Cooperation and Stability in Modal Logics: Comparing Frameworks and Determining Descriptive Difficulty

Cédric Dégremont and Lena Kurzen

ILLC, Universiteit van Amsterdam

Abstract. We develop a unified perspective on modal logics for cooperation of agents with preferences. We prove embedding results between classes of modal logics for cooperation, clarifying the relations between them. We show how game– and social choice theoretical notions can be interpreted on three different classes of models, and identify via invariance results the expressive power the notions require. Explicit definability results in extended modal languages are given for each notion and class of models. Complexity results for extended modal logics are then used to obtain upper bounds on the complexity (model checking and SAT) of modal logics expressing the notions. This analysis shows how demanding certain game– and social choice theoretical notions are in terms of complexity and expressivity and how the choice of models (models with coalitional power as a primitive vs. more complex power or action based models) effects the expressive power and complexity required to express the notions. We found e.g. opposite results for different classes of models as to whether strict or non-strict stability notions are easier to express.

1 Introduction

We can think of cooperative and non-cooperative game theory (GT) as theories of *stability* of states of interactive systems, and social choice theory (SCT) as a theory of *fairness* and *efficiency* of such states. Further questions, e.g., from computational SCT, are whether the system might *loop* or is guaranteed to *terminate*. This paper contributes to the project of bringing the perspective of descriptive complexity to the analysis of problems raised by the analysis of multi-agent systems in terms of *stability*, *efficiency* and *termination*, and connected concepts used to reason about cooperation in philosophy, social sciences and computer science. We are concerned with studying the expressive power required for logical languages to reason about interactive systems in terms of such notions. Consequences in computational complexity can then be drawn, paving the way to a descriptive perspective on the complexity of certain types of GT- and SCT-reasoning. This requires an abstract perspective (as e.g. in [13]) and representing the action or power structure together with the preferences of agents as simple graphs, as it is done in modal logics (ML) for cooperation in multi-agent systems. We aim towards a unified perspective on MLs for cooperation of agents with preferences, both on a model-theoretic and syntactic level. Our objective is *not* to propose a new ML but a unifying perspective on existing ones.

One such logic s for cooperation is Coalition Logic (CL) [12]. It uses formulas $\langle\!\langle C \rangle\!\rangle \phi$ saying that coalition C has a joint strategy to ensure that ϕ. Another class of cooperation logics aims to make coalitional power explicit by representing the actions or strategies by which coalitions can achieve results [15, 4, 9]. Also the concept *preferences* plays a crucial role in reasoning about strategic interaction of (groups of) agents and has received much attention in ML [10].

Aim. First we want to determine the expressive power and complexity needed for MLs expressing GT and SCT concepts. This depends on the models under consideration. We analyze three classes of models for cooperation. Additionally, we clarify the results by analyzing the relation between the models, and their relation to existing frameworks. This comparison determines how demanding different notions are on each class of models. Our results help to make design choices when developing MLs for cooperation since we know the impact of certain choices on the complexity and expressive power required to express GT and SCT notions. Moreover, we clarify the relationship between complexity and expressive power results of existing cooperation logics by embedding results.

Methodology. First, we focus on classes of models/logics for cooperation with natural model theoretical properties. We consider different normal MLs and extend them with agents' preferences as total preorders over the states. We analyze the relations between them and to existing frameworks by giving embedding results. Then we focus on a set of notions of interest for reasoning about cooperation, and give their natural interpretations in each of the models. We determine the expressive power required by these notions by checking under which operations these properties are invariant. Using characterization results for extended MLs, we then obtain extended modal languages that can express the notions. Among these, we choose those with the lowest expressive power and give explicit definability results for the notions. Using known complexity results for extended MLs, we also obtain upper bounds (UB) on the complexity of MLs (satisfiability (SAT) and model checking (MC) (combined complexity)) expressing each notion.

Structure. Sec. 2 presents three classes of models for reasoning about cooperation and the interpretations of extended modal languages on them. Sec. 3 has our main results, first giving embedding results showing the relation between the modal cooperation frameworks we consider, and their relation to frameworks from the literature. Sec. 3.3 then gives invariance and explicit definability results for several GT and SCT properties, and also upper bounds on the complexity of MLs able to express them. Sec. 4 concludes. Proofs are in the full paper.

2 Three ways of modelling cooperation

The classes of models we consider correspond to models from the literature. We take simplifying models or generalizations, avoiding additional complexity due to constraints on the models. Thus, we can distinguish clearly how the notions themselves are demanding and evaluate from a high level perspective how appropriate the models are for reasoning about which aspects of cooperation.

First, *coalition labelled transition systems* [7] focus on preferences and their interaction with cooperation, simplifying the computation of coalitional powers as they are directly represented as accessibility relations. The second class, *action-based coalitional models*, represents coalitional power in terms of actions. The third class, [5] *power-based coalitional models*, focuses on reasoning about and computing coalitional power itself, encoding groups' choices as partitions of the state space. Preferences are total preorders (TPO) over the states.

2.1 The models

Our models are based on a finite set of agents $\mathtt{N}$. j ranges over $\mathtt{N}$. PROP is the set of propositional letters and NOM a set of nominals, which is disjoint from PROP. A nominal is true in exactly one state. We let $p \in$ PROP and $i \in$ NOM.

Coalition-labelled transition systems. *Sequential* systems, Kripke models with a relation for each coalition, can be used for reasoning about coalitional power: a group has the power to move the system into exactly the states accessible by its relation. These models generalize [13]'s conversion/preference games and [14]'s models of coalitional interaction. The former take an abstract view on game-theoretical models, based on the idea that GT is a theory of stable vs. unstable states in interactive systems. Here, the focus is not on how coalitional power arises from individuals powers, but coalitional power is taken as a primitive. Thus we can focus on the expressive power required by the notions themselves and by reasoning about preferences.

