resolution method [6] to a logic of *outcome models* $\mathcal{G} = ((S_j)_{j \in \Sigma}, f, W, [\![-]\!])$ where the S_j are sets and W is a finite set, $[\![-]\!]$ interprets sorted algebraic operations over the S_j, and $f : \left(\prod_{j \in \Sigma} S_j\right) \to W$ is an outcome function. One-step games in $\mathsf{G}W$ are (operation-free reducts of) outcome models where the S_j are finite; for the time being, we allow infinite S_j for readability, explaining in the proof sketches in Sections 5 and 6 how finiteness can be regained. Formulae of *set-valued first-order* *logic* are clause sets formed over literals of the form $A(\bar{t})$ where $A \subseteq W$ and $\bar{t}$ is an Σ-tuple of terms (i.e. a clause is a finite set of literals, read disjunctively, and a clause set is a finite set of clauses, read conjunctively). Terms live in a sorted setting with one sort j (interpreted as S_j) for each agent j, and the j-th term in $\bar{t}$ has sort j. Terms are built from sorted variables and function symbols with given sort profiles (e.g. $g : 1 \times 0 \to 2$ takes moves of agents 1 and 0, and produces a move of agent 2) in the standard way, ensuring well-sortedness. Function symbols are interpreted as sorted functions on the S_j, respecting the sort profile; this induces an interpretation of (tuples of) terms depending on sort-respecting valuations of the variables as usual. We write $[\![\bar{t}]\!]\eta$ for the interpretation of a tuple $\bar{t}$ of terms under a valuation η. An outcome model $\mathcal{G}$ as above *satisfies* a literal $A(\bar{t})$ under a valuation η (notation: $\mathcal{G}, \eta \models A(\bar{t})$) if $f([\![\bar{t}]\!]\eta) \in A$, and $\mathcal{G}$ satisfies a clause Γ under η (notation: $\mathcal{G}, \eta \models \Gamma$) if $\mathcal{G}, \eta \models A(\bar{t})$ for some literal $A(\bar{t})$ in Γ. Finally, $\mathcal{G}$ *satisfies* a clause Γ (notation: $\mathcal{G} \models \Gamma$) if $\mathcal{G}, \eta \models \Gamma$ for every valuation η. A clause set is satisfiable if there exists an outcome model that satisfies all its clauses. We will generate clauses from modal atoms in $\Lambda(\mathsf{V})$ (Definition 3.2); e.g. given a $\mathcal{P}W$-valuation $\tau : \mathsf{V} \to \mathcal{P}W$, modalized atoms $[C]\,a$ and $\langle C\rangle\,a$ induce singleton clauses of the form

$$\{\tau(a)(e_C, x_{\overline{C}})\} \qquad \text{(for } [C]\,a) \qquad (1)$$

$$\{\tau(a)(x_C, g_{\overline{C}}(x_C))\} \qquad \text{(for } \langle C\rangle\,a) \qquad (2)$$

respectively, where x_C, $x_{\overline{C}}$ are tuples of variables (implicitly universally quantified, and representing moves for the agents in C and $\overline{C}$, respectively); e_C is a family of Skolem constants witnessing the ability of C to force a; and $g_{\overline{C}}$ is a family of Skolem functions producing countermoves $g_{\overline{C}}(x_C)$ for the agents in $\overline{C}$ that keep C from enforcing $\neg a$ using x_C. Of course these symbols are fresh so that clauses induced by different modalized atoms have disjoint sets of function symbols and variables, which we will later distinguish via superscripts in proofs.

We implicitly normalize clauses to mention each tuple of terms at most once (rewriting $A(\bar{t}), B(\bar{t})$ into $(A \cup B)(\bar{t})$), and operate on clauses using the *(set-valued) resolution rule*

$$(SR)\ \frac{\Gamma, A(\bar{t}) \qquad B(\bar{u}), \Delta}{\Gamma\sigma, (A \cap B)(\bar{t}\sigma), \Delta\sigma}$$

where σ is the most general unifier (mgu) of $\bar{t}$ and $\bar{u}$, with variables in the premises made disjoint by suitable renaming; as usual, we write "," for union of clauses and omit set brackets around singleton clauses (so $\Gamma, A(\bar{t})$ is shorthand for $\Gamma \cup \{A(\bar{t})\}$). A clause is *blatantly inconsistent* if all its literals are of the form $\emptyset(\bar{t})$. A clause set ϕ is *blatantly inconsistent* if it contains a blatantly inconsistent clause, and *inconsistent* if a blatantly inconsistent clause can be derived from it using the resolution rule; otherwise, ϕ is *consistent*.

Recall that unification can fail either due to a *clash*, i.e. when terms with distinct head symbols need to be unified, or at the *occurs check*, which happens when a variable needs to be unified with a term that contains it. In particular, this happens in clauses (2) associated to diamonds: E.g. the modal atoms $\langle\{0\}\rangle\,a$ and $\langle\{1\}\rangle\,b$ generate clauses $\{\tau(a)(x_0, g_1^1(x_0))\}$ and $\{\tau(b)(g_0^2(x_1), x_1)\}$, whose (tuples of) argument terms fail to unify since no substitution solves $x_0 = g_0^2(g_1^1(x_0))$.

Set-valued propositional resolution in *set-valued propositional logic* simplifies the above setup by replacing tuples $\bar{t}$ of terms in literals $A(\bar{t})$ with elements y of some index set Y; models are then just functions $f : Y \to W$, and f *satisfies* a literal $A(y)$ if $f(y) \in A$. The resolution rule is just like the above but of course does not involve unification and substitution, i.e. just derives $\Gamma, (A \cap B)(y), \Delta$ from $\Gamma, A(y)$ and $B(y), \Delta$.

▶ **Theorem 4.1** (Soundness and completeness of set-valued resolution). *A clause set in set-valued propositional (first-order) logic is satisfiable iff it is consistent under set-valued propositional (first-order) resolution.*

Proof sketch. Soundness ("only if") is clear (see Appendix B). Completeness ("if") of the propositional variant depends on W being finite. It proceeds via maximally consistent clause sets (MCS) and a Hintikka lemma stating in particular that an MCS containing $(A \cup B)(y)$ must also contain one of $A(y), B(y)$. Completeness of the first-order variant is by adaptation of the completeness proof for standard first-order resolution, going via Herbrand models (i.e. models having the set of ground terms as the carrier set) and reduction to completeness of set-valued propositional resolution. ◀

Of course, the Herbrand models constructed in the proof of Theorem 4.1 are in general infinite. For purposes of constructing finite models, we identify a property of "sufficient completeness" of a model for a set of terms.

▶ **Definition 4.2.** A set $\mathcal{T}$ of (tuples of) terms is *closed under unification* if whenever $t, s \in \mathcal{T}$ are unifiable and σ is an mgu of t, s, then $u\sigma \in \mathcal{T}$ for every $u \in \mathcal{T}$.

▶ Remark 4.3. If $\mathcal{T}$ is closed under unification, then $\mathcal{T}$ is in particular closed under injective renaming of variables: For $u \in \mathcal{T}$, every injective renaming σ is an mgu of u, u, so that $u\sigma \in \mathcal{T}$.

We will treat tuples of terms like terms in the following, in particular mentioning equations between tuples of terms and unifiers of such equations; this is to be understood via componentwise equality in the evident sense.

▶ **Definition 4.4.** A *solution* of an equation $t = s$ in an outcome model $\mathcal{G}$ is a valuation η such that $[\![t]\!]\eta = [\![s]\!]\eta$ in $\mathcal{G}$. Let $\mathcal{T}$ be a set of tuples of terms. We say that $\mathcal{G}$ is *$\mathcal{T}$-equationally complete* if whenever an equation $\bar{t} = \bar{s}$ with $\bar{t}, \bar{s} \in \mathcal{T}$ has a solution in $\mathcal{G}$, then $\bar{t}, \bar{s}$ are unifiable, and the mgu σ of $\bar{t}, \bar{s}$ is a *most general solution* of $\bar{t} = \bar{s}$ in $\mathcal{G}$, i.e. every solution η of $\bar{t} = \bar{s}$ in $\mathcal{G}$ has the form $\eta(x) = [\![\sigma(x)]\!]\eta'$ for some valuation η'; we then say briefly that η *factorizes through* σ.

