Bounded Model Checking Knowledge and Branching Time in Synchronous Multi-agent Systems

Xiangyu Luo†, Kaile Su‡, Abdul Sattar*, Qingliang Chen†, Guanfeng Lv†
†Department of Computer Science, Sun Yat-sen University, Guangzhou, P. R. China
‡Institute for Integrated and Intelligent Systems, Griffith University, Brisbane, Australia
xiang.yu.luo@yahoo.com.cn

ABSTRACT

We present an approach to the verification of temporal epistemic properties in synchronous multi-agent systems (MAS) via bounded model checking (BMC). Based on the semantics of synchronous interpreted systems, we extend the temporal logic $\text{CTL}^*$ by incorporating epistemic modalities and obtain the so-called temporal epistemic logic $\text{CTL}^*K$. Though $\text{CTL}^*K$ is of great expressive power in both temporal and epistemic dimensions, we show that BMC method is still applicable for the universal fragment of $\text{CTL}^*$. We present in some detail a BMC algorithm by using the semantics of synchronous interpreted system. In our approach, agents’ knowledge interpreted in synchronous semantics can be skillfully attained by the state position function, which avoids extending the encoding of the states and the transition relation of the plain temporal epistemic model for time domain.

Categories and Subject Descriptors
I.2.11 [Artificial Intelligence]: DAI-Multiagent systems

General Terms
Theory, Verification

Keywords
Bounded model checking, temporal epistemic logic, bounded semantics, translation to SAT

1. INTRODUCTION

Model checking is a popular technique for automatic formal verification of finite state systems. Recently, verification of MAS has become an active field of research. In the multi-agent paradigm, particular emphasis is given to the formal representation of the mental attitudes of agents, such as agents’ knowledge, beliefs, desires, intentions and so on. However, the formal specifications of the traditional model checking are most commonly expressed as a formula of temporal logics such as $\text{CTL}$ and $\text{LTL}$. So, the research of MAS verification has focused on the extension of traditional model checking to incorporate epistemic modalities for describing information and motivation attitudes of agents.

In order to overcome the state explosion problem of the BDD-based symbolic model checking, we adopt bounded model checking (BMC) in this paper. The basic idea of BMC is to explore a part of the model sufficient to check a particular formula and translate the existential model checking problem over the part of the model into a test of propositional satisfiability.

The aim of this paper is to develop a BMC method for an expressive logic, called $\text{ACTLK}$, which incorporates epistemic modalities into $\text{ACTL}^*$ (the universal fragment of $\text{CTL}^*$). The significance of $\text{ACTLK}$ is that the temporal expressive power of $\text{ACTLK}$ is greater than that of $\text{ACTL}K$ [3]. For example, we permit the subformula of an epistemic formula to be a $\text{state or path}$ formula, while $\text{ACTLK}$ only subsumes state formulas. It is convenient to use $\text{ACTLK}$ to specify and verify dynamic knowledge of agents in dynamic environments.

2. LOGIC $\text{CTL}^*K$ AND ITS SUBSETS

In this paper, we extend the temporal logic $\text{CTL}^*$ by incorporating epistemic operators, which include $K_i$ (knows), $D_i$ (distributed knowledge), $E_i$ (everyone knows) and $C_i$ (common knowledge), where $i \in A$, $\Gamma \subseteq A$ and $A$ is a set of agents. To solve the existential model checking problem, we add four dual epistemic operators related to the operators mentioned above. If $Y \in \{K_i, D_i, E_i, C_i\}$ and $\varphi$ is a formula, then $\overline{Y}$ is the dual operator of $Y$ and $Y \varphi \equiv \overline{Y} \overline{\varphi}$. We call the resulting logic $\text{CTL}^*K$.

Here we adopt the synchronous interpreted systems semantics [1], which assumes that the agents have perfect recall or the agents have access to a shared clock and run in synchrony. So each agent always “knows” the time. Formally, for all agents $i$ and all points $(r, n)$ and $(r', n')$, if $(r, n) \sim_i (r', n')$, then $n = n'$, where the indistinguishability relation $(r, n) \sim_i (r', n')$ indicates that the $i$-local state $r_i(n)$ is equal to $r'_i(n')$.

