Quantified Coalition Logic

Thomas Ågotnes
Faculty of Engineering
Bergen University College, Norway
Thomas.Agotnes@ii.uib.no

Wiebe van der Hoek **Michael Wooldridge**
Computer Science Department
The University of Liverpool, UK
{wiebe,mjw}@csc.liv.ac.uk

Abstract

We add a limited but useful form of quantification to Coalition Logic, a popular formalism for reasoning about cooperation in game-like multi-agent systems. The basic constructs of *Quantified Coalition Logic* (QCL) allow us to express properties as "there exists a coalition C satisfying property P such that C can achieve φ". We give an axiomatization of QCL, and show that while it is no more expressive than Coalition Logic, it is exponentially more succinct. The time complexity of QCL model checking for symbolic and explicit state representations is shown to be no worse than that of Coalition Logic. We illustrate the formalism by showing how to succinctly specify such social choice mechanisms as majority voting, which in Coalition Logic require specifications that are exponentially long in the number of agents.

1 Introduction

Game theoretic models of cooperation has proved to be a valuable source of techniques and insights for the field of multi-agent systems, and cooperation logics such as Alternating-time Temporal Logic (ATL) [Alur *et al.*, 2002] and Coalition Logic (CL) [Pauly, 2001] have proved to be powerful and intuitive knowledge representation formalisms for such games. Many interesting properties of cooperative scenarios require *quantification over coalitions*. However, existing cooperation logics provide no direct facility for such quantification, and expressing such properties therefore requires formulae that are exponentially long in the number of agents. Examples include expressing the notion of a *weak veto player* [Wooldridge and Dunne, 2004] in CL, or solution concepts from cooperative game theory such as nonemptyness of the core in Coalitional Game Logic [Ågotnes *et al.*, 2006]. An obvious solution would be to extend, for example, ATL, with a first-order-style apparatus for quantifying over coalitions. In such a quantified ATL, one might express the fact that agent i is a necessary component of every coalition able to achieve φ by the following formula:

$$\forall C : \langle\langle C\rangle\rangle\Diamond\varphi \rightarrow (i \in C)$$

However, adding quantification in such a naive way leads to undecidability over infinite domains (using basic quantificational set theory we can define arithmetic), and very high computational complexity even over finite domains. The question therefore arises whether we can add quantification to cooperation logics in such a way that we can express useful properties of cooperation in games *without* making the resulting logic too computationally complex to be of practical interest. Here, we answer this question in the affirmative.

We introduce *Quantified Coalition Logic* (QCL), by modifying the existing cooperation modalities of CL in order to enable quantification. In CL, the basic cooperation constructs are $\langle C\rangle\varphi$, meaning that coalition C can achieve φ[1]; these operators are in fact modal operators with a neighbourhood semantics. In QCL, we replace these operators with expressions $\langle P\rangle\varphi$ and $[P]\varphi$; here, P is a *predicate over coalitions*, and the two sentences express the fact that *there exists a coalition C satisfying property P such that C can achieve φ* and *all coalitions satisfying property P can achieve φ*, respectively. Thus we add a limited form of quantification to CL *without* the apparatus of quantificational set theory. *Our main contribution is the development of* QCL, *a logic that is equally expressive as* CL, *no worse than* CL *with respecto to the computational problem of model checking, but at the same time exponentially more succinct than* CL.

The remainder of the paper is structured as follows. After a brief review of CL, we introduce a language for expressing coalition predicates, and show that the satisfiability problem for this language is NP-complete. We then introduce QCL, and give its complete axiomatization. We show that while QCL is no more expressive than Coalition Logic, it is nevertheless exponentially more succinct, in a precise formal sense. We then extend the language of coalition predicates to talk about the cardinality of coalitions, and show that the corresponding completeness and succinctness results also hold for QCL over this extended language. We illustrate QCL by showing how it can be used to succinctly specify a social choice mechanism, which in CL requires specifications that are exponentially long in the number of agents; we then round off with some conclusions.

[1] We adopt a notation which is in line with that in ATL: we use $\langle C\rangle$ for 'there is a coalition C such that ...' where [Pauly, 2001] uses $[C]$, and we write $[C]$ for Pauly's $\langle C\rangle$.

2 Coalition Logic

Since QCL is based on Pauly's Coalition Logic CL [Pauly, 2001], we first briefly introduce the latter. CL is a propositional modal logic, containing an indexed collection of unary modal operators $\langle C\rangle$ and $[C]$, where C is a coalition, i.e., a subset of a given set of agents Ag. The intended interpretation of $\langle C\rangle\varphi$ is that C *can achieve* φ, or, that C *is effective for* φ. Formulae of CL are defined by the following grammar (with respect to a set Φ_0 of Boolean variables, and a fixed set Ag of agents):

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle C\rangle\varphi$$

where $p \in \Phi_0$ is an atomic proposition and C a subset of Ag. As usual, we use parentheses to disambiguate formulae where necessary, and define the remaining connectives of classical logic as abbreviations: $\bot \equiv \neg\top$, $\varphi \to \psi \equiv (\neg\varphi) \vee \psi$ and $\varphi \leftrightarrow \psi \equiv (\varphi \to \psi) \wedge (\psi \to \varphi)$.