▶ **Theorem 4.5.** *Let $\mathcal{T}$ be a set of tuples of terms that is closed under unification, and let $\mathcal{G}$ be $\mathcal{T}$-equationally complete. Let ϕ be a clause set such that $\bar{t} \in \mathcal{T}$ for every literal $B(\bar{t})$ occurring in ϕ. If ϕ is consistent under set-valued first-order resolution, then ϕ is satisfiable over $\mathcal{G}$.*

Proof. By completeness of set-valued propositional resolution (Theorem 4.1), it suffices to show that the clause set $\phi^{\mathcal{G}}$ consisting of all instances over $\mathcal{G}$ of clauses in ϕ is consistent under set-valued propositional resolution. Formally, an instance $[\![\Gamma]\!]\eta$ over $\mathcal{G}$ of a clause Γ is induced by an A-valuation η, and given as

$$[\![\Gamma]\!]\eta = \{B([\![\bar{t}]\!]\eta) \mid B(\bar{t}) \in \Gamma\}.$$

Since $\mathcal{T}$ is closed under unification, we can assume w.l.o.g. that ϕ is closed under set-valued first-order resolution (since all terms that appear when closing ϕ under resolution remain in $\mathcal{T}$); then it suffices to show that $\phi^{\mathcal{G}}$ is closed under set-valued propositional resolution, since ϕ and, hence, $\phi^{\mathcal{G}}$ do not contain blatantly inconsistent clauses.

So let $\Gamma, A(\bar{t})$ and $B(\bar{s}), \Delta$ be clauses in ϕ, with variables made disjoint. By the latter restriction, resolvable instances of these clauses in $\mathcal{G}$ can be assumed to use the same valuation; so let η be a valuation such that $[\![\bar{t}]\!]\eta = [\![\bar{s}]\!]\eta$. Then in particular $\bar{t} = \bar{s}$ is solvable in $\mathcal{G}$. Since

$\bar{t}, \bar{s} \in \mathcal{T}$, it follows by $\mathcal{T}$-equational completeness of $\mathcal{G}$ that $\bar{t}, \bar{s}$ are unifiable, hence have an mgu σ, and that σ is a most general solution of $\bar{t} = \bar{s}$ in $\mathcal{G}$. This implies that η has the form $\eta(x) = [\![\sigma(x)]\!]\eta'$ for some A-valuation η'. Thus, the resolvent $[\![\Gamma, (A \cap B)(\bar{t}), \Delta]\!]\eta$ of the two instances has the form $[\![\Gamma\sigma, (A \cap B)(\bar{t}\sigma), \Delta\sigma]\!]\eta'$, and hence is in $\phi^{\mathcal{G}}$ as required since $\Gamma\sigma, (A \cap B)(\bar{t}\sigma), \Delta\sigma$ is in ϕ by closure of ϕ under resolution. ◀

5 The AMC, Coalgebraically

To illustrate the use of one-step tableau rules, we briefly indicate how to obtain the ExpTime upper bound for the AMC by Theorem 3.3. The requisite functor G and the associated predicate liftings have been recalled in Section 3. We recall the known rule set [22, 4]:

$$(CD)\ \frac{[D_1]\,a_1, \dots, [D_\alpha]\,a_\alpha}{a_1, \dots, a_\alpha}$$

$$(C)\ \frac{[D_1]\,a_1, \dots, [D_\alpha]\,a_\alpha, \langle E\rangle\,b, \langle\Sigma\rangle\,c_1, \dots, \langle\Sigma\rangle\,c_\beta}{a_1, \dots, a_\alpha, b, c_1, \dots, c_\beta}$$

where for each j, k, $D_j \cap D_k = \emptyset$ and $D_j \subseteq E$. Soundness of these rules is straightforward (they say in particular that disjoint coalitions can combine their abilities and that coalitions inherit the abilities of subcoalitions); for illustration, we show one-step tableau completeness using set-valued resolution (Section 4), alternative to proofs in the literature [27, 8, 21].

▶ **Theorem 5.1** (One-step tableau completeness). *The rules (C), (CD) are one-step tableau complete w.r.t. AMC.*

By Theorem 3.3, this implies the known (tight) ExpTime upper bound for satisfiability checking in the AMC [21].

Proof. As indicated above, we present a proof producing infinite sets of moves in one-step games, and then discuss how finiteness of move sets is regained using the notion of $\mathcal{T}$-equationally complete (finite) model (Theorem 4.5).

Let τ be a $\mathcal{P}W$-valuation, and let $\Xi = \{[D_1]\,a_1, \dots, [D_\alpha]\,a_\alpha, \langle C_1\rangle\,c_1, \dots, \langle C_\beta\rangle\,c_\beta\}$ such that for every instance of (C) or (CD) that applies to (some subset of) Ξ, the conclusion Θ satisfies $[\![\Theta]\!]\tau \neq \emptyset$. We have to show that $[\![\Xi]\!]\tau \neq \emptyset$. To this end, we translate Ξ into a clause set ϕ in set-valued first-order logic (Section 4), generating one (singleton) clause for each modalized atom $[D_j]\,a_j$ and $\langle C_j\rangle\,c_j$ according to (1) and (2) (Section 4), with distinct Skolem constants $e^j_{D_j}$ and Skolem functions $g^j_{\overline{C_j}}$, respectively. By Theorem 4.1, it suffices to show that ϕ is consistent under set-valued resolution. We observe the following.

1. Two clauses b_j and b_k of shape (1), for $j \neq k$, resolve only if $D_j \cap D_k = \emptyset$ – otherwise, unification fails due to a clash between e^j_i and e^k_i for each agent $i \in D_j \cap D_k$.
2. Similarly, a clause b_j of shape (1) resolves with a clause d_k of shape (2) only if $D_j \cap \overline{C}_k = \emptyset$, i.e. $D_j \subseteq C_k$.
3. Similarly, two clauses d_j and d_k of shape (2), for $k \neq j$, resolve only if $\overline{C_j} \cap \overline{C_k} = \emptyset$, i.e. $C_j \cup C_k = \Sigma$
4. Crucially, two clauses d_j and d_k of shape (2) , for $k \neq j$, resolve only if at least one of C_j and C_k is Σ: Assume that $p \in \overline{C_j}$ and $q \in \overline{C_k}$. By the previous item, $p \in C_k$ and $q \in C_j$, so x_p is an argument in g^k_q and x'_q (renamed for purposes of the resolution step) is an argument in g^j_p, implying that unification of d_j and d_k fails at the occurs check (cf. p. 4). This explains why only one $\langle E\rangle$ with $E \neq N$ is needed in rule (C).

These observations imply that a resolution proof of a blatantly inconsistent (necessarily singleton) clause from ϕ will witness a rule match of either (C) or (CD) (depending on whether clauses of shape (2) are involved), and blatant inconsistency means that $[\![\Theta]\!]\tau = \emptyset$ for the corresponding rule conclusion Θ, contradicting the assumption on Ξ.

Finitely many moves. As indicated in Section 4, the model of Ξ thus produced will have infinitely many moves per agent, namely the ground terms generated by the Skolem constants and functions. We can replace these with finitely many moves where agents play *Skolem symbols* paired with *colours* – simulating the effect of the occurs check from the unification procedure – taken from a finite abelian group U (with neutral element 0 and group operation $+$) that contains distinct elements $u_1, \dots, u_\beta$ (e.g. $U = \mathbb{Z}/\beta\mathbb{Z}$). Specifically, all agents receive (for simplicity) the same moves, namely

- moves $(e^j, 0)$ for $j = 1, \dots, \alpha$, intended as witnesses for $[D_j]\, a_j$, and
- moves (g^j, u) for $j = 1, \dots, \beta$ and $u \in U$, intended as witnesses for $\langle C_j \rangle\, c_j$.