Definition 1. *A* $\wp(\mathtt{N})$-LTS *(Labeled Transition Systems indexed by coalitions in* $\wp(\mathtt{N})$*) is of the form* $\langle W, \mathtt{N}, \{ \overset{C}{\rightarrow} \mid C \subseteq \mathtt{N}\}, \{ \leq_j \mid j \in \mathtt{N}\}, V\rangle$*, where* $W \neq \emptyset$*,* $\mathtt{N} = \{1, \ldots, n\}$ *for some* $n \in \mathbb{N}$*,* $\overset{C}{\rightarrow} \subseteq W \times W$ *for each* $C \subseteq \mathtt{N}$*,* $\leq_j \subseteq W \times W$ *for each* $j \in \mathtt{N}$*, and* $V : \text{PROP} \cup \text{NOM} \rightarrow \wp(W)$*,* $|V(i)| = 1$ *for each* $i \in$ NOM.

W is a set of states. $w \overset{C}{\rightarrow} v$ means 'coalition C can change the state from w into v'. $\leq$ is a TPO. $w \leq_j v$ means 'j finds v at least as good (a.l.a.g) as w'. $w \in V(p)$ means that p is true at w. As an example, a multi-agent resource allocation setting as in [8] has a set $\mathcal{R}$ of resources. Let $W = \mathtt{N}^{\mathcal{R}}$, i.e. the state space is the set of all allocations. If a deal $\delta = (A, A')$ involves exactly agents in $C \subseteq \mathtt{N}$, add (A, A') to $\overset{C}{\rightarrow}$. Preference relations are defined in the obvious way. We can then study resource allocation settings from an abstract perspective as graphs, and find logical characterizations of relevant properties; e.g. the existence of loops.

Action-based coalitional models. In action-based models, coalitional power comes from the agents' abilities to perform actions, as in models in [4, 15].

Definition 2 (ABC). *A* $\mathtt{N}, (\mathtt{A_j})_{\mathtt{j} \in \mathtt{N}}$-ABC *(action-based coalitional model indexed by a finite set of agents* $\mathtt{N}$ *and a collection of finite sets of actions* $(A_j)_{j \in \mathtt{N}}$*) is of the form* $\langle W, \mathtt{N}, \{\overset{j,a}{\longrightarrow} \mid j \in \mathtt{N}, a \in A_j\}, \{\leq_j \mid j \in \mathtt{N}\}, V\rangle$*, where* $W \neq \emptyset$*,*

$\mathtt{N} = \{1, \ldots, n\}$*, for some* $n \in \mathbb{N}$*; for each* $j \in \mathtt{N}$ A_j *is a finite set,* $\xrightarrow{j,a} \subseteq W \times W$ *for each* $j \in \mathtt{N}, a \in A_j$*,* $\leq_j \subseteq W \times W$ *is a* TPO *for each* $j \in \mathtt{N}$*, and* $V : \text{PROP} \cup \text{NOM} \to \wp(W)$*,* $|V(i)| = 1$ *for each* $i \in \text{NOM}$*. For* $R \subseteq W \times W$*, we write* $R[w] := \{v \in W \mid wRv\}$*.*

$\xrightarrow{j,a} [w] \subseteq X$ means that at w, j can guarantee by doing a that the next state is in X. This holds iff for some Y,$X \supseteq Y \in \{\xrightarrow{j,a} [w] \mid a \in A_j\}$; ($Y$ is then in the *exact power* of j at w). We let powers be additive: powers of coalitions arise from individuals' powers. W.l.o.g. let $C = \{1, \ldots, |C|\}$. Then, at w, $C \subseteq \mathtt{N}$ can force the next state to be in X iff for some Y, $X \supseteq Y \in \{\bigcap_{j \in C} \xrightarrow{j,a_j} [w] \mid (a_1, \ldots, a_{|C|}) \in \times_{j \in C} A_j\}$; ($Y$ is in the exact power of C at w).

An ABC model is *reactive*, if the following two conditions are fulfilled: 1. for any $(a_j)_{j \in \mathtt{N}} \in \times_{j \in N}(A_j)$, and for all w, $(\bigcap_{j \in \mathtt{N}} \xrightarrow{j,a_j} [w]) \neq \emptyset$, i.e. for every collective choice there is some next state. 2. agents always have available actions: for all $j \in \mathtt{N}$ and $w \in W$, there is some $a_j \in A_j$ such that $\xrightarrow{j,a_j} [w] \neq \emptyset$. We say that an ABC model $\mathcal{M}$ is N-*determined* if for all $w \in W$, if there is some $j \in \mathtt{N}$ and some $a_j \in A_j$ with $v \in \xrightarrow{j,a_j} [w]$, then there is a profile $(a_j)_{j \in \mathtt{N}} \in \times_{j \in \mathtt{N}}(A_j)_j$ with $(\bigcap_{j \in \mathtt{N}} \xrightarrow{j,a_j} [w]) = \{v\}$. $\mathtt{ABC}^{\mathtt{N}R}$ is the class of N-*determined* reactive ABC models.

Power-based coalitional models. These models generalize those of CL's normal simulation NCL [5], and additionally have a preference relation for each agent.