A model $\mathcal{M}$ for (over Φ_0, Ag) is a triple $\mathcal{M} = \langle S, \mathcal{E}, \pi\rangle$ where

- $S = \{s_1, \ldots, s_o\}$ is a finite non-empty set of *states*;
- $\mathcal{E} : 2^{Ag} \times S \to 2^{2^S}$ is an *effectivity function*, where $T \in \mathcal{E}(C, s)$ is intended to mean that from state s, the coalition C can cooperate to ensure that the next state will be a member of T; and
- $\pi : S \to 2^{\Phi_0}$ is a valuation function.

Call the set of all models *general models*, denoted by $\mathcal{G}en$. It is possible to define a number of constraints on effectivity functions, depending upon exactly which kinds of scenario they are intended to model [Pauly, 2001, pp.24–39]. Unless stated otherwise, we will assume that our models $\mathcal{M}$ are *weak playability* models $\mathcal{M} \in \mathcal{WP}$, where effectivity functions are *outcome monotonic*, i.e., $\forall C \subseteq Ag, \forall s \in S, \forall X, Y \subseteq S$, if $X \in \mathcal{E}(C, s)$ and $X \subseteq Y$, then $Y \in \mathcal{E}(C, s)$, and moreover that effectivity functions satisfy Pauly's *weak playability* conditions [Pauly, 2001, p.30]. An *interpretation* for CL is a pair $\mathcal{M}, s$ where $\mathcal{M}$ is a model and s is a state in $\mathcal{M}$. The satisfaction relation "$\models_{CL}$" for CL holds between interpretations and formulae of CL. We say that coalition C *can enforce* φ in s if for some $T \in \mathcal{E}(C, s)$, φ is true in all $t \in T$. That is, C can make a choice such that, irrespective of the others' choices, φ will hold. Formally, the satisfaction relation is defined by the following inductive rule (we assume the cases for p, $\top$, negation and disjunction are clear):

> $\mathcal{M}, s \models_{CL} \langle C\rangle\varphi$ iff $\exists T \in \mathcal{E}(C, s)$ such that $\forall t \in T$, we have $\mathcal{M}, t \models_{CL} \varphi$.

The notions of truth of φ in a model ($\mathcal{M} \models_{CL} \varphi$) and validity in a class of models $\mathcal{C}$ ($\mathcal{C} \models_{CL} \varphi$) are defined as usual. The inference relation $\vdash_{CL}$ for CL is given in table 1 (taken from [Pauly, 2001], but adapted to our notation): it is sound and complete with respect to the class of weak playability models $\mathcal{WP}$ ([Pauly, 2001, p. 55]).

Prop	$\vdash_{CL} \psi$
$Ag\bot$	$\vdash_{CL} \neg\langle Ag\rangle\bot$
$\top$	$\vdash_{CL} \neg\langle\emptyset\rangle\bot \to \langle C\rangle\top$
$\bot$	$\vdash_{CL} \langle C\rangle\bot \to \langle C'\rangle\bot$
Ag	$\vdash_{CL} \neg\langle\emptyset\rangle\neg\varphi \to \langle Ag\rangle\varphi$
S	$\vdash_{CL} (\langle C_1\rangle\varphi_1 \wedge \langle C_2\rangle\varphi_2) \to \langle C_1 \cup C_2\rangle(\varphi_1 \wedge \varphi_2)$
MP	$\vdash_{CL} \varphi \to \psi,\ \vdash_{CL} \varphi\ \Rightarrow \vdash_{CL} \psi$
Distr	$\vdash_{CL} \varphi \to \psi\ \Rightarrow \vdash_{CL} \langle C\rangle\varphi \to \langle C\rangle\psi$

Table 1: Axioms and Rules for Coalition Logic. In (*Prop*), ψ is a propositional tautology, in axiom ($\bot$), we require $C' \subseteq C$, and for (S), $C_1 \cap C_2 = \emptyset$.

3 Quantified Coalition Logic

If we have n agents in Ag, and one wants to express that *some* coalition can enforce some atomic property p, one needs to enumerate 2^n disjunctions of the form $\langle C\rangle p$. The idea behind Quantified Coalition Logic (QCL) is to avoid this blow-up in the length of formulas. Informally, QCL is a propositional modal logic, containing an indexed collection of unary modal operators $\langle P\rangle\varphi$ and $[P]\varphi$. The intended interpretation of $\langle P\rangle\varphi$ is that *there exists a set of agents* C*, satisfying predicate* P*, such that* C *can achieve* φ. We refer to expressions P as *coalition predicates*, and we now define a language for coalition predicates; QCL will then be parameterised with respect to such a language. Of course, many coalition predicate languages are possible, with different properties, and later we will investigate another such language. Throughout the remainder of this paper, we will assume a fixed, finite set Ag of agents.