We refer to the first component of a move as its *move symbol*, and to the second as its *colour*. By $\mathsf{col}(m_C)$ we denote the sum of all colours of the moves in a joint move m_C for C.

Let $\mathcal{T}$ be the unification closure of the set of all tuples of argument terms occuring in clauses from ϕ. By the above analysis, all tuples in $\mathcal{T}$ essentially have the shape $(x_A, e_B, g_{\overline{A \cup B}}(x_A, e_B))$ where x_A are variables, e_B are Skolem constants possibly from different box modalities, and $g_{\overline{A \cup B}}$ are Skolem functions from a single diamond (as Skolem functions for different diamonds do not initially occur in the same tuple of terms and such occurrences are not introduced during unification due to the occurs check); any one of x_A, e_B, g may be absent. The (finite) model $\mathcal{G}$ is then defined over coloured moves. Skolem constants e^j are interpreted as $(e^j, 0)$, and Skolem functions g^j_i for $i \in \overline{C_j}$ are interpreted as mapping a joint move m_{C_j} of C_j to $(g^j, u_j - \mathsf{col}(m_{C_j}))$ if i is the least element of $\overline{C_j}$, and to $(g^j, 0)$ otherwise, thus ensuring that $\mathsf{col}(m_{C_j}, g^j(m_{C_j})) = u_j$. We proceed to show that $\mathcal{G}$ is $\mathcal{T}$-equationally complete, obtaining by Theorem 4.5 and consistency of ϕ under set-valued first-order resolution that ϕ is satisfiable over $\mathcal{G}$.

So let $t, u \in \mathcal{T}$ such that $t = u$ has a solution η in $\mathcal{G}$. We proceed by case distinction on the shape of $t = u$:

$(x_A, e_B) = (x'_{A'}, e'_{B'})$: In the simplest case the terms just consist of variables (x_A, $x'_{A'}$) and Skolem constants (e_B, $e'_{B'}$). Given the interpretation of the Skolem constants in $\mathcal{G}$, it is clear that e_B and e'_B must agree on $B \cap B'$ so t, u are unifiable. The solution η necessarily replaces variables in $A \cap B'$ and $A' \cap B$ with the respective interpretations of Skolem constants on the other side of the equality. Hence, the solution η factorizes through the mgu of t and u.

$(x_A, e_B, g^j_{\overline{A \cup B}}(x'_A, e'_B)) = (x_{A'}, e_{B'})$: This case is similar to the previous one, using the observation that given the interpretation of g^j in $\mathcal{G}$, the equation can only have a solution if $(\overline{A \cup B}) \cap B' = \emptyset$, i.e. $(\overline{A \cup B}) \subseteq A'$.

$(x_A, e_B, g^j_{\overline{A \cup B}}(x_A, e_B)) = (x'_{A'}, e'_{B'}, g^k_{\overline{A' \cup B'}}(x'_{A'}, e'_{B'}))$: The interpretations of the terms $g^j_{\overline{A \cup B}}(x_A, e_B)$ and $g^k_{\overline{A' \cup B'}}(x_{A'}, e'_{B'}))$ in $\mathcal{G}$ (under η) have the form (g^j, c) and (g^k, d) for some c and d, respectively. The case where $j = k$ is essentially like the previous cases. The interesting case is where $j \neq k$, in which case necessarily $\overline{A \cup B} \subseteq A'$ and $\overline{A' \cup B'} \subseteq A$; this is the case where unification of t, u fails at the occurs check as explained above. However, the construction of $\mathcal{G}$ ensures that now $t = u$ also has no solution in $\mathcal{G}$, as the respective interpretations of g^j and g^k ensure that the colour of the whole joint move is u_j on the left and u_k on the right. ◀

The proof for the AMCDES proceeds in a quite similar fashion, and will be presented in less detail.

6 AMCDES Satisfiability

We now extend this treatment to obtain ExpTime satisfiability checking for AMCDES, cast coalgebraically using the functor and predicate liftings presented in Section 3. We have one-step rules (DES_0), (DES_1), where (DES_1) is

$$(DES_1)\ \frac{[D_1, P_{G_1}]\, a_1, \dots, [D_\alpha, P_{G_\alpha}]\, a_\alpha, \langle E, Q_K\rangle\, b, \langle C_1, r_{H_1}\rangle\, c_1, \dots, \langle C_\beta, r_{H_\beta}\rangle\, c_\beta}{(a_j)_{j\in I_q}, b, (c_j)_{j\in J_q} \mid \cdots \text{ for } q \in Q_K}$$

(i.e. the rule has one conclusion for each q) where $Ag(Q_K) = K$; the r_{H_j} are (non-disjunctive) explicit joint moves for coalitions H_j; $I_q \subseteq \{1, \dots, \alpha\}$, $J_q \subseteq \{1, \dots, \beta\}$ for each $q \in Q_K$; and the following side conditions hold, with $L := \bigcup_{j=1}^{\alpha} G_j \cup \bigcup_{j=1}^{\beta} H_j$:

1. For each j, k, $D_j \cap D_k = \emptyset$.
2. For each j, $C_j \cup H_j = \Sigma$.
3. $\bigcup_{j=1}^{\alpha} D_j \cap L = \emptyset$.
4. $\bigcup_{j=1}^{\alpha} D_j \subseteq E$.
5. $E \cup K \supseteq L$.
6. $r_{H_j} =_\sqcap q$ for all $q \in Q_K$, $j \in J_q$.
7. There is a joint explicit move l for $E \cap L$ such that $r_{H_j} =_\sqcap l$ for each $q \in Q_K$, $j \in J_q$, and moreover for each $j \in I_q$ there exists $p \in P_{G_j}$ such that $p =_\sqcap q$ and $p =_\sqcap l$.

Rule (DES_0) is a variant of (DES_1) obtained by instantiating to $\langle E, Q_K\rangle\, b = \langle \Sigma, \{()\}\rangle\, \top$, $I_{()} = \{1, \dots, \alpha\}$, and $J_{()} = \{1, \dots, \beta\}$, and then omitting the (valid) literal $\langle \Sigma, \{()\}\rangle\, \top$ from the rule premiss; side conditions 4.–6. then become trivial and can be omitted.

Rule (DES_1) extends the rules for the basic AMC as recalled in Section 5. The new features are intuitively understood as follows. Imagine that $D_1, \dots, D_n$ play moves witnessing their ability to (conditionally) enforce $a_1, \dots, a_n$. According to $\langle E, Q_K\rangle$, K can then play some move $q \in Q_K$ additionally ensuring b; the q-th conclusion of (DES_1) captures the constraints on the next state reached in this situation. These additionally depend on the moves chosen by the remaining agents (those in $E \setminus \bigcup D_i$): If the arising joint move restricts to one of the moves in P_{G_j}, then D_j successfully enforces a_j, and if it restricts to r_{H_j}, then the next state must satisfy c_j (note that since $C_j \cup H_j = \Sigma$, $\langle C_i, r_{H_k}\rangle\, c_j$ says that c_j is enforced as soon as H_j play r_{H_j}). The index sets I_q and J_q indicate for which j this applies, and side conditions 6 and 7 ensure that a corresponding joint move actually exists. For definiteness, we note

▶ **Lemma 6.1** (One-step soundness). *The rules (DES_0), (DES_1) are one-step tableau sound w.r.t. AMCDES.*

Proof. By the above, it suffices to show soundness of (DES_1), formalizing the above intuitive explanation. Write ϕ for the premiss of the rule, and ψ_q for the conclusion associated to $q \in Q_K$. Let τ be a $\mathcal{P}W$-valuation such that $[\![\phi]\!]\tau \neq \emptyset$, and fix $\mathcal{G} = ((k_j), f, \iota) \in [\![\phi]\!]\tau$; we have to show that $[\![\psi_q]\!]\tau \neq \emptyset$ for some $q \in Q_K$. We refer to side conditions by their numbers:

- For each $j \in \{1, \dots, \alpha\}$, we have a joint move e_j for D_j witnessing $[D_j, P_{G_j}]\, a_j$. By 1., the e_j can be combined into a joint move e for $\bigcup_{j=1}^{\alpha} D_j$.