Definition 3 (PBC-Model). *A* $\wp(\mathtt{N})$-PBC-*model (power based coalitional model indexed by a finite set of coalitions* $\wp(\mathtt{N})$*) is a tuple* $\langle W, \mathtt{N}, \{\sim_C \mid C \subseteq \mathtt{N}\}, F_{\mathbf{X}}, \{\leq_j \mid j \in \mathtt{N}\}, V\rangle$*, where* $W \neq \emptyset$*, each* $\sim_C \subseteq W \times W$ *is an equivalence relation,* $F_{\mathbf{X}} : W \to W$ *is a total function,* $\leq_j \subseteq W \times W$ *is a* TPO *for each* $j \in \mathtt{N}$*, and* $V : \text{PROP} \cup \text{NOM} \to \wp(W)$*,* $|V(i)| = 1$ *for each* $i \in \text{NOM}$*.*

$F_{\mathbf{X}}$ determines the system's actual course: from w, we move to $F_{\mathbf{X}}(w)$. $\sim_C$ describes C's lack of power: $w \sim_C v$ means that C cannot decide between w and v and thus neither whether we move to $F_{\mathbf{X}}(w)$ or $F_{\mathbf{X}}(v)$. But C can choose an equivalence class $[w]_{\sim_C}$, thus restricting the set of possible next states to $F_{\mathbf{X}}[[w]_{\sim_C}]$. The models of NCL are PBC models with additional properties.

Definition 4 (NCL-Independence). *For every* $C \subseteq \mathtt{N}$*,* $\sim_\emptyset \subseteq (\sim_C \circ \sim_{\overline{C}})$*.*

Definition 5 (NCL-Model). *An* NCL *model is a* PBC *model satisfying the following conditions:*

1. *For all* $C, D \subseteq \mathtt{N}$*, if* $D \subseteq C$*, then* $\sim_C \subseteq \sim_D$ *.*
2. NCL-*Independence.*
3. $\sim_{\mathtt{N}} = id = \{(w, v) \in W \times W \mid w = v\}$*.*

Alternating-time temporal models. An alternating transition system [1] is a tuple $\langle W, \mathtt{N}, \delta, V\rangle$ where $W \neq \emptyset$, $\delta : W \times \mathtt{N} \to \wp(\wp(W))$ and satisfies *Consistency*: for all $C \subseteq \wp(\mathtt{N})$, $w \in W$ and $(X_j)_{j \in C} \in \times_{j \in C}\delta(w, j)$ we have $(\bigcap_{j \in C} X_j) = \emptyset$.

2.2 Extended modal languages

For each type of models, we introduce a language from which we will actually use different fragments to define different GT and SCT notions.
Language interpreted on $\wp(\mathtt{N}) - \mathtt{LTS}$. $\alpha ::= \ \leq_j \ | \ C \ | \ \alpha \cap \alpha \ | \ \overline{\alpha}$
$\phi ::= \ p \mid i \mid x \mid \neg\phi \mid \phi \wedge \phi \mid \langle\alpha\rangle\phi \mid @_i\phi \mid @_x\phi \mid \ \downarrow x.\phi$ where $j \in \mathtt{N}, C \in \wp(\mathtt{N}) - \{\emptyset\}$, $p \in \textsc{prop}$, $i \in \textsc{nom}$, $x \in \textsc{svar}$. $\textsc{svar}$ is a countable set of variables.

Semantics. Programs α are interpreted as relations. Formulas are interpreted with an assignment $g : \textsc{svar} \to W$. We skip booleans.

$$
\begin{array}{lll}
\mathcal{M}, w, g \Vdash i \text{ iff } w \in V(i) & & \mathcal{M}, w, g \Vdash \langle\alpha\rangle\phi \text{ iff } \exists v : wR_\alpha v \\
\mathcal{M}, w, g \Vdash x \text{ iff } w = g(x) & & \qquad\qquad \text{and } \mathcal{M}, v, g \Vdash \phi \\
R_{\leq_j} & = \leq_j & \mathcal{M}, w, g, \Vdash @_i\phi \text{ iff } \mathcal{M}, v, g \Vdash \phi \\
R_C & = \xrightarrow{C} & \qquad\qquad \text{for } V(i) = \{v\} \\
R_{\beta\cap\gamma} & = R_\beta \cap R_\gamma & \mathcal{M}, w, g, \Vdash @_x\phi \text{ iff } \mathcal{M}, g(x), g \Vdash \phi \\
R_{\overline{\beta}} & = (W \times W) \setminus R_\beta & \mathcal{M}, w, g, \Vdash \downarrow x.\phi \text{ iff } \mathcal{M}, w, g[x := w] \Vdash \phi
\end{array}
$$

Language interpreted on ABC models. We now give the basic language for ABC (the extension with hybrid and boolean ML formulas is as for $\wp(\mathtt{N}) - \mathtt{LTS}$). More precisely, we have a family of languages indexed by collections $(A_j)_{j\in\mathtt{N}}$.
$\alpha ::= \ \leq_j \ | \ a_j \mid \alpha^{-1} \mid \alpha\cap\alpha \mid \overline{\alpha}$ $\phi ::= \ p \mid i \mid x \mid \neg\phi \mid \phi\wedge\phi \mid \langle\alpha\rangle\phi \mid @_i\phi \mid @_x\phi \mid \ \downarrow x.\phi$ where $j \in \mathtt{N}$, $a_j \in A_j$ (the set of actions available to j) and $p \in \textsc{prop}$.

$$R_{a_j} = \xrightarrow{j,a} \qquad R_{\alpha^{-1}} = \{(v, w) | wR_\alpha v\}$$

We only give a few clauses to give the intuition.