Coalition Predicates: Syntactically, we introduce two atomic predicates $subseteq$ and $supseteq$, and derive a stock of other predicate forms from these. Formally, the syntax of coalition predicates is given by the following grammar:

$$P ::= subseteq(C) \mid supseteq(C) \mid \neg P \mid P \vee P$$

where $C \subseteq Ag$ is a set of agents. One can think of the atomic predicates $subseteq(C)$ and $supseteq(C)$ as a stock of $2^{|Ag|+1}$ propositions, one for each coalition, which are then to be evaluated in a given coalition Co. The circumstances under which a concrete coalition Co satisfies a coalition predicate P, are specified by a satisfaction relation "$\models_{cp}$", defined by the following four rules:

$Co \models_{cp} subseteq(C)$ iff $Co \subseteq C$

$Co \models_{cp} supseteq(C)$ iff $Co \supseteq C$

$Co \models_{cp} \neg P$ iff not $Co \models_{cp} P$

$Co \models_{cp} P_1 \vee P_2$ iff $Co \models_{cp} P_1$ or $Co \models_{cp} P_2$

Now we can be precise about what it means that "a coalition Co satisfies P": it just means $Co \models_{cp} P$. We will assume the conventional definitions of implication ($\to$), biconditional ($\leftrightarrow$), and conjunction ($\wedge$) in terms of $\neg$ and $\vee$.

Coalitional predicates $subseteq(\cdot)$ and $supseteq(\cdot)$ are in fact not independent. They are mutually definable – due to the fact that the set of all agents Ag is assumed to be finite. We then have that ([Ågotnes and Walicki, 2006]) $subseteq(C) \equiv \bigwedge_{i \in Ag \setminus C} \neg supseteq(\{i\})$ and $supseteq(C) \equiv \bigwedge_{C' \subseteq Ag, C \not\subseteq C'} \neg subseteq(C')$. The reason that we include both types of predicates as primitives is a main motivating factor of this paper: we are interested in *succinctly* expressing quantification in coalition logic.

We find it convenient to make use of the following derived predicates:

$$\begin{array}{rcl}
eq(C) & \equiv & subseteq(C) \wedge supseteq(C) \\
subset(C) & \equiv & subseteq(C) \wedge \neg eq(C) \\
supset(C) & \equiv & supseteq(C) \wedge \neg eq(C) \\
incl(i) & \equiv & supseteq(\{i\}) \\
excl(i) & \equiv & \neg incl(i) \\
any & \equiv & supseteq(\emptyset) \\
nei(C) & \equiv & \bigvee_{i \in C} incl(i) \\
ei(C) & \equiv & \neg nei(C)
\end{array}$$

The reader may note an obvious omission here: we have not introduced any explicit way of talking about the *cardinality* of coalitions; such predicates will be discussed in Section 4.

We say that a coalition predicate P is *Ag-consistent* if for some $Co \subseteq Ag$, we have $Co \models_{cp} P$, and P is *Ag-valid* if $Co \models_{cp} P$ for all $Co \subseteq Ag$.

The *model checking problem for coalition predicates* is the problem of checking whether, for given Co and P, we have $Co \models_{cp} P$ [Clarke *et al.*, 2000]. It is easy to see that this problem is decidable in polynomial time. The *satisfiability problem for coalition predicates* is the problem of deciding whether P is consistent. We get the following.

Theorem 1 *The satisfiability problem for coalition predicates is* NP*-complete.*

Quantified Coalition Logic: We now present QCL. Its formulae are defined by the following grammar:

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle P \rangle \varphi \mid [P]\varphi$$

Models for CL and QCL are the same. The satisfaction relation for the new operators is as follows.

> $\mathcal{M}, s \models_{QCL} \langle P \rangle \varphi$ iff $\exists C \subseteq Ag$: $C \models_{cp} P$ and $\exists T \in \mathcal{E}(C, s)$ such that $\forall t \in T$, we have $\mathcal{M}, t \models_{QCL} \varphi$.
>
> $\mathcal{M}, s \models_{QCL} [P]\varphi$ iff $\forall C \subseteq Ag$: $C \models_{cp} P$ implies $\exists T \in \mathcal{E}(C, s)$ such that $\forall t \in T$, $\mathcal{M}, t \models_{QCL} \varphi$.