- By 3., e can be combined with (the interpretation of) the explicit move l postulated in 7. into a move x_0 for $(E \cap L) \cup \bigcup_{j=1}^{\alpha} D_j \subseteq E$, where the inclusion is by 4. Extend x_0 arbitrarily to a move x for the whole coalition E.
- Since $\mathcal{G} \in [\![\langle E, Q_K \rangle\, b]\!]$ and $Ag(x) = E$, there is some $q \in Q_K$ and a joint move m_q for Σ such that $x, q \sqsubseteq m_q$ and $f(m_q) \in \tau(b)$.
- To obtain that $f(m_q) \in [\![\psi_q]\!]\tau$ for this q, it remains to show that $f(m_q)$ satisfies the remaining literals a_j, c_j of ψ_q:
  - For $j \in I_q$, we have $e_j \sqsubseteq m_q$ and, by 5. and 7., $\iota[p] \sqsubseteq m_q$ for some $p \in P_{G_j}$, so that $\mathcal{G} \in [\![[D_j, P_{G_j}]\, a_j]\!]\tau$ implies $f(m_q) \in \tau(a_j)$.
  - For $j \in J_q$, we have $\iota[r_{H_j}] \sqsubseteq m_q$ by 5., 6., and 7. Since $C_j \cup H_j = \Sigma$, we thus have that $\mathcal{G} \in [\![\langle C_k, r_{H_k} \rangle\, c_k]\!]\tau$ implies $f(m_j) \in \tau(c_k)$. ◀

It remains to prove ompleteness:

▶ **Lemma 6.2** (One-step tableau completeness). *The rules (DES_0), (DES_1) are one-step tableau complete w.r.t. AMCDES.*

Proof. Let τ be a $\mathcal{PW}$-valuation, and let $\Xi = \{[D_1, P_{G_1}]\, a_1, \ldots, [D_\alpha, P_{G_\alpha}]\, a_\alpha, \langle C_1, R_{H_1} \rangle\, c_1, \ldots, \langle C_\beta, R_{H_\beta} \rangle\, c_\beta\}$ such that every instance of (DES_0) or (DES_1) whose premise is contained in Ξ has a conclusion that is non-empty under τ. We have to show that $[\![\Xi]\!]\tau \neq \emptyset$. We translate Ξ into a clause set ϕ in set-valued first-order logic by including for each $[D_j, P_{G_j}]\, a_j$ and each $p \in P_{G_j}$ a singleton clause

$$\{\tau(a_j)(e^j_{D_j}, x_{\overline{D_j \cup G_j}}, p)\}, \tag{3}$$

(so $e^j_{D_j}$ witnesses $[D_j, P_{G_j}]\, a_j$), and for each $\langle C_j, R_{H_j} \rangle\, c_j$ a clause

$$\{\tau(c_j)(x_{C_j}, g^j_{\overline{C_j \cup H_j}}(x_{C_j}), r) \mid r \in R_{H_j}\} \tag{4}$$

(so the $g^j_{\overline{C_j \cup H_j}}$ are Skolem functions witnessing $\langle C_j, R_{H_j} \rangle\, c_j$). We now proceed as in the proof of Theorem 5.1: We first show that ϕ is consistent under set-valued resolution, obtaining by Theorem 4.1 that ϕ is satisfiable in a model that may have infinitely many moves, and then present a finite $\mathcal{T}$-equationally complete model for the unification closure $\mathcal{T}$ of the involved terms. Write b^p_j for clauses of type (3) for given $j = 1, \ldots, \alpha$ and $p \in G_j$, and d_j for the j-th clause of type (4).

Unlike in the proof of Theorem 5.1, we thus may have non-singleton clauses, of shape (4). However, we shall see that these non-singleton clauses do not resolve among each other. We note the following observations.

1. b^p_j and b^q_j, for $p \neq q$, do not resolve (and resolving b^p_j with itself is pointless).
2. b^p_j and b^q_k, for $k \neq j$, resolve only if $D_j \cap D_k = D_j \cap G_k = D_k \cap G_j = \emptyset$, and moreover $p =_\sqcap q$.
3. b^p_j and d_k resolve, at the d_k-literal for $r \in R_{H_k}$, only if $D_j \subseteq C_k$, and hence in particular also $D_j \cap H_k = \emptyset$, $D_j \cup G_j \subseteq C_k \cup H_k$ (equivalently $\overline{C_k \cup H_k} \subseteq \overline{D_j \cup G_j}$), and $r =_\sqcap p$.
4. d_j and d_k, for $k \neq j$, resolve, at the d_j-literal for $r \in H_j$ and the d_k-literal for $r' \in H_k$, only if $C_j \cup C_k \cup H_k = \Sigma$ (equivalently $\overline{C_k \cup H_k} \subseteq C_j$), $C_k \cup C_j \cup H_j = \Sigma$, and $r =_\sqcap r'$.
5. Like in the proof of Theorem 5.1, it follows that d_j and d_k resolve only if at least one of $\langle C_j, R_{H_j} \rangle$ and $\langle C_k, R_{H_k} \rangle$ is a grand coalition modality (since otherwise unification fails at the occurs check), in which case the corresponding clause is a singleton.
6. Clauses obtained from clauses of shape (4) by resolving with singleton clauses retain essentially shape (4), only with some of the variables x_i replaced with constants. Resolution of such clauses is thus subject to the same restrictions; in particular, non-singleton clause of this kind they will not resolve among each other.

Thus, a proof of a blatantly inconsistent clause from ϕ by set-valued resolution will involve either zero or one clauses d_j where $C_j \cup H_j \neq \Sigma$. We will refer to resolution proofs of the first kind as type-0 and to proofs of the second kind as type-1.

Type-0 proofs. We show that in this case, the impossibility of deriving a blatantly inconsistent clause is obtained via rule (DES_0). To apply (DES_0) to the set of modal atoms involved in the proof, we need to show the side conditions of the rule (1.–3. and 7). Indeed, condition 2. holds by the definition of type-0 proofs. As no disjunctive diamond is involved in a type-0 proof, all involved clauses are singletons. Hence, 1., 3., and 7. directly follow from the observations above. The type-0 proof at hand thus induces a match of rule (DES_0) to a subset of Ξ; the conclusion of this rule match having non-empty extension under τ means precisely that the resolution proof does not produce a blatantly inconsistent clause.

Type-1 proofs. Those consist in successively resolving all literals of a single clause of the form d_{j_0} where $C_{j_0} \cup H_{j_0} \neq \Sigma$ with suitable singleton clauses, of the form either b_j^p or d_k where $C_k \cup H_k = \Sigma$. We will refer to these resolution steps as "resolving into d_{j_0}", although of course d_{j_0} will have been modified by previous resolution steps as described above. To match the notation of rule (DES_1), we rename $\langle C_{j_0}, R_{H_{j_0}} \rangle$ into $\langle E, Q_K \rangle\, b$ (so that all the $\langle C_j, R_{H_j} \rangle\, c_j$ that remain have $C_j \cup H_j = \Sigma$ and hence $|R_{H_j}| = 1$). The literals in d_{j_0} are then indexed over $q \in Q_K$. Let I_q be the set of all j such that for some $p \in P_j$, b_j^p is resolved into d_{j_0} at the literal for q, and put $G = \bigcup_{q \in Q_K, j \in I_q} G_j$; similarly, let J_q be the set of all j such that d_j (a singleton clause) is resolved into d_{j_0} at the literal for q, and put $H = \bigcup_{q \in Q_K, j \in J_q} H_j$. Notice that two clauses resolve only if whenever they both assign a constant (either a Skolem constant or an explicit move) to a certain agent, then the constant is the same in both clauses; this implies condition 7. Conditions 1. and 3. are established as in the type-0 case, condition 2. is ensured by the above renaming, and the remaining conditions follow directly from the above observations. The type-1 proof at hand thus induces a match of rule (DES_1) to a subset of Ξ; a conclusion of this rule match having non-empty extension under τ means precisely that the resolution proof does not produce a blatantly inconsistent clause.