$$
\begin{array}{l}
\mathcal{M}, w, g \Vdash \langle a_j\rangle\phi \text{ iff } \exists v : w \xrightarrow{j,a} v \text{ and } \mathcal{M}, v, g \Vdash \phi \\
\mathcal{M}, w, g \Vdash \langle\leq_j\rangle\phi \text{ iff } \exists v : w \leq_j v \text{ and } \mathcal{M}, v, g \Vdash \phi \\
\mathcal{M}, w, g \Vdash \langle\alpha\rangle\phi \text{ iff } \exists v : wR_\alpha v \text{ and } \mathcal{M}, v, g \Vdash \phi
\end{array}
$$

We will make use of some shortcuts when writing big disjunctions or unions. For $C \subseteq \mathtt{N}$, we let $\boldsymbol{C} := \times_{j\in C} A_j$. For an action profile $\boldsymbol{a_j} = (a_j)_{j\in N} \in \boldsymbol{C}$ we often write $\bigcap \boldsymbol{a_j}$ to stand for $\bigcap_{j\in C} a_j$. As an example, for the language indexed by $A_1 = T_1, M_1, B_1$ and $A_2 = L_2, R_2$ instead of writing $[T_1 \cap L_2]p \vee [M_1 \cap L_2]p \vee [B_1\cap L_2]p \vee [T_1\cap R_2]p \vee [M_1\cap R_2]p \vee [B_1\cap R_2]p$, we often write $\bigvee_{\boldsymbol{a_j}\in\boldsymbol{\{1,2\}}}[\cap\boldsymbol{a_j}]p$.

Language interpreted on PBC/NCL models. The language $\mathcal{L}_{\mathtt{NCL}}$ for PBC and NCL is given as defined in [5], extended with hybrid and boolean modal logic formulas as for $\wp(\mathtt{N}) - \mathtt{LTS}$.

$$\alpha ::= \ \leq_j \big| \alpha^{-1} \big| \alpha \cap \alpha \quad \phi ::= \ p \big| i \big| x \big| \neg\phi \big| \phi \wedge \phi \big| \langle C\rangle\phi \big| \mathbf{X}\phi \big| \langle\alpha\rangle\phi \big| @_i\phi \big| @_x\phi \big| \downarrow x.\phi$$

where $j \in \mathtt{N}$ (the set of agents), $C \in \wp(\mathtt{N})$ and $p \in \textsc{prop}$.

$$
\begin{array}{ll}
\mathcal{M}, w, g \Vdash \langle C\rangle\phi & \text{iff } \exists v : w \sim_C v \text{ and } \mathcal{M}, v, g \Vdash \phi \\
\mathcal{M}, w, g \Vdash \mathbf{X}\phi & \text{iff } \mathcal{M}, F_{\mathbf{X}}(w), g \Vdash \phi \\
\mathcal{M}, w, g \Vdash \langle\leq_j\rangle\phi & \text{iff } \exists v : w \leq_j v \text{ and } \mathcal{M}, v, g \Vdash \phi \\
\mathcal{M}, w, g \Vdash \langle\alpha\rangle\phi & \text{iff } \exists v : wR_\alpha v \text{ and } \mathcal{M}, v, g \Vdash \phi
\end{array}
$$

Language interpreted on alternating transition systems. $\mathcal{L}_{\texttt{ATL}}$ is defined as follows: $\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \langle\langle C\rangle\rangle \mathbf{X}\phi$; p ranging over PROP and C over $\wp(\texttt{N})$. Now for finite sequences λ, let $Last(\lambda)$ be the last element of λ, and W^+ the set of non-empty finite sequences. For $j \in \texttt{N}$, let a strategy for j be a function $f_j : W^+ \to \wp(W)$, such that for each finite sequence of states λ, $f_j(\lambda) \in \delta(j, Last(\lambda))$. Let a collective strategy for C be $F_C = (f_j)_{j\in C}$. $out(w, F_C) = \{\lambda|\lambda[0] = w\&\forall i \geq 0(w_{i+1} \in \bigcap_{j\in C} f_j(\lambda_i))\}$ where λ_i is the prefix of λ of length $i+1$. $\mathcal{L}_{\texttt{ATL}}$ is interpreted as follows: $\mathcal{M}, w \Vdash \langle\langle C\rangle\rangle \mathbf{X}\phi$ iff $\exists F_C : \forall\lambda \in out(w, F_C) : \mathcal{M}, \lambda[1] \Vdash \phi$.

$(\bot)$	$\vdash \neg\langle\langle C\rangle\rangle \mathbf{X}\bot$
$(\top)$	$\vdash \langle\langle C\rangle\rangle \mathbf{X}\top$
(Σ)	$\vdash \neg\langle\langle\emptyset\rangle\rangle \mathbf{X}\neg\phi \rightarrow \langle\langle\Sigma\rangle\rangle \mathbf{X}\phi$
(S)	$\vdash (\langle\langle C_1\rangle\rangle \mathbf{X}\phi \wedge \langle\langle C_2\rangle\rangle \mathbf{X}\psi) \rightarrow \langle\langle C_1 \cup C_2\rangle\rangle \mathbf{X}(\phi \wedge \psi)$ for $C_1 \cap C_2 = \emptyset$
$(\langle\langle C\rangle\rangle \mathbf{X}mon)$	$\vdash \phi \rightarrow \psi$ implies $\vdash \langle\langle C\rangle\rangle \mathbf{X}\phi \rightarrow \langle\langle C\rangle\rangle \mathbf{X}\psi$

Table 1. Axiomatization of ATL.

We introduced all the frameworks for which this paper investigates how demanding reasoning about cooperation is. The languages defined above will be used to express SCT and GT notions on the respective models. We will also clarify the relationship between ATL and classes of models we considered.

3 Comparing modal logics for cooperation

This section gives our main results from the analysis of the different ways of modelling cooperation. First we determine the relation between the classes of models; then we analyze how demanding different concepts from GT and SCT are on them. We start by analyzing coalitional power as modelled in PBC, NCL and CL and analyze relations between standard assumptions on coalitional power.