Readers familiar with modal logic may wonder why we did not introduce the universal coalition modality $[P]\varphi$ as the dual $\neg\langle P \rangle \neg\varphi$. In fact such a definition would not serve the desired purpose. Consider the pattern of quantifiers in the semantics of $\langle\cdot\rangle$: $\exists\exists\forall$. Taking the dual $\neg\langle\cdot\rangle\neg$ would yield the quantifiers $\forall\exists\exists$, rather than the desired $\forall\exists\forall$ pattern. Of course, this does not mean that $[P]\varphi$ is not definable from $\langle P \rangle \varphi$ (and the propositional connectives) in some *other* way. In fact:

$$[P]\varphi \equiv \bigwedge_{\{C \mid C \models_{pc} P\}} \langle eq(C) \rangle \varphi$$

Thus, for expressiveness, $\langle C \rangle$ together with the propositionals are adequate connectives, and $[P]\varphi$ is definable. The reason we introduce the box cooperation modality as a separate construct is one of the main motivations in this paper, as discussed before for the different predicate operators: succinctness of expression.

Example QCL Expressions: To get a flavour of the kind of properties we can express in QCL, we present some example QCL formulae. First, note that the conventional CL/ATL ability expression is defined simply as: $\langle C \rangle \varphi \equiv \langle eq(C) \rangle \varphi$. We can also succinctly express properties such as the solution concepts from *Qualitative Coalitional Games* [Wooldridge and Dunne, 2004]. For example, a *weak veto player* for φ is an agent that must be present in any coalition that has the ability to bring about φ: $WVETO(C, \varphi) \equiv \neg\langle excl(i) \rangle \varphi$. Of course, if *no* coalition has the ability to achieve φ, then this means that *every* agent is a veto player for φ. A *strong veto player* for φ is thus an agent that is both a weak veto player for φ and that is a member of *some* coalition that can achieve φ: $VETO(i, \varphi) \equiv WVETO(i, \varphi) \wedge \langle incl(i) \rangle \varphi$. A coalition C is *weakly minimal for* φ if no subset of C can achieve φ: $WMIN(C, \varphi) \equiv \neg\langle subset(C) \rangle \varphi$. And C are simply *minimal* if they are weakly minimal and also able to bring about φ: $MIN(C, \varphi) \equiv \langle C \rangle \varphi \wedge WMIN(C, \varphi)$. Finally, $GC(C)$ says that C is the *grand* coalition: $GC(C) \equiv [supset(C)]\bot \wedge \neg\langle subseteq(C) \rangle \bot$.

Model Checking: *Model checking* is currently regarded as perhaps the most important computational problem associated with any temporal/modal logic, as model checking approaches for such logics have had a substantial degree of success in industry [Clarke *et al.*, 2000]. The *explicit state model checking problem* for QCL is as follows:

> Given a model $\mathcal{M}$, state s in $\mathcal{M}$, and formula φ of QCL, is it the case that $\mathcal{M}, s \models_{QCL} \varphi$?

Notice that in this version of the problem, we assume that the components of the model $\mathcal{M}$ are *explicitly enumerated* in the input. It is known that the corresponding problem for Coalition Logic may be solved in polynomial time $O(|\mathcal{M}| \cdot |\varphi|)$ [Pauly, 2001, p.50] (as may the explicit state ATL model checking problem [Alur *et al.*, 2002]). Perhaps surprisingly, the QCL model checking problem is no worse:

Theorem 2 *The explicit state model checking problem for* QCL *may be solved in polynomial time.*

Of course, this result is not terribly useful, since it assumes a representation of $\mathcal{M}$ that is not feasible, since it is exponentially large in the number of agents and Boolean variables in the system. Implemented model checkers use *succinct* languages for defining models; for example, the REACTIVE MODULES LANGUAGE (RML) of Alur et al [Alur and Henzinger, 1999]. Assuming an RML representation, Coalition Logic model checking is PSPACE-complete [Hoek *et al.*, 2006], and thus no easier than theorem proving in the same logic [Pauly, 2001, p.60]. It is therefore more meaningful to ask what the model checking complexity of QCL is for such a representation. We only give a very brief summary of RML – space restrictions prevent a complete description; see [Alur and Henzinger, 1999; Hoek *et al.*, 2006] for details.

In REACTIVE MODULES, a system is specified as a collection of *modules*, which correspond to agents. Here is a (somewhat simplified) example of an RML module:

```
            module $toggle$ controls $x$
init        [] $\top \rightsquigarrow x' := \top$
            [] $\top \rightsquigarrow x' := \bot$
update      [] $x \rightsquigarrow x' := \bot$
            [] $(\neg x) \rightsquigarrow x' := \top$
```

This agent *toggle*, controls a single Boolean variable, x. The *choices* available to the agent at any given time are defined by the `init` and `update` rules. The `init` rules define the choices available to the agent with respect to the initialisation of its variables, while the `update` rules define the agent's choices subsequently. The `init` rules define two choices for the initialisation of this variable: assign it the value $\top$ or the value $\bot$. Both of these rules can fire initially, as their conditions ($\top$) are always satisfied; in fact, only one of the available rules will ever *actually* fire, corresponding to the "choice made" by the agent on that decision round. With respect to `update` rules, the first rule says that if x has the value $\top$, then the corresponding choice is to assign it the value $\bot$, while the second rule 'does the opposite'. In other words, the agent non-deterministically chooses a value for x initially, and then on subsequent rounds toggles this value.