Finitely many moves. As indicated above, we obtain a model with finitely many moves by constructing a finite $\mathcal{T}$-equationally complete model $\mathcal{G}$, where $\mathcal{T}$ is the unification closure of the tuples of terms occurring in ϕ. This construction is essentially the same as for the AMC, up to the presence of additional constant symbols, viz. the explicit strategies occurring in ϕ. These constants can be treated exactly like the Skolem constants already present in the proof of Theorem 5.1. The full proof is available in Appendix B. ◀

Since the rules (DES_1), (DES_0) are algorithmically sufficiently harmless, our main result follows from Lemmas 6.1 and 6.2 by Theorem 3.3:

▶ **Theorem 6.3.** *Satisfiability checking for the AMCDES is ExpTime-complete.*

7 Conclusions

We have introduced the alternating-time μ-calculus with disjunctive explicit strategies (AMCDES), which extends ATL with explicit strategies (ATLES) [29] with fixpoint operators and disjunction over explicit strategies of opposing agents in non-grand modalities. We have employed methods from coalgebraic logic to show that model checking with fixed interpretation of explicit strategies is in QP as well as in NP ∩ coNP, and in NP with open interpretation of strategies, and moreover that satisfiability checking is in ExpTime.

The coalgebraic treatment in fact implies a whole range of additional results, e.g. reasoning in the next-step fragment of the logic extended with nominals (ExpTime with global axioms, and PSpace without) [24, 17, 9]; cut-free sequent systems for the next-step fragment [19]; and completeness of a Kozen-Park axiomatization for flat (i.e. single-variable) fragments of the AMCDES, e.g. ATL with disjunctive explicit strategies [25]. A special case of the latter result is completeness of ATLES as proved already in Walther et al. [29].

In ongoing work we are extending our axiomatization and complexity results to allow strategy disjunction also in grand coalition modalities. A natural but more challenging further extension would be to add negative strategies prohibiting moves for some agents as suggested by Herzig et al. [12].

References

1 Rajeev Alur, Thomas Henzinger, and Orna Kupferman. Alternating-time temporal logic. *J. ACM*, 49:672–713, 2002. doi:10.1145/585265.585270.

2 Cristian Calude, Sanjay Jain, Bakhadyr Khoussainov, Wei Li, and Frank Stephan. Deciding parity games in quasipolynomial time. In *Theory of Computing, STOC 2017*, pages 252–263. ACM, 2017.

3 Krishnendu Chatterjee, Thomas A. Henzinger, and Nir Piterman. Strategy logic. *Inf. Comput.*, 208(6):677–693, 2010. doi:10.1016/j.ic.2009.07.004.

4 Corina Cîrstea, Clemens Kupke, and Dirk Pattinson. EXPTIME tableaux for the coalgebraic μ-calculus. *Log. Methods Comput. Sci.*, 7(3), 2011. doi:10.2168/LMCS-7(3:3)2011.

5 Corina Cîrstea, Alexander Kurz, Dirk Pattinson, Lutz Schröder, and Yde Venema. Modal logics are coalgebraic. *Comput. J.*, 54(1):31–41, 2011. doi:10.1093/comjnl/bxp004.

6 Melvin Fitting. *First-Order Logic and Automated Theorem Proving*. Springer, 2nd edition, 1996. doi:10.1007/978-1-4612-2360-3.

7 Valentin Goranko and Fengkui Ju. Towards a logic for conditional local strategic reasoning. In Patrick Blackburn, Emiliano Lorini, and Meiyun Guo, editors, *Logic, Rationality, and Interaction, LORI 2019*, volume 11813 of *LNCS*, pages 112–125. Springer, October 2019. doi:10.1007/978-3-662-60292-8_9.

8 Valentin Goranko and Govert van Drimmelen. Complete axiomatization and decidability of alternating-time temporal logic. *Theor. Comput. Sci.*, 353(1-3):93—117, March 2006. doi:10.1016/j.tcs.2005.07.043.

9 Rajeev Goré, Clemens Kupke, Dirk Pattinson, and Lutz Schröder. Global caching for coalgebraic description logics. In *Automated Reasoning, IJCAR 2010*, volume 6173 of *LNCS*, pages 46–60. Springer, 2010. doi:10.1007/978-3-642-14203-1.

10 Erich Grädel, Wolfgang Thomas, and Thomas Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research*, volume 2500 of *LNCS*. Springer, 2002. doi:10.1007/3-540-36387-4.

11 Daniel Hausmann and Lutz Schröder. Game-based local model checking for the coalgebraic mu-calculus. In *Concurrency Theory, CONCUR 2019*, volume 140 of *LIPIcs*, pages 35:1–35:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, August 2019. doi:10.4230/LIPIcs.CONCUR.2019.35.

12 Andreas Herzig, Emiliano Lorini, and Dirk Walther. Reasoning about actions meets strategic logics. In *Logic, Rationality, and Interaction, LORI 2013*, volume 8196 of *LNCS*, pages 162–175. Springer, 2013. doi:10.1007/978-3-642-40948-6_13.

13 Marc-Uwe Kling. *Das Känguru-Manifest.* Ullstein, Berlin, 2011.

14 Marc-Uwe Kling. *The Kangaroo Chronicles.* Voland & Quist, 2016. Translated by Sarah Cossaboon and Paul-Henri Campbell.

15 Fabio Mogavero, Aniello Murano, Giuseppe Perelli, and Moshe Vardi. Reasoning about strategies: On the model-checking problem. *ACM Trans. Comput. Log.*, 15(4):34:1–34:47, 2014. doi:10.1145/2631917.

16 Fabio Mogavero, Aniello Murano, Giuseppe Perelli, and Moshe Vardi. Reasoning about strategies: on the satisfiability problem. *LMCS*, 13(1), 2017. doi:10.23638/LMCS-13(1:9)2017.

17 Robert Myers, Dirk Pattinson, and Lutz Schröder. Coalgebraic hybrid logic. In *Foundations of Software Science and Computational Structures, FOSSACS 2009*, volume 5504 of *LNCS*, pages 137–151. Springer, 2009. doi:10.1007/978-3-642-00596-1.

18 Rocco De Nicola and Frits W. Vaandrager. Action versus state based logics for transition systems. In Irène Guessarian, editor, *Semantics of Systems of Concurrent Processes, LITP Spring School on Theoretical Computer Science 1990*, volume 469 of *LNCS*, pages 407–419. Springer, April 1990. doi:10.1007/3-540-53479-2_17.

19 Dirk Pattinson and Lutz Schröder. Cut elimination in coalgebraic logics. *Inf. Comput.*, 208(12):1447–1468, 2010. doi:10.1016/j.ic.2009.11.008.

20 Marc Pauly. A modal logic for coalitional power in games. *J. Log. Comput.*, 12(1):149–166, 2002. doi:10.1093/logcom/12.1.149.

21 Sven Schewe. *Synthesis of Distributed Systems*. PhD thesis, Universität des Saarlandes, 2008.

22 Lutz Schröder and Dirk Pattinson. PSPACE bounds for rank-1 modal logics. *ACM Trans. Comput. Log.*, 10(2):13:1–13:33, 2009. doi:10.1145/1462179.1462185.