3.1 On the relation between PBC and NCL models

We say that C can *force* a set X at w iff at w C can guarantee that the next state is in X; i.e. C can force X if some subset of X is in the exact power of C at w. Some reasonable assumptions about the coalitional powers reflect the independence of agents and are generally assumed in the literature [12, 5, 2]. We consider two assumptions and show their relation. Let $P_C(w)$ be the collection of exact powers of C at w; $P_C(w)$ contains the possible sets of states coalition C can choose from at w. Let $\overline{C} = \texttt{N} \setminus C$ and $\overline{X} = W \setminus X$. Independence of coalitions says that for all choices of two disjoint coalitions there is a resulting next state.

Definition 6 (Independence of coalitions (IC)). *$\forall w$, if $C \cap D = \emptyset$ then $\forall X \in P_C(w)$ $\forall Y \in P_D(w)$: $X \cap Y \neq \emptyset$.*

The next condition says that the powers of C and $\overline{C}$ have to be consistent.

Definition 7 (Condition about complementary coalitions (CCC)). *$\forall w$, $\forall X$, if $\exists X'$ with $X \supseteq X' \in P_C(w)$, then there is no Y such that $\overline{X} \supseteq Y \in P_{\overline{C}}(w)$.*

Coalition monotonicity says that if a coalition can achieve something then so can all supersets of it.

Definition 8 (Coalition monotonicity (CM)). *$\forall w\, \forall X$, if $C \subseteq D$ and $\exists Y$ such that $X \supseteq Y \in P_C(w)$, then $\exists Z$ such that $X \supseteq Z \in P_D(w)$.*

Fact 1. IC implies CCC. **Fact 2.** CCC + CM implies IC.
Note that on `PBC` models, CCC is actually the following:

$$\forall w\ [\forall X \text{ if } \exists v (v \in \sim_\emptyset [w] \text{ and } \sim_C [v] \subseteq X), \text{ then } \neg\exists t (t \in \sim_\emptyset [w] \text{ and } \sim_{\overline{C}} [t] \subseteq \overline{X})].$$

For `NCL`, [5] takes the condition of `NCL`-Independence (Def. 4).

Proposition 1. *On `PBC` models, CCC is equivalent to `NCL`-Independence.*

To sum up, we have shown that IC implies CCC; and together with CM, CCC also implies IC. Moreover, IC and `NCL`-Independence are actually equivalent.

3.2 On the relation between `NCL` and `CL`

Let us analyze the relation between `CL` and its normal simulation `NCL`. First, we briefly recall the semantics of `CL`. For the details we refer the reader to [12].

Definition 9 (`CL`-Model). *A `CL`-model is a pair $((\mathtt{N}, W, E), V)$ where $\mathtt{N}$ is a set of agents, $S \neq \emptyset$ is a set of states, $E : W \to (\wp(\mathtt{N}) \to \wp(\wp(W)))$ is called an effectivity structure. It satisfies the conditions of **playability**:*

- *Liveness:* $\forall C \subseteq \mathtt{N} : \emptyset \notin E(C)$,
- *Termination:* $\forall C \subseteq \mathtt{N} : W \in E(C)$,
- $\mathtt{N}$*-maximality.* $\forall X \subseteq W : (W \setminus X \notin E(\emptyset) \Rightarrow X \in E(N))$
- *Outcome monotonicity.* $\forall X \subseteq X' \subseteq W, C \subseteq \mathtt{N} : (X \in E(C) \Rightarrow X' \in E(C))$,
- *Superadditivity.* $\forall X_1, X_2 \subseteq W, C_1, C_2 \subseteq \mathtt{N} : ((C_1 \cap C_2 = \emptyset\ \&\ X_1 \in E(C_1)\ \&\ X_2 \in E(C_2)) \Rightarrow X_1 \cap X_2 \in E(C_1 \cup C_2))$.

$V : \text{PROP} \to \wp(W)$ is a propositional valuation function.

The language $\mathcal{L}_{\mathtt{CL}}$ of `CL` is a standard modal language with a modality $\langle\!\langle C \rangle\!\rangle$ for each $C \subseteq \mathtt{N}$. The intended meaning of $\langle\!\langle C \rangle\!\rangle\, \phi$ is "coalition C has the power to achieve that ϕ". The semantics is as follows: $M, w \vDash \langle\!\langle C \rangle\!\rangle\, \phi$ iff $[\![\phi]\!]_M \in E(w)(C)$. Let us now give a brief overview of `NCL`. In [5], a translation τ from $\mathcal{L}_{\mathtt{CL}}$ to $\mathcal{L}_{\mathtt{NCL}}$ is given such that for all $\phi \in \mathcal{L}_{\mathtt{CL}}$, ϕ is satisfiable in an `CL` model iff $\tau(\phi)$ is satisfiable in an `NCL` model. τ is defined as follows: $\tau(p) = p, \tau(\langle\!\langle C \rangle\!\rangle\, \phi) = \langle \emptyset \rangle [C] \mathbf{X} \tau(\phi)$. The main result is then that ϕ is a theorem of `CL` iff $\tau(\phi)$ is one of `NCL`. Via completeness of `CL` and soundness of `NCL`, it follows that whenever $\tau(\phi)$ is satisfied in an `NCL` model, then there is a `CL` model that satisfies ϕ.

We give a *constructive* proof of their result to get a clear view of how the two frameworks are related. We show how to translate pointed `NCL` models (M, w) into `CL` models $f(M, w)$ such that for all $\phi \in \mathcal{L}_{\mathtt{CL}}$, $(M, w) \Vdash \tau(\phi)$ iff $f(M, w) \Vdash \phi$.