The following can be proved by an adaption and extension of the proof in [Hoek *et al.*, 2006].

Theorem 3 *The model checking problem for* QCL *assuming an* RML *representation for models is* PSPACE*-complete.*

This result, we believe, is potentially much more interesting than that for explicit state model checking, since it tells us that QCL model checking is no more complex than Coalition Logic *even for a realistic representation of models*.

Expressive Power: We now argue that QCL is equivalent in expressive power to Coalition Logic. To begin, consider the following translation τ from QCL formulae to CL formulae. For atoms p and $\top$, τ is the identity, and it distributes over disjunction, and moreover:

$$\begin{array}{lcl} \tau(\langle P\rangle\varphi) & = & \bigvee_{\{C \mid C \models_{pc} P\}} \langle C\rangle\tau(\varphi) \\ \tau([P]\varphi) & = & \bigwedge_{\{C \mid C \models_{pc} P\}} \langle C\rangle\tau(\varphi) \end{array}$$

We already know from above that we have a translation in the other direction: let us call it δ, with defining clause $\delta(\langle C\rangle\varphi) = \langle eq(C)\rangle\delta(\varphi)$.

As an example, suppose $Ag = \{a, b, c\}$ and let $P = (supset(\{a\}) \vee supset(\{b\}) \vee supset(\{c\})) \wedge \neg eq(\{a, b, c\})$. Now, consider the QCL formula $\psi = \langle P\rangle q$. Then $\tau(\psi) = \langle\{a, b\}\rangle q \vee \langle\{a, c\}\rangle q \vee \langle\{b, c\}\rangle q$ while $\delta(\tau(\psi)) = \langle eq(\{a, b\})\rangle q \vee \langle eq(\{a, c\})\rangle q \vee \langle eq(\{b, c\})\rangle q$. Hence, one can think of $\delta(\tau(\varphi))$ as a *normal form* for φ, where the only coalition predicate in φ is eq. That QCL and CL have equal expressive power follows from the fact that the two translations preserve truth.

Theorem 4 *Let $\mathcal{M}$ be a model, and s a state, and let φ be a* QCL *formula, and ψ a* CL *formula. Then:*

1. $\mathcal{M}, s \models_{QCL} \varphi$ *iff* $\mathcal{M}, s \models_{CL} \tau(\varphi)$
2. $\mathcal{M}, s \models_{CL} \psi$ *iff* $\mathcal{M}, s \models_{QCL} \delta(\psi)$

$P0$	$\vdash_{cp} supseteq(\emptyset)$
$P1$	$\vdash_{cp} supseteq(C) \wedge supseteq(C')$ $\leftrightarrow supseteq(C \cup C')$
$P2$	$\vdash_{cp} supseteq(C) \to \neg subseteq(C')$
$P3$	$\vdash_{cp} subseteq(C \cup \{a\}) \wedge \neg supseteq(a)$ $\to subseteq(C)$
$P4$	$\vdash_{cp} subseteq(C) \to subseteq(C')$
$Prop$	$\vdash_{cp} \psi$
MP	$\vdash_{cp} \varphi \to \psi, \vdash_{cp} \varphi \Rightarrow \vdash_{cp} \psi$
δAx	$\vdash_{QCL} \delta(Ax)$
$\delta\langle\rangle$	$\vdash_{QCL} \langle P\rangle\varphi \leftrightarrow$ $\bigvee_{\{C \mid C \vdash_{cp} P\}} \langle eq(C)\rangle\varphi$
$\delta[]$	$\vdash_{QCL} [P]\varphi \leftrightarrow$ $\bigwedge_{\{C \mid C \vdash_{cp} P\}} \langle eq(C)\rangle\varphi$
δR	$\delta(R)$

Table 2: Axioms and Rules for Quantified Coalition Logic. The condition of $P2$ is $C \not\subseteq C'$, for $P4$ it is $C \subseteq C'$, ψ in $Prop$ is a propositional tautology; Ax in δAx is any CL-axiom, R in δR is any CL-rule

Axiomatization: The translations introduced above provide the key to a complete axiomatization of QCL. First, recall Pauly's axiomatization of Coalition Logic (Table 1). Given this, and the translations defined previously, we obtain an axiom system for QCL-formulae as follows. First, QCL includes the δ translation of all the CL axioms and rules, and axioms that state that the δ-translation is correct: see the lower part of Table 2. On top of that, QCL is *parametrised* by an inference relation $\vdash_{cp}$ for coalition predicates. The axioms for this in Table 2 are taken from [Ågotnes and Walicki, 2006].