Then, the epistemic relations used by $K_i, D_i, E_i$ and $C_i$ are defined as $\sim_i = \bigcap_{r \in \Gamma} \sim_i = \bigcup_{r \in \Gamma} \sim_i = \overline{f}$, the transitive closure of $\sim_i$, respectively. We only define the synchronous semantics of epistemic operators as follows:

$$(r, n) \models Y \varphi \ i f \ there \ is \ a \ run \ r' \ and \ time \ n' \ with \ (r, n) \sim (r', n') \ \ and \ n = n' \ such \ that \ (r', n') \models \varphi.$$

where $(Y, \sim) \in \{(K_i, \sim_i), (D_i, \sim_i), (E_i, \sim_i), (C_i, \sim_i)\}.

The $\text{ECTLK}$ logic is the restriction of $\text{CTL}^*K$ such that the negation can be applied only to propositions. The $\text{ECTLK}$ logic is also the restriction of $\text{CTL}^*K$ such that its language is defined as $\{\sim \varphi | \varphi \in \text{ECTLK}\}$. 
3. BOUNDED SEMANTICS OF ECTL* K

In this section we present the bounded semantics for ECTL* K in synchronous interpreted systems. It is an extension of the method presented in [2]. The main idea of the BMC method is that the validity of an ECTL* K formula $\varphi$ can be determined by checking the satisfiability of a propositional formula $[M, \varphi]_k = [M^{(\varphi^{[c_0]})}]_k$, where $[\varphi]_k$ is a number of constraints that must be satisfied on

$M_k$ for $\varphi$ to be satisfied, and $[M^{(\varphi^{[c_0]})}]_k$ represents the (partial) model $M_k$ under consideration, which consists of a part of valid $k$-paths in $M_k$. Definition 4.7 of [2] plus the following functions determine the number of those $k$-paths is sufficient for checking formula $\varphi$, such that the validity of $\varphi$ in $M_k$ is equivalent to the validity of $\varphi$ in the part of $M_k$.

Once $[M, \varphi]_k$ is constructed, the validity of formula $\varphi$ over $M_k$ can be determined by checking the satisfiability of the propositional formula $[M, \varphi]_k$ via a SAT solver. Thus, the BMC problem for ECTL* K ($M \models_k \varphi$) is translated into a propositional satisfiability problem. We give the BMC algorithm for ECTL* K as follows: let $\varphi \not\models \psi$ if $\psi$ is an ECTL* K formula. Then, start with $k := 1$, test the satisfiability of $[M, \varphi]_k \models_k [M^{(\varphi^{[c_0]})}]_k$ via a SAT solver, and increase $k$ by one either until $[M, \varphi]_k$ becomes satisfiable or $k$ reaches $|M| : [\varphi] : 2^{|\varphi|}$.

The translation for $[M^{(\varphi^{[c_0]})}]_k$ is omitted here because it is similar to that of [2]. Now we give some details of the translations for $[M, \varphi]_k$, i.e. $[\varphi]_k$.

According to our BMC method, we are currently implementing $\text{ECTL}^\ast$-model checking in NuSMV. We are keen to explore how sophisticatedly SAT solvers developed in constraint satisfaction community can be used for BMC. In addition, we are also interested in the extension of our approach to permit agents to have perfect recall [1]. Furthermore, in order to overcome the intrinsic limitation of BMC techniques, we will extend our BMC method to unbounded model checking for the full CTL* K language.

5. CONCLUSIONS

According to our BMC method, we are currently implementing the $\text{ACTL}^\ast$-K bounded model checker (MCTK), which is an extension of the $\text{LTL}$ BMC modules in NuSMV. We are keen to explore how sophisticatedly SAT solvers developed in constraint satisfaction community can be used for BMC. In addition, we are also interested in the extension of our approach to permit agents to have perfect recall [1]. Furthermore, in order to overcome the intrinsic limitation of BMC techniques, we will extend our BMC method to unbounded model checking for the full CTL* K language.

6. REFERENCES