Proposition 2 ([5]). *For all $\phi \in \mathcal{L}_{\texttt{CL}}$, if $\tau(\phi)$ is satisfiable in a pointed model M, w of* NCL*, then ϕ is satisfiable in a model $f(M, w)$ of* CL*.*

Proof. The full paper has a new, constructive proof of this result.

So, we have shown how to transform NCL models into corresponding CL models, thus shedding some light on the relation between the two frameworks.

On the relation between ABC and ATL.

We give a translation $tr : \mathcal{L}_{\texttt{ATL}} \to \mathcal{L}_{\texttt{ABC}}$ such that for any $\phi \in \mathcal{L}_{\texttt{ATL}}$ there is a pointed ATL model $\mathcal{M}, w$ such that $\mathcal{M}, w \Vdash \phi$ iff there is an ABC model $\mathcal{M}', v$ with $\mathcal{M}', v \Vdash tr(\phi)$. Given a pointed alternating transition systems $\mathcal{M}, w$ with $\mathcal{M} = \langle W, \texttt{N}, \delta, V\rangle$ and $\mathcal{M}, w \Vdash \phi$, we show how to construct an ABC model $TR(\mathcal{M})$ with $\mathcal{M}, w \Vdash \phi$ iff $TR(M), f(v) \Vdash tr(\phi)$ where f is a function from the domain of $\mathcal{M}$ to that of $TR(M)$. First, we recall an important result we need:

Theorem 1 ([11]). *Every satisfiable formula $\phi \in \mathcal{L}_{\texttt{ATL}}$ is satisfiable in a finite Concurrent Game System.*

Concurrent Games Systems (CGS) are almost the same as alternating transition systems and it is thus easy to give a transformation in both directions such that satisfiability is invariant and the size of the target model is bounded:

Corollary 2 *Every satisfiable formula $\phi \in \mathcal{L}_{\texttt{ATL}}$ is satisfiable in a finite alternating transition system.*

Henceforth, we assume a finite domain of ATL models.

Transforming ATL models into ABC models. We give a procedural definition of our transformation. Copy the state space W, the valuation V and the set of agents N. For all pairs $(w, i) \in W \times \texttt{N}$, $\delta(w, i)$ is finite. Label each element in $\delta(w, i)$ with an action name $a^1_{w,i}, \ldots, a^{|\delta(w,i)|}_{w,i}$. Let *Label* be this labeling function. Now for each set of states $X^w_i \in \delta(w, i)$ and for each $v \in X^w_i$ we add the pair (w, v) to $\xrightarrow{i, a^k_{w,i}}$ where $a^k_{w,i}$ is the appropriate label, i.e. $a^k_{w,i} = Label(w, i, X^w_i)$. We define a function $f : Dom(\mathcal{M}) \to Dom(TR(\mathcal{M}))$ mapping a state to itself.

Translating $\mathcal{L}_{\texttt{ATL}}$ into $\mathcal{L}_{\texttt{ABC}}$. The translation is model-dependent. Given an ATL model $\mathcal{M}$, define for each $j \in \texttt{N}$ a set of actions $A_j = \bigcup_{w \in |\mathcal{M}|} \bigcup_{X^w_j \in \delta(w,j)} Label(w, j, X^w_j)$. Since $|W|$ and $\delta(w, i)$ are finite, so is A_j. The translation $tr : \mathcal{L}_{\texttt{ATL}} \to \mathcal{L}_{\texttt{ABC}}$ is recursively defined as follows:

$$\begin{aligned}
tr(p) &:= p \qquad\qquad tr(\neg\phi) := \neg tr(\phi)\\
tr(\phi \wedge \psi) &:= tr(\phi) \wedge tr(\psi)\\
tr(\langle\langle C\rangle\rangle \mathbf{X}\phi) &:= \textstyle\bigvee_{\mathbf{c} \in \times_{j \in C} A_j} [\bigcap_{a_j \in |\mathbf{c}|} a_j] tr(\phi), C \neq \emptyset\\
tr(\langle\langle \emptyset\rangle\rangle \mathbf{X}\phi) &:= [\textstyle\bigcup_{j \in \texttt{N}} \bigcup_{a_j \in A_j} a_j] tr(\phi)
\end{aligned}$$

Lemma 1. *For all $\phi \in \mathcal{L}_{\texttt{ATL}}$ if there exists a pointed* ATL *model $\mathcal{M}, w$ such that $\mathcal{M}, w \Vdash \phi$ then there exists a reactive $\texttt{ABC}^{\texttt{N}}$ model $\mathcal{M}', v$ such that $M', v \Vdash tr(\phi)$*

Lemma 2. $\forall \phi \in \mathcal{L}_{\texttt{ATL}}$*: if $\models_{\texttt{ATL}} \phi$ then $\models_{\texttt{ABC}^{\texttt{N}R}} tr(\phi)$.*

Proposition 3. *For all $\phi \in \mathcal{L}_{\texttt{ATL}}$ there exists a pointed* ATL *model $\mathcal{M}, w$ such that $\mathcal{M}, w \Vdash \phi$ iff there exists an $\texttt{ABC}^{\texttt{N}R}$ model $\mathcal{M}', v$ such that $M', v \Vdash tr(\phi)$.*

3.3 What GT/SCT notions demand: complexity and expressivity.

This section analyzes the complexity of describing and reasoning about interactive systems from an abstract perspective. We summarize the main results that we obtained when investigating how much expressive power and complexity is required for expressing each of the notions on each class of models. Our analysis works towards establishing a link between descriptive complexity and complexity results from a computational social choice and algorithmic game theory perspective. As mentioned, we obtain our results by determining under which operations on models (frames) certain properties from GT and SCT are invariant (closed). For the definitions of these operations and the underlying characterization results, the reader is referred to [3, 6]. Before we start let us mention that all notions that we discuss are first-order and therefore their data complexity is in LOGSPACE. In general termination notion are examples of notions that would not be first-order-definable but definable in first-order logic with least fixed points. We start with the simplest notions of coalitional power and preferences.