Theorem 5

1. *([Ågotnes and Walicki, 2006]) $\vdash_{cp}$ is sound and complete: for any P, $\models_{cp} P \quad \Leftrightarrow \quad \vdash_{cp} P$*
2. *For any* CL *formula φ, $\vdash_{CL} \varphi \quad \Rightarrow \quad \vdash_{QCL} \delta(\varphi)$*
3. *Let φ be any* QCL *formula. Then $\vdash_{QCL} \varphi \leftrightarrow \delta(\tau(\varphi))$ and, in particular, $\vdash_{QCL} \varphi$ iff $\vdash_{QCL} \delta(\tau(\varphi))$.*

Theorem 6 (Completeness and Soundness) *Let φ be an arbitrary* QCL*-formula. Then: $\vdash_{QCL} \varphi$ iff $\models_{QCL} \varphi$*

Examples of derivable properties include:

$$\begin{array}{ll} \models_{QCL} [P_1]\varphi \to [P_2]\varphi & \text{when } \models_{cp} P_1 \to P_2 \\ \models_{QCL} ([P_1]\varphi \wedge [P_2]\varphi) \to [P_1 \vee P_2]\varphi & \end{array}$$

These illustrate that we not only have primitive modal operators, but also some kind of operations over them, like negation and conjunction. This of course is very reminiscent of Boolean modal logic, where one studies algebraic operations like complement, meet and join on modal operators [Gargov and Passy, 1987]. We will not pursue the details of the connection here.

Succinctness: Theorem 4 tells us that the gain of QCL over CL is not its expressivity. Rather, the advantage of QCL is in its *succinctness* of representation. For example, for the QCL formula $\langle any\rangle q$, the translated CL formula $\tau(\langle any\rangle q)$ is exponentially longer, since it has to explicitly enumerate all coalitions in Ag. Is it however *generally* the case that $\tau(\varphi)$ is shorter than φ? Since the translation does some computations under $\models_{cp}$, this is in general not the case. For instance, if $P = supseteq(\{a\}) \wedge supseteq(\{c\}) \wedge supseteq(\{b\}) \wedge (subseteq(\{a,b,c\}) \vee subseteq(\{a,b,d\}))$, then $\psi = \langle P\rangle q$ would have as a τ-translation $\langle\{a,b,c\}\rangle q$, which is shorter than the original QCL-formula ψ. But then again, $\delta(\tau(\psi))$ is a QCL formula that is equivalent to ψ, but that has a size similar to $\tau(\psi)$.

To make this all precise, let us define the length $\ell(\varphi)$ of both QCL and CL formulas φ, as follows:

$$\begin{array}{lcl}
\ell(\top) = \ell(p) & = & 1 \\
\ell(\varphi_1 \vee \varphi_2) & = & \ell(\varphi_1) + \ell(\varphi_2) + 1 \\
\ell(\neg\varphi) & = & \ell(\varphi) + 1 \\
\ell(\langle P\rangle\varphi) = \ell[P]\varphi) & = & predsize(P) + \ell(\varphi) \\
\ell(\langle C\rangle\varphi) & = & coalsize(C) + \ell(\varphi)
\end{array}$$

with

$$\begin{array}{lcl}
predsize(subseteq(C)) & = & coalsize(C) + 1 \\
predsize(supseteq(C)) & = & coalsize(C) + 1 \\
predsize(\neg P) & = & predsize(P) + 1 \\
predsize(P_1 \vee P_2) & = & predsize(P_1) + predsize(P_2) + 1 \\
coalsize(C) & = & \mid C \mid
\end{array}$$

Let φ and ψ be X and Y formulas, respectively, where X and Y both range over CL and QCL. Then we say that they are equivalent with respect to some class of models if they have the same satisfying pairs $\mathcal{M}, s$, that is, for each $\mathcal{M}, s$ in the class of models it is the case that $\mathcal{M}, s \models_X \varphi$ iff $\mathcal{M}, s \models_Y \psi$. This definition naturally extends to sets of formulas.

In the following theorem we show that QCL is *exponentially more succinct* than CL, over general models. This notion of relative succinctness is taken from [Lutz, 2006], who demonstrates that public announcement logis is more succinct than epistemic logic.

Theorem 7
There is an infinite sequence of distinct QCL formulas $\varphi_0, \varphi_1, \ldots$ such that, not only is the CL formula $\tau(\varphi_i)$ equivalent to φ_i for every $i \geq 0$, but *every* CL formula ψ_i that is equivalent, with respect to general models, to φ_i has the property $\ell(\psi_i) \geq 2^{|\varphi_i|}$.