23 Lutz Schröder and Dirk Pattinson. Modular algorithms for heterogeneous modal logics via multi-sorted coalgebra. *Math. Struct. Comput. Sci.*, 21(2):235–266, 2011. doi:10.1017/S0960129510000563.

24 Lutz Schröder, Dirk Pattinson, and Clemens Kupke. Nominals for everyone. In *International Joint Conference on Artificial Intelligence, IJCAI 2009*, pages 917–922, July 2009. URL: http://ijcai.org/proceedings/2009.

25 Lutz Schröder and Yde Venema. Completeness of flat coalgebraic fixpoint logics. *ACM Trans. Comput. Log.*, 19(1):4:1–4:34, 2018. doi:10.1145/3157055.

26 Wiebe van der Hoek, Wojciech Jamroga, and Michael Wooldridge. A logic for strategic reasoning. In *Autonomous Agents and Multiagent Systems, AAMAS 2005*, pages 157–164. ACM, 2005. doi:10.1145/1082473.1082497.

27 Govert van Drimmelen. Satisfiability in alternating-time temporal logic. In *Logic in Computer Science, LICS 2003*, pages 208–217. IEEE Comp. Soc., June 2003. doi:10.1109/LICS.2003.1210060.

28 Dirk Walther. *Strategic Logics: Complexity, Completeness and Expressivity*. PhD thesis, University of Liverpool, 2007.

29 Dirk Walther, Wiebe van der Hoek, and Michael Wooldridge. Alternating-time temporal logic with explicit strategies. In *Theoretical Aspects of Rationality and Knowledge, TARK 2007*, pages 269–278. ACM Press, 2007. doi:10.1145/1324249.1324285.

30 Thomas Ågotnes. Action and knowledge in alternating-time temporal logic. *Synthese*, 149(2):375–407, March 2006. doi:10.1007/s11229-005-3875-8.

A Appendix: AMCDES Model Checking Details

Summary of Results on Coalgebraic Model Checking

Given a functor F, we assume a representation of the elements of FX, for finite X, as strings over some alphabet. Specifically, we represent elements of $((k_j)_{j\in\Sigma}, f) \in \mathsf{G}X$ as tabulations of f.

Model checking results [11] for the full coalgebraic μ-calculus require only very simple properties of the predicate liftings:

▶ **Definition A.1.** The *one-step satisfaction problem* is to determine, given a finite set X, $Y \subseteq X$, $\heartsuit \in \Lambda$, and $t \in FX$, whether $t \in [\![\heartsuit]\!]_X(Y)$.

▶ **Theorem A.2** (Model checking via one-step satisfaction [11, Theorem 11])**.** *If the one-step satisfaction problem is in P, then the model checking problem for the coalgebraic μ-calculus over this logic is in NP ∩ coNP.*

The proof of this upper bound is via parity games, specifically by noting that Cîrstea et al.'s *evaluation games* [4] are exponentially large but have only polynomially many Eloise-nodes, so that winning strategies for Eloise can be guessed and verified in (nondeterministic) polynomial time.

On the other hand, to obtain a model checking algorithm in QP (deterministic quasipolynomial time $2^{\mathcal{O}((\log n)^k)}$ for some k; a complexity class not currently known to be comparable with NP) we need to show that we can design suitable one-step satisfaction arenas for use in model checking games (we use standard terminology for games, e.g. [10]):

▶ **Definition A.3.** A *one-step satisfaction arena* A for a set X, a modality $\heartsuit \in \Lambda$, and $t \in FX$ is an acyclic arena for games with two players Eloise and Abelard (recall that an arena is like a game in that it specifies nodes, each assigned to one of the players, and allowed moves between nodes but does not include a winning condition; acyclicity refers to the move relation), with a single initial node, with X as the set of terminal nodes, and with additional inner nodes. A *one-step game* on A additionally specifies a winning condition in the shape of a subset Y of the terminal nodes; then, Eloise wins plays that either get stuck at an inner Abelard node without successors or terminate in a node in Y. We say that A is *sound and complete* if for every $Y \subseteq X$, Eloise wins (the initial node of) the one-step game on A with winning condition Y iff $t \in [\![\heartsuit]\!]_X(Y)$.

▶ **Theorem A.4** (Model checking via one-step games [11, Corollary 18])**.** *If for every set X, $\heartsuit \in \Lambda$, and $t \in FX$, there is a sound and complete one-step satisfaction arena with polynomially many inner nodes in the representation size of $\heartsuit$ and t, then the model checking problem for the μ-calculus over this logic is in QP.*

The model checking procedure underlying this theorem is to construct a polynomial-size model checking parity game using one-step games as building blocks; by well-known recent advances in parity game solving [2], these games can be solved in quasipolynomial time.

Proof of Theorem 2.9

Proof. The one-step satisfaction problem for the AMCDES is to check whether $((k_j), f, \iota) \in [\![[C,O]]\!]_X(Y)$ can be decided in P for given C, O, $Y \subseteq X$, and a one-step game with explicit strategies $((k_j), f, \iota) \in \mathsf{G}_{\mathsf{ES}}X$. This can be done by iterating over joint moves of C in an outer loop and over joint moves of $\overline{C}$ in an inner loop. Since f needs to tabulate the outcomes of all joint moves of Σ, both loops have at most linearly many (in the size of f) iterations per invocation, making for a quadratic overall number of iterations of the inner loop, and hence polynomial run time.

Algorithm 1 One-step Satisfaction Algorithm.

```
for m_C ← [k_C] do
    x := ⊤
    for o ← O, m_C̄ ← [k_{Σ\C\Ag(O)}] do
        if f(m_c, m_C̄, ι[o]) ∉ Y then x := ⊥
    if x then return ⊤
return ⊥
```

By Theorem A.2, we thus obtain the NP $\cap$ coNP bound for the fixed case. The NP bound for the open case follows by guessing history-free strategies.

For the QP bound, we use Theorem A.4 and adapt the one-step satisfaction arenas for the AMC [11, Example 15.5] to obtain small one-step satisfaction arenas for the AMCDES:

The one-step satisfaction arena $A_{[C,O],w} = (V_{[C,O],w}, E_{[C,O],w})$ for X, $[C, O]$, and a one-step game $((k_j), f, \iota) \in \mathsf{G}_{\mathsf{ES}}X$ for disjoint $C, D \subseteq \Sigma$, $O \subseteq \prod_{a \in D} M_a$ is constructed as follows. The node set $V_{[C,O],w}$ consists of an initial node $([C, O], w)$ belonging to Eloise, and additionally a set of inner nodes $I_{[C,O],w} := [k_C]$ belonging to Abelard i.e. one node for each joint move of C. The set $E_{[C,O],w}(x)$ of moves available at a node x is

$$E_{[C,O],w}(x) = \begin{cases} I_{[C,O],w} \text{ if } x = ([C, O], w) \\ \{f(x, m_{\bar{C}}, o) \mid m_{\bar{C}} \in [k_{\overline{C \cup D}}], o \in \iota[O]\} \end{cases}$$

It is easy to see that the size of the arena is thus linear in the tabulation size of f. The soundness and completeness of the resulting one-step satisfaction game stems from the fact that the moves of Eloise and Abelard essentially construct the witnessing moves from the original game. ◀

B Appendix: Omitted Proofs and Further Details

Proof of Theorem 4.1

Soundness. It suffices to show that the rule (SR) is sound. Let $\Gamma, A(\bar{t})$ and $B(\bar{u}), \Delta$ be two clauses such that $\bar{t}$ and $\bar{u}$ are unifiable, and let $\sigma = mgu(\bar{t}, \bar{u})$. Let $\mathcal{G} = ((S_j)_{j \in N}, f, W, [\![-]\!])$ be an outcome model satisfying both $\Gamma, A(\bar{t})$ and $B(\bar{u}), \Delta$. Let η be a valuation such that $\mathcal{G}, \eta \not\models \Gamma\sigma, \Delta\sigma$; we have to show $\mathcal{G}, \eta \models (A \cap B)(\bar{t}\sigma)$. By the evident substitution lemma, $\mathcal{G}, \eta_\sigma \not\models \Gamma, \Delta$ where $\eta_\sigma(x) = [\![\sigma(x)]\!]\eta$ for all x; hence necessarily $\mathcal{G}, \eta_\sigma \models A(\bar{t})$ and $\mathcal{G}, \eta_\sigma \models B(\bar{u})$. Again by the substitution lemma, $\mathcal{G}, \eta \models A(\bar{t}\sigma)$ and $\mathcal{G}, \eta \models B(\bar{u}\sigma)$. Since $\bar{t}\sigma = \bar{u}\sigma$, our goal $\mathcal{G}, \eta \models (A \cap B)(\bar{t}\sigma)$ follows by the semantics of literals.