Simple coalitional power and preference. The property of a coalition C having the power to ensure that in the next state p will be the case turns out to be invariant under bisimulations on $\wp(\mathtt{N})-\mathtt{LTS}$ and on $\mathtt{PBC}, \mathtt{NCL}$. Thus, it can be expressed using the respective basic multi-modal languages, i.e. by $\langle C\rangle p$ and $\langle\emptyset\rangle[\mathrm{C}]\mathbf{X}\mathrm{p}$, respectively. Since the complexity of MC and SAT of these logics is known, for $\wp(\mathtt{N})-\mathtt{LTS}$ and $\mathtt{PBC}$, we thus get PSPACE and P as upper bounds on SAT and MC of logics expressing the notion. For $\mathtt{NCL}$, the respective upper bounds are NEXPTIME and P. On $\mathtt{ABC}$ models on the other hand, saying that a coalition can achieve something involves the intersection of the relations for the actions for the agents. It is not invariant under bisimulation but under $\cap$-bisimulation; thus it can be expressed in the basic language with intersection: $\bigvee_{\boldsymbol{a_j}\in \boldsymbol{C}}[\bigcap \boldsymbol{a_j}]p$. The upper bounds on SAT and MC that we obtain are then again PSPACE and P, respectively.

	Invariance	Formula	UB for MC, SAT
$\wp(\mathtt{N})-\mathtt{LTS}$	Bisimulation	$\langle C\rangle p$	P, PSPACE
$\mathtt{ABC}$	$\cap$-Bisimulation	$\bigvee_{\boldsymbol{a_j}\in \boldsymbol{C}}[\bigcap \boldsymbol{a_j}]p$	P, PSPACE
$\mathtt{PBC}$	Bisimulation	$\langle\emptyset\rangle[C]\mathbf{X}p$	P, PSPACE

Table 2. *"C can ensure that in the next state p is true."*

The simplest preference notion is that of an agent finding some state at least as good in which p is true. Since, in all our models preferences are represented in the same way and the preference fragments of the different languages we consider are the same, we get the same results for this notion on all the models.

Coalition C can make agent j happy. The basic combination of coalitional power and individual preference is the ability of a coalition to ensure that the

	Invariance	Formula	UB for MC, SAT
$\wp(\texttt{N}) - \texttt{LTS}, \texttt{ABC}, \texttt{PBC}$	Bisimulation	$\langle \leq_j \rangle p$	P, PSPACE

Table 3. *"j finds a state a.l.a.g. where p is true."*.

next state will be one that is at least as good for some agent. This property turns out to be easiest to express on $\wp(\texttt{N}) - \texttt{LTS}$, since here it is invariant under $\cap$-bisimulation. For ABC and PBC on the other hand, we have to express that the states accessible by one relation are a subset of the states accessible by another relation. This is not invariant under any bisimulations but under taking generated submodels and disjoint unions.

	Formula	MC, SAT
$\wp(\texttt{N})\texttt{LTS}$	$\langle C\cap \leq_j \rangle \top$	P, PSPACE
ABC	$\bigvee_{\boldsymbol{a_j} \in \boldsymbol{C}} (\downarrow x.[\bigcap \boldsymbol{a_j}](\downarrow y.@_x \langle \leq_j \rangle y))$	PSPACE, Π^0_1
PBC	$\downarrow x.\langle \emptyset \rangle [C] \mathbf{X} \downarrow y.@_x \langle \leq_j \rangle y$	PSPACE, Π^0_1

Table 4. *"C can move the system into a state a.l.a.g. for j"*

Nash-stability. Nash-stability says that no single agent has the power to make the system move into a state that is *strictly* better for him.

	Formula	UB for SAT
$\wp(\texttt{N}) - \texttt{LTS}$	$\bigwedge_{j\in \texttt{N}} \downarrow x.[j\cap \leq_j]\langle \leq_i \rangle x$	Π^0_1
ABC	$\bigwedge_{j\in \texttt{N}} \bigwedge_{a_j \in A_j} \downarrow x.\langle a_j \rangle \langle \leq \rangle x$	EXPTIME
PBC	$\bigwedge_{j\in \texttt{N}} \downarrow x.[\emptyset]\langle \{j\} \rangle \mathbf{X} \langle \leq \rangle x$	EXPTIME

Table 5. "*The current state is Nash stable.*"

On all these models, Nash-stability is invariant under taking generated submodels and disjoint unions, and can be expressed in a modal logic with model checking problem (combined complexity) in PSPACE.

Strong Nash-stability. Strong Nash-stability says that no single agent has the power to make the system move into a state that is *a.l.a.g.* for him. Since we take preferences as TPOs, if a state is strongly Nash-stable, it is Nash-stable.

On $\wp(\texttt{N}) - \texttt{LTS}$, strong Nash-stability is invariant under $\cap$-bisimulation. On ABC and PBC only under GSM and DU. Comparing with the results for Nash-stability, we can see that on $\wp(\texttt{N}) - \texttt{LTS}$ strong Nash-stability is easier to express than Nash-stability whereas on ABC and PBC we get opposite results.