4 Coalition Size

As we noted earlier, an obvious omission from our language of coalition predicates is designated predicates for expressing cardinality properties of coalitions. In this section, we explore extensions to the framework for this purpose. The obvious approach is to introduce primitive coalition predicates $geq(n)$, where $n \in \mathbb{N}$, with semantics as follows:

$$C \models geq(n) \text{ iff } |C| \geq n$$

Given this predicate, we can define several obvious derived predicates (see also [Ågotnes and Alechina, 2006] for a discussion of a similar language).

$$\begin{array}{rcl}
gt(n) & \equiv & geq(n+1) \\
lt(n) & \equiv & \neg geq(n) \\
leq(n) & \equiv & lt(n+1) \\
maj(n) & \equiv & geq(\lceil (n+1)/2 \rceil) \\
ceq(n) & \equiv & (geq(n) \wedge leq(n))
\end{array}$$

The first natural question is whether $geq(n)$ is definable in QCL. Indeed it is:

$$geq(n) \equiv \bigvee_{C \subseteq Ag, |C| \geq n} supseteq(C) \qquad (1)$$

However, we again see that such a definition leads to exponentially large formulae, which justifies extending the predicate language of QCL with an *atomic* coalition predicate $geq(n)$ for every $n \in \mathbb{N}$. Call the resulting logic QCL($\geq$), and let $\models_{cp\geq}$ and $\models_{QCL(\geq)}$ denote the satisfiability relations for QCL($\geq$) predicates and QCL($\geq$) formulae, respectively. Once again, the gain is not expressiveness but *succinctness*. As another example of the added succinctness, consider the CL formula $\langle C\rangle p$. In QCL this cannot in general be written by any less complex formula than $\langle subseteq(C) \wedge supseteq(C)\rangle p$, but in QCL($\geq$) it can be simplified somewhat to $\langle supseteq(C) \wedge \neg geq(|C|+1)\rangle p$ (which in general is simpler since one of the enumerations of the agents in C is replaced by a number).

A subtle but important issue when reasoning with the logic is the way in which the natural number argument of the $geq(\ldots)$ predicate is *represented*. Suppose, (following standard practice in complexity theory), that we represent the argument in binary. Now, we ask whether a given coalition predicate P is satisfiable, where P contains a constraint $geq(n)$. Now checking the satisfiability of such constraints is not obviously in NP. The problem is that the witness C to the satisfiability of P is *exponentially larger than the constraint* $geq(n)$. Of course, if we express the natural number n in *unary*, then this is not an issue. But unary is not a realistic or practical representation for numbers. It turns out, however, that we do in fact get NP completeness for the satisfiability problem also for QCL($\geq$), although the argument requires some more work. The reason is that we can use an efficient encoding of the witness C. This was shown by [Ågotnes and Alechina, 2006] for a similar problem (cf. Section 5).

Let $Ag(P)$ and $subp(P)$ denote the set of agents, and the set of sub-predicates, respectively, occurring in a predicate P.

Lemma 1 *Any satisfiable* QCL*($\geq$) predicate P is satisfied by a coalition consisting of no more than $1 + max_P$ agents, where $1 + max_P$ equals*

$$max(\{|Ag(P)|, max(\{geq(n) : geq(n) \in subp(P)\})\})$$

Theorem 8 *The satisfiability problem for* QCL*($\geq$) coalition predicates is* NP*-complete.*

It is straightforward to lift the translation τ from QCL to CL to the case when also the additional predicates of QCL($\geq$) are allowed, and it is easy to see that Theorem 4 holds also for QCL($\geq$) formulae. For axiomatisation, we only need to add axioms for the $geq(n)$ predicates to the predicate calculus. That can be achieved simply by adding (1) as an axiom

$(MIN0)$	$\vdash_{cp} geq(0)$
$(MIN1)$	$\vdash_{cp} geq(n) \rightarrow geq(m) \;\; (m < n)$
$(MIN2)$	$\vdash_{cp} supseteq(\{a_1\}) \wedge \cdots \wedge supseteq(\{a_k\})$
	$\rightarrow geq(k) \;\; \forall i \neq j \; a_i \neq a_j$

Table 3: Extra predicate calculus axioms for QCL(≥).

schema. A more "direct" axiomatisation of $geq(n)$ is shown in Table 3, taken from [Ågotnes and Alechina, 2006]. Let $\vdash_{cp\geq}$ denote derivability in the QCL predicate calculus (from Table 2) extended with the axioms in Table 3. The following is easily obtained from a similar result in [Ågotnes and Alechina, 2006]:

Lemma 2 *The* QCL*(≥) predicate calculus is sound and complete: for any* QCL*(≥) predicate P, $\models_{cp\geq} P \quad \Leftrightarrow \vdash_{cp\geq} P$*

Let $\vdash_{QCL(\geq)}$ denote derivability in the system obtained by replacing $\vdash_{cp}$ with $\vdash_{cp\geq}$ in the definition of $\vdash_{QCL}$ (Table 2).

