Optimality in Abstractions of Model Checking

Rance Cleaveland[1]*, Purush Iyer[1]** and Daniel Yankelevich[2]

[1] Dept of Computer Science, North Carolina State University, Raleigh, NC 27695-8206, USA.
[2] University of Buenos Aires, Argentina.

Abstract. This paper investigates the use of abstract-interpretation-inspired techniques for improving the performance of procedures for determining when systems satisfy formulas in branching-time temporal logic. A framework for abstracting system descriptions is developed, and a particular method for generating abstract systems from given abstractions on system states is defined and shown to be both safe and optimal, in the sense that concrete systems satisfy all the temporal formulas enjoyed by their abstracted counterparts. One may then use a model checker on an abstracted (and hence smaller) system in order to infer properties of a concrete system.

1 Introduction

One popular approach to the analysis of systems of concurrent processes involves the use of a temporal logic to formulate properties that the system being studied should enjoy [13]. Such logics enable users to specify the behavior a system should exhibit as it evolves over time. The utility of this approach to specification and verification has been illustrated in numerous case studies, and in the case of finite-state systems *model-checking* algorithms have been developed for determining automatically when systems satisfy formulas in certain logics [5, 14]. These algorithms have been implemented, and the resulting tools used to analyze a variety of different systems. However, the practical applicability of model checking remains limited by the *state-explosion* phenomenon: the number of states a concurrent system exhibits may be exponential in the number of processes, and as a model checker must in the worst case examine every system state, the automatic analysis of systems of many processes remains in many cases beyond the capabilities of existing tools. Coping with state explosion remains one of the central issues confronting researchers in automatic verification of finite-state concurrent programs.

In this paper we develop a framework for alleviating state explosion arising in the checking of systems against specifications given in the very expressive branching-time temporal logic CTL* [9]. The framework is based on the use of

* Research supported by NSF grant CCR-9120995, ONR Young Investigator Award N00014-92-J-1582, NSF Young Investigator Award CCR-9257963, and NSF grant CCR-9402807.
** Research supported in part by NSF grant CCR-9404619.

state abstractions, which intuitively group states in a concrete system into equivalence classes that may be viewed as states in an abstract, and therefore smaller, system. In particular, using the methodology of abstract interpretation [6, 10] we identify when system abstractions induced by state abstractions are safe in the sense that the CTL* formulas satisfied by abstracted systems are guaranteed to hold for the associated concrete systems. We also establish a restricted optimality result by showing that a particular system abstraction is the most precise one possible among a class of safe abstractions.

The remainder of the paper develops along the following lines. In the next section we present Kripke structures, which we use to model systems, and define the semantics of CTL*. The section following then introduces democratic Kripke structures, which will be used as abstractions of Kripke structures, and recasts the semantics of CTL* with respect to them. Section 4 defines an *abstraction* ordering on the states of a democratic Kripke structure and explores properties of such an ordering. Section 5 then defines a class of democratic Kripke structures that are "safe" approximations for a given Kripke structure, while the section following establishes that a particular method for constructing such approximations is, in a certain sense, optimal. The final section contains our conclusions and directions for future research.

Related Work. Several recent papers also address the use of state abstractions in model checking [1, 3, 4, 8, 11]. The approaches generally fall into two categories. The first (and the one pursued by this paper) treats abstractions as functions mapping system states to abstract states and is followed by [1, 3]. Both papers present techniques for "lifting" such abstractions to "safe" system abstractions for fragments of branching time logics and present several compelling examples illustrating the savings that such approaches may yield in model checking. However, in [3] the abstraction mechanism presented does not work for all of CTL*, and connections with abstract interpretation are not pursued. While the authors of [1] present their results in the context of abstract interpretation, they restrict their abstract models to be bisimilar to the concrete model when considering their full branching-time logic. However, authors of both papers do not characterize when other constructions might be safe or address the question of optimality.

The second approach requires that the collection of abstract states form a complete lattice and that the abstraction function mapping (sets of) concrete states to abstract states be a component in a Galois insertion/connection. This framework is pursued by [4, 8, 11]; each presents a mechanism for lifting a particular abstraction on states into a safe one on Kripke structures in which formulas in all of CTL* are preserved. However, the connection with abstract interpretation is generally left unexplored, as no optimality results are presented.

In light of these comments, the contributions of this paper may be summarized as follows:

- We extend the intuitive "functions as abstractions" approach of [1, 3] to handle all CTL* specifications.

– Using some simple machinery from abstract interpretation, we also give optimal constructions for generating abstract systems.