Completeness. The completeness proof for the propositional variant proceeds via *maximally consistent clause sets*, defined in the expected way. By Zorn's lemma, we have

▶ **Lemma B.1** (Lindenbaum lemma for set-valued propositional resolution). *Every consistent clause set in set-valued propositional logic is contained in a maximally consistent set.*

Moreover, we have the following set of Hintikka properties:

▶ **Lemma B.2** (Hintikka lemma for set-valued propositional resolution). *Let ϕ be a maximally consistent clause set in set-valued propositional logic. Then*

1. *A clause Γ, Δ is in ϕ iff $\Gamma \in \phi$ or $\Delta \in \phi$.*
2. *A clause $\Gamma, (A \cup B)(y)$ is in ϕ iff one of $\Gamma, A(y)$ and $\Gamma, B(y)$ is in ϕ.*
3. *For every $y \in Y$, $W(y) \in \phi$.*

Proof. *1, "if":* Assume w.l.o.g. that $\Gamma \in \phi$. By maximality, it suffices to show that $\phi \cup \{\Gamma, \Delta\}$ remains consistent. So assume that a blatantly inconsistent clause can be derived from $\phi \cup \{\Gamma, \Delta\}$. Then by removing literals from the clauses in this derivation, we obtain a derivation of a blatantly inconsistent clause from $\phi \cup \{\Gamma\}$, contradiction.

1, "only if": By maximality, it suffices to show that one of $\phi \cup \{\Gamma\}$ and $\phi \cup \{\Delta\}$ is consistent. Assume the contrary. Then one can derive a blatantly inconsistent clause Γ' from $\phi \cup \{\Gamma\}$. Adding Δ to all clauses in the derivation (that is, to the original Γ and then to all clauses newly produced by the resolution rule), we obtain a derivation of Γ', Δ from $\phi \cup \{\Gamma, \Delta\}$. Similarly, we have a derivation of a blatantly inconsistent clause Δ' from $\phi \cup \{\Delta\}$, from which we obtain a derivation of Γ', Δ' from $\phi \cup \{\Gamma', \Delta\}$. Chaining the two derivations, we obtain a derivation of the blatantly inconsistent clause Γ', Δ' from $\phi \cup \{\Gamma, \Delta\}$, contradiction.

2, "if": Assume w.l.o.g. that $\Gamma, A(y)$ is in ϕ. By maximality, it suffices to show that $\phi \cup \{\Gamma, (A \cup B)(y)\}$ is consistent. Assume the contrary, i.e. we can derive a blatantly inconsistent clause from $\Gamma, (A \cup B)(y)$. Tracing $(A \cup B)(y)$ through the derivation in the obvious sense (with $A \cup B$ possibly transformed into strictly smaller subsets by the resolution rule) and intersecting with A at each occurrence, we obtain a derivation of a blatantly inconsistent clause from $\phi \cup \{\Gamma, A(y)\} = \phi$, contradiction.

2, "only if": By contraposition, again using maximality: assume that both $\phi \cup \{\Gamma, A(y)\}$ and $\phi \cup \{\Gamma, B(y)\}$ are inconsistent; we have to show that $\phi \cup \{\Gamma, (A \cup B)(y)\}$ is inconsistent. By assumption, we can derive from $\phi \cup \{\Gamma, A(y)\}$ a blatantly inconsistent clause, necessarily of the form $\Gamma', \emptyset(y)$ (since no $y \in Y$ can be made to disappear by the resolution rule). Tracing $A(y)$ through the derivation and taking unions with B at each occurrence, we obtain a derivation of $\Gamma', B(y)$ from $\phi \cup \{\Gamma, (A \cup B)(y)\}$. Similarly, we can derive a blatantly inconsistent clause from $\phi \cup \{\Gamma, B(y)\}$. Replacing literals $C(z)$ with $\emptyset(z)$ and adding new literals of the form $\emptyset(z)$, we obtain a derivation of a blatantly inconsistent clause Θ from $\phi \cup \{\Gamma', B(y)\}$. Chaining derivations, we obtain a derivation of Θ from $\phi \cup \{(A \cup B)(y)\}$, showing the required inconsistency.

3: Clear. ◀

Now fix a maximally consistent clause set ϕ, and assume that W is finite; we construct a model, i.e. a function $f_\phi : Y \to W$, from ϕ as follows. For $y \in Y$, we have $W(y) \in \phi$ by the Hintikka lemma, and then, again by the Hintikka lemma and by finiteness of W, $\{w_y\}(y) \in \phi$ for some $w_y \in W$, which by consistency of ϕ is moreover unique; we put $f_\phi(y) = w_y$.

▶ **Lemma B.3** (Truth lemma for set-valued propositional resolution). *Given a maximally consistent clause set ϕ in set-valued propositional logic over a finite set W, the function f_ϕ constructed above satisfies ϕ.*

Proof. Induction over the size of clauses Γ, measured as the sum of the cardinalities of the subsets of W occurring in Γ. The inductive step makes a case distinction over whether there is more than one or exactly one literal in Γ (the case of zero literals does not occur, as a clause without literals is blatantly inconsistent), and then proceeds according to the relevant clause of the Hintikka lemma. We are left with the induction base, where Γ has the form $\{w\}(y)$; in this case, the claim holds by construction of f_ϕ. ◀

In combination with Lemma B.1, this proves completeness of the propositional variant. Completeness for the first-order variant is then shown via a form of Herbrand theory. We build a *Herbrand universe* where the moves of each agent i are ground terms of sort i. We denote these sets of moves by S_i. A *ground substitution* replaces variables by ground terms, respecting sorts. *Ground instances* of literals $A(\bar{t})$, clauses, and clause sets are obtained by applying a ground substitution.