Theorem 9 (Completeness and Soundness) *Let φ be a* QCL*(≥)-formula. Then: $\vdash_{QCL(\geq)} \varphi$ iff $\models_{QCL(\geq)} \varphi$*

To illustrate the use of QCL(C) for reasoning about multi-agent systems, consider the expression of *majority voting*:

> An electorate of n voters wishes to select one of two outcomes ω_1 and ω_2. They want to use a simple majority voting protocol, so that outcome ω_i will be selected iff a majority of the n voters state a preference for it. No coalition of less than majority size should be able to select an outcome, and *any* majority should be able to choose the outcome (i.e., the selection procedure is not influenced by the "names" of the agents in a coalition). One outcome must be selected, but both outcomes should not be selected simultaneously.

We express these requirements as follows. First: *any majority should be able to select an outcome*.

$$([maj(n)]\omega_1) \wedge ([maj(n)]\omega_2)$$

No coalition that is not a majority can select an outcome.

$$(\neg\langle\neg maj(n)\rangle\omega_1) \wedge (\neg\langle\neg maj(n)\rangle\omega_2)$$

Either outcome ω_1 or ω_2 must result.

$$\langle any\rangle(\omega_1 \vee \omega_2)$$

Both outcomes cannot be selected simultaneously.

$$\langle any\rangle\neg(\omega_1 \wedge \omega_2)$$

Notice that majority voting cannot be succinctly specified using regular Coalition Logic.

5 Related Work and Conclusions

Quantified Coalition Logic adds a limited but useful form of quantification to Coalition Logic, which is computationally tractable. The motivation is succinctness rather than expressiveness: QCL is exponentially more expressive than CL.

While first-order temporal logics have been studied in the literature, and CL can be seen as the next-time fragment of ATL which again is a generalisation of the branching-time temporal logic Computational Tree Logic (CTL), we are not aware of any other works on quantification in CL or ATL. Lately, there has been some work on generalising the coalition modalities in another direction: to explicitly include actions and strategies [van der Hoek *et al.*, 2005; Ågotnes, 2006].

Opportunities for future work include a more detailed understanding of the relationship between QCL and Boolean modal logic.

References

[Ågotnes and Alechina, 2006] T. Ågotnes and N. Alechina. Knowing minimum/maximum n formulae. In *Proc. ECAI-2006*, 2006.

[Ågotnes and Walicki, 2006] T. Ågotnes and M. Walicki. Complete axiomatizations of finite syntactic epistemic states. In *DALT III:*, vol 3904 of *LNCS/LNAI*, pages 33 – 50. Springer, 2006.

[Ågotnes *et al.*, 2006] T. Ågotnes, W. van der Hoek, and M. Wooldridge. On the logic of coalitional games. In *Proc. AAMAS-2006*, pages 153–160, 2006.

[Ågotnes, 2006] T. Ågotnes. Action and knowledge in alternating-time temporal logic. *Synthese*, 149(2):375–407, 2006.

[Alur and Henzinger, 1999] R. Alur and T. A. Henzinger. Reactive modules. *Formal Meth. in Sys. Design*, 15(11):7–48, July 1999.

[Alur *et al.*, 2002] R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *JACM*, 49(5):672–713, September 2002.

[Clarke *et al.*, 2000] E. M. Clarke, O. Grumberg, and D. A. Peled. *Model Checking*. MIT Press, 2000.

[Gargov and Passy, 1987] G. Gargov and S. Passy. A note on Boolean Modal Logic. In *Mathematical Logic and Applications*, pages 253–263. Plenum Press, 1987.

[Hoek *et al.*, 2006] W. van der Hoek, A. Lomuscio, and M. Wooldridge. On the complexity of practical ATL model checking. In *Proc. AAMAS-2006*, pages 201–208, 2006.

[Lutz, 2006] C. Lutz. Complexity and succinctness of public announcement logic. In *Proc. AAMAS-2006*, pages 137–144, 2006.

[Pauly, 2001] M. Pauly. *Logic for Social Software*. PhD thesis, University of Amsterdam, 2001.

[van der Hoek *et al.*, 2005] W. van der Hoek, W. Jamroga, and M. Wooldridge. A logic for strategic reasoning. In *Proceedings of AAMAS-2005*, pages 157–153, 2005.

[Wooldridge and Dunne, 2004] M. Wooldridge and P. E. Dunne. On the computational complexity of qualitative coalitional games. *Artificial Intelligence*, 158(1):27–73, 2004.