	Formula	UB for SAT
$\wp(\mathtt{N}) - \mathtt{LTS}$	$\bigwedge_{j\in\mathtt{N}}[i\cap\leq_j]\bot$	$\mathsf{P}, \mathsf{PSPACE}$
ABC	$\neg\bigvee_{j\in\mathtt{N}}\bigvee_{a_j\in A_j}\downarrow x.[a_j]\langle\leq^{-1}\rangle x$	PSPACE, Π^0_1
PBC	$\neg\bigvee_{j\in\mathtt{N}}\downarrow x.\langle\emptyset\rangle[\{j\}]\mathbf{X}\langle\leq^{-1}\rangle x$	PSPACE, Π^0_1

Table 6. "*The current state is strongly Nash stable.*"

4 Conclusion

Our embeddings results show that action- and power-based models, together with coalition-labelled transition systems, constitute three natural families of cooperation logics with different primitives. The main open problem is to extend action-based models to reason about transitive closure in order to simulate more powerful logics such as $\mathtt{ATL}^*$. The stability notions that we considered in this work express that agents do not have an incentive to change the current state within one step. In order to express more sophisticated stability notions for interactive systems fixed point logics such as the modal μ-calculus are needed.

Our invariance results showed that many social choice-theoretical and game-theoretical notions are not invariant under bounded morphic images, in many cases it is only a matter of allowing the underlying logics to reason about the intersection of two relations. Being able to express intersection is crucial when reasoning about cooperation of agents in normal MLs.

Our definability results together with known upper bounds on combined complexity of model checking and satisfiability have shown that whether strong or weak stability or efficiency notions are less demanding crucially depends on the choice of primitives. In action- and power-based models expressing the latter type notions turns out to be easier, while in coalition labelled transition systems the situation is just the opposite. This has to do with whether coalitional power can be expressed in a simple way, and thus whether the intersection of relations is sufficient or whether we need to express something like a subset relation.

Our definability results made use of very big conjunctions and disjunctions. When taking conjunctions/disjunctions over all coalitions, they will be exponentially related to the number of agents. The consequences we draw about the upper bounds on the complexity of satisfiability or of combined complexity of model checking is thus to be balanced by the fact that we generally use very big conjunctions or disjunctions that might well be exponential if we take the number of agents as a parameter for the complexity results.

Our invariance results indicate that our definability results are tight to some extent. Indeed, this shows that within a large family of extended modal languages with a natural model-theoretical characterization we could not improve on them. It follows that upper bounds are accurate to some extent. Naturally, it is always possible to design ad hoc logics to express exactly the notion of interest. It leads us to the question of lower bounds. Can we use results from the computational social choice literature to obtain lower bounds on the *data* complexity of model-checking a logic that can express some notion? In general, the difficulty is that usually the results from this literature take e.g. the number of resources (and/or number of agents) as primitives, while the data complexity of a modal logic is usually taken relatively to its state space, which is in general exponentially bigger than the number of resources. It is natural to expect that the interesting hardness results would be for logarithmic complexity classes.

References

1. Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. Alternating-time temporal logic. In *Compositionality: The Significant Difference*, volume 1536 of *Lecture Notes in Computer Science*, pages 23–60. Springer-Verlag, 1998.
2. Nuel Belnap, Michael Perloff, and Ming Xu. *Facing the future: Agents and choices in our indeterminist world.* Oxford University Press, Oxford, 2001.
3. Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*, volume 53. CUP, MA, 2001.
4. Stefano Borgo. Coalitions in action logic. In *IJCAI '07: Proceedings of the twentieth international joint conference on Artificial Intelligence*, pages 1822–1827, Hyderabad, India, 2007.
5. Jan Broersen, Andreas Herzig, and Nicolas Troquard. Normal Coalition Logic and its conformant extension. In Dov Samet, editor, *TARK'07*, pages 91–101. PUL, 2007.
6. Balder ten Cate. *Model theory for extended modal languages.* PhD thesis, University of Amsterdam, 2005. ILLC Dissertation Series DS-2005-01.
7. Cédric Dégremont and Lena Kurzen. Modal logics for preferences and cooperation: Expressivity and complexity. In *Knowledge Representation for Agents and Multi-Agent Systems, Post-proceedings of KRAMAS 2008*, volume 5605 of *Lecture Notes in Artificial Intelligence.* Springer-Verlag, 2009.
8. Ulle Endriss, Nicolas Maudet, Fariba Sadri, and Francesca Toni. Negotiating socially optimal allocations of resources. *Journal of Artificial Intelligence Research*, 25:315–348, 2006.
9. Jelle Gerbrandy and Luigi Sauro. Plans in cooperation logic: a modular approach. In *Proceedings of the IJCAI Workshop on Nonmontonic Reasoning, Action and Change (NRAC 2007)*, Hyderabad (India), 2007.
10. Patrick Girard. *Modal Logic for Preference Change.* PhD thesis, Stanford, 2008.
11. Valentin Goranko and Govert van Drimmelen. Complete axiomatization and decidability of alternating-time temporal logic. *Theor. Comput. Sci.*, 353(1-3):93–117, 2006.
12. Marc Pauly. A modal logic for coalitional power in games. *JLC*, 12(1):149–166, 2002.
13. Stéphane Le Roux, Pierre Lescanne, and René Vestergaard. Conversion/preference games. *CoRR*, 2008.
14. Krister Segerberg. Bringing it about. *Journal of Philosophical Logic*, 18(4):327–347, 1989.
15. Dirk Walther, Wiebe van der Hoek, and Michael Wooldridge. Alternating-time temporal logic with explicit strategies. In *TARK '07: Proceedings of the 11th conference on theoretical aspects of rationality and knowledge*, pages 269 – 278. ACM, 2007.