Now let ϕ be a clause set in set-valued first-order logic that is closed under set-valued first-order resolution and not blatantly inconsistent; it suffices to show that such ϕ are satisfiable. We denote by $I(\phi)$ the set of ground instances of clauses in ϕ. To show that ϕ is satisfiable over the Herbrand universe, it suffices to establish that $I(\phi)$ is satisfiable. Clearly, $I(\phi)$ is not blatantly inconsistent. We show that it is moreover closed under set-valued propositional resolution (implying that $I(\phi)$ is satisfiable, and hence that ϕ is satisfiable). A pair of resolvable clauses in $I(\phi)$ has the form $\Gamma\theta, A(\bar{t}\theta)$ and $B(\bar{u}\theta), \Delta\theta$ where $\Gamma, A(\bar{t})$ and $B(\bar{u}), \Delta$ are in ϕ, w.l.o.g. with disjoint sets of variables, and θ is a ground substitution such that $\bar{t}\theta = \bar{u}\theta$. In particular, $\bar{t}$ and $\bar{u}$ are unifiable, and thus have a most general unifier σ; by definition of the latter, there exists θ' such that $\theta = \sigma\theta'$. Since ϕ is closed under resolution, it follows that the resolvent $\Gamma\sigma, (A \cap B)(\bar{t}\sigma), \Delta\sigma$ is in ϕ. Applying the ground substitution θ' to this clause, we obtain that the propositional resolvent $\Gamma\theta, (A \cap B)(\bar{t}\theta), \Delta\theta$ is in $I(\phi)$, as required. ◀

Remarks on One-step Tableau Completeness for the AMC (Theorem 5.1)

In the proof of Theorem 5.1, one could equally well have used previous one-step model constructions implicit in van Drimmelen, Goranko, and Schewe [27, 8, 21]; we provide our construction for illustration, in preparation for the treatment of disjunctive explicit strategies, to which, as far as we can see, the previous constructions do not adapt (they do extend to explicit strategies without strategy disjunction). We note that the model construction becomes much simpler if one excludes the grand coalition (as, effectively, in ATLES): In the rule (C), the literals $\langle\Sigma\rangle\, c_j$ disappear; in the proof of one-step tableau completeness of the arising rule, one can just use a single move $\bot$ as witness for all $\langle C_j\rangle\, c_j$ in Ξ (in the notation of the original proof of Theorem 5.1), using non-determinism to ensure satisfaction of the $\langle C_j\rangle\, c_j$. In detail, this is seen as follows.

As indicated above, in the absence of grand coalition modalities, rule (C) specializes to

$$(C^-)\ \frac{[D_1]\, a_1, \dots, [D_\alpha]\, a_\alpha, \langle E\rangle\, b}{a_1, \dots, a_\alpha, b}$$

with the same side conditions as (C). The shorter proof of one-step tableau completeness then runs as follows. Let τ be a $\mathcal{P}W$-valuation, and let $\Xi = \{[D_1]\, a_1, \dots, [D_\alpha]\, a_\alpha, \langle C_1\rangle\, c_1, \dots, \langle C_\beta\rangle\, c_\beta\}$ (where $D_j \neq \Sigma$, $C_j \neq \Sigma$ for all j) be such that every rule match of (C^-) to Ξ has non-empty conclusion under τ. We have to construct an element of $[\![\Xi]\!]\tau$. Give every agent moves e_j for $j = 1, \dots, n$ intended as witnesses for $[D_j]\, a_j$, and a single refusal move $\bot$; write (slightly abusively) e_{D_j} for the joint move of D_j that is e_j in all components. Define a non-deterministic outcome function f by $f(m_\Sigma) = \bigcap_{e_{D_j} \sqsubseteq m_\Sigma} \tau(a_j)$, noting that this set is non-empty thanks to rule (CD) since for $j \neq k$, having both $e_{D_j} \sqsubseteq m_\Sigma$ and $e_{D_k} \sqsubseteq m_\Sigma$ implies $D_j \cap D_k = \emptyset$. Then f clearly satisfies $[D_j]\, a_j$ under τ. To see that f also satisfies $\langle C_j\rangle\, c_j$, let m_{C_j} be a joint move of C_j. Let m_Σ be the joint move of Σ extending m_{C_j} by letting all other agents pick $\bot$. We have to show that $f(m_\Sigma) \cap \tau(c_j) \neq \emptyset$. But this is immediate by rule (C^-), since $e_{D_k} \sqsubseteq m_\Sigma$ implies $D_k \subseteq C_j$.

We note further that excluding grand coalition modalities is equivalent to making the outcome function non-deterministic: It is clear that excluding grand coalition modalities is equivalent to always taking the set of agents to consist of the agents Σ_ϕ mentioned in the target formula ϕ and one extra agent $*$ (convert models with larger set C of additional agents into one with only $*$ by taking the previous joint moves of C to be the moves of $*$). Then, note that ϕ is satisfiable in a CGS with set $\Sigma_\phi \cup \{*\}$ of agents iff ϕ is satisfiable in a *non-deterministic CGS* with set $\Sigma = \Sigma_\phi$ of agents, where a non-deterministic CGS is defined like a CGS except that the outcome function f_q at a state q returns a non-empty set of possible post-states rather than just a single post-state. Over such a non-deterministic CGS, a formula $[C]\,\psi$ is satisfied at a state q if C has a joint move m_C such that for all joint moves $m_{\overline{C}}$ of $\overline{C}$, *all* possible post-states of q under the induced joint move of Σ satisfy ψ. A non-deterministic CGS with set Σ of agents is converted into a CGS with set $\Sigma \cup \{*\}$ of agents by giving $*$ all states as moves, allowing $*$ to pick one of the possible post-states determined by the other agents (with some possible post-state chosen arbitrarily if $*$ plays a state that is not a possible post-state). Conversely, a CGS S with set $\Sigma \cup \{*\}$ of agents is converted into a non-deterministic CGS with set Σ of agents by taking the possible post-states under a joint move m_Σ of the agents in Σ to be the set of all post-states of joint moves in S extending m_Σ. Both conversions clearly preserve satisfaction of formulae ϕ mentioning only agents in Σ.

Proof of One-step Tableau Completeness for the AMCDES (Lemma 6.2) with Finite Sets of Moves

Proof. Similarly to how the finite moves were achieved in the proof of Theorem 5.1, we will colour the moves to simulate the effect of the occurs check in unification. We use the same terminology and notation for colours as in the proof of Theorem 5.1, and take the colours from the same Abelian group U. Let ϕ be the clause set constructed in the ongoing proof as shown in the main part of the paper. Now, all agents receive (for simplicity) the same moves, namely

- moves $(e^j, 0)$ for $j = 1, \ldots, \alpha$, intended as witnesses for the moves of the agents in D_j in $[D_j, P_{G_j}]\, a_j$,
- moves $(p, 0)$ for $j = 1, \ldots, \alpha$, $p \in P_{G_j}$ witnessing explicit moves from $[D_j, P_{G_j}]\, a_j$,
- moves $(r, 0)$ for $j = 1, \ldots, \beta$, $r \in R_{H_j}$ witnessing explicit moves from $\langle C_j, R_{H_j}\rangle\, c_j$, and
- moves (g^j, u) for $j = 1, \ldots, \beta$ and $u \in U$, intended as witnesses for $\langle C_j, R_{H_j}\rangle\, c_j$.

Let $\mathcal{T}$ be the unification closure of all argument terms occuring in clauses in ϕ. All tuples in $\mathcal{T}$ have the shape $(x_A, e_B, p_C, r_D, g_{\overline{A \cup B \cup C \cup D}}(x_A, e_B, p_C, r_D))$ where the x_A are variables; the e_B are Skolem constants and p_C, r_D are constants for named moves, from possibly different boxes and diamonds; and the $g_{\overline{A \cup B \cup C \cup D}}$ are Skolem functions from a single diamond, as Skolem functions from multiple diamonds do not occur together in the starting terms and such occurrences are not introduced during unification due to the occurs check.

The (finite) model $\mathcal{G}$ is then defined over coloured moves. Skolem constants e^j are interpreted as $(e^j, 0)$, explicit strategies r and p are interpreted as $(r, 0)$ and $(p, 0)$, and Skolem functions g_i^j for $i \in \overline{C_j}$ are interpreted as mapping a joint move m_{C_j} of C_j to $(g^j, u_j - \mathsf{col}(m_{C_j}))$ if i is the least element of $\overline{C_j}$, and to $(g^j, 0)$ otherwise, thus ensuring that $\mathsf{col}(m_{C_j}, g^j(m_{C_j})) = u_j$. It remains to show that $\mathcal{G}$ is $\mathcal{T}$-equationally complete, obtaining by Theorem 4.5 and consistency of ϕ under set-valued first-order resolution that ϕ is satisfiable over $\mathcal{G}$. Indeed, observing that the symbols for explicit strategies represent constants in the unification process and are translated exactly like the Skolem constants, we can treat them as part of e_B and proceed in the same way as in Theorem 5.1. ◀