2 Kripke Structures and CTL*

In this paper we use *Kripke structures* to model the behavior of (unabstracted) systems. These are defined as follows.

Definition 1. A Kripke structure (abbrev. KS) has form (Q, δ, q_S, AP, I), where

– Q is a countable set of states;
– $\delta \subseteq Q \times Q$ is the transition relation;
– $q_S \in Q$ is the distinguished start state;
– AP is a set of atomic propositions; and
– $I : Q \rightarrow 2^{AP}$ is the interpretation function.

Intuitively, Q represents the set of states the system may enter, while δ represents the "execution steps" it may engage in. The interpretation function indicates which atomic propositions hold in a given state. For technical convenience, in what follows we restrict our attention to Kripke structures whose transition relations are total and whose interpretation functions are injective. That is, we require that for any $q \in Q$ there exist some $q' \in Q$ with $(q, q') \in \delta$ and that for any $q, q' \in Q$ with $I(q) = I(q')$ we have $q = q'$. These restrictions are standard [2].

We now present the syntax and semantics of CTL* [9], the branching time temporal logic used for formulating properties of systems. The set of formulae is given inductively as follows.

Definition 2. Given a set AP of atomic propositions, the state and path formulae of CTL*are defined as follows:

– If $p \in AP$ then p and $\neg p$ are state formulae.
– If ϕ_1 and ϕ_2 are state formulae then $\phi_1 \vee \phi_2$ and $\phi_1 \wedge \phi_2$ are state formulae.
– If ψ is a path formula then $\forall(\psi)$ and $\exists(\psi)$ are state formulae.
– If ϕ is a state formula then it is also a path formula.
– If ψ_1 and ψ_2 are path formulae then $\psi_1 \vee \psi_2$ and $\psi_1 \wedge \psi_2$ are path formulae.
– If ψ_1 and ψ_2 are path formula then $\mathrm{X}\psi_1$, $\psi_1 \mathrm{U} \psi_2$ and $\psi_1 \mathrm{V} \psi_2$ are path formulae.

We will use Φ to denote the set of all state formulae and Ψ to denote the set of all path formulae. We will also abuse terminology on occasion and refer to the set of state formulae as CTL* formulae.

In order to define the semantics of CTL* we first introduce the following terms.

Definition 3. Let Q be a set and $\delta \subseteq Q \times Q$ be a relation.

- Q^ω is the set of infinite sequences of elements of Q.
- A δ-path starting from $q \in Q$ is a sequence $\pi \equiv q_0 q_1 \ldots \in Q^\omega$ such that $q_0 = q$ and $(q_i, q_{i+1}) \in \delta$ for all $i \geq 0$.
- $\Pi(\delta, q)$ is the set of all δ-paths starting from q, while $\Pi(\delta) = \bigcup_{q \in Q} \Pi(\delta, q)$ is the set of all δ-paths.
- If $\pi = q_0 q_1 \ldots$ is a δ-path then define $\pi^i = q_i q_{i+1} \ldots$ to be the i^{th} suffix of π.

We now define the semantics of CTL* as follows.

Definition 4. Given a Kripke structure $K = (Q, \delta, q_S, AP, I)$, the satisfaction relation

$$\models_K \subseteq (Q \times \Phi) \cup (\Pi(\delta) \times \Psi)$$

is given inductively as follows:

- $q \models_K p$ iff $p \in I(q)$.
- $q \models_K \neg p$ iff $p \notin I(q)$.
- $q \models_K \phi_1 \wedge \phi_2$ iff $q \models_K \phi_1$ and $q \models_K \phi_2$.
- $q \models_K \phi_1 \vee \phi_2$ iff $q \models_K \phi_1$ or $q \models_K \phi_2$.
- $q \models_K \forall(\psi)$ iff for all $\pi \in \Pi(\delta, q)$, $\pi \models_K \psi$.
- $q \models_K \exists(\psi)$ iff there exists $\pi \in \Pi(\delta, q)$ with $\pi \models_K \psi$.
- $\pi \models_K \phi$, where ϕ is a state formula, iff $\pi = q_0 q_1 \ldots$ and $q_0 \models_K \phi$.
- $\pi \models_K \psi_1 \wedge \psi_2$ iff $\pi \models_K \psi_1$ and $\pi \models_K \psi_2$.
- $\pi \models_K \psi_1 \vee \psi_2$ iff $\pi \models_K \psi_1$ or $\pi \models_K \psi_2$.
- $\pi \models_K \mathrm{X}\psi$ iff $\pi^1 \models_K \psi$.
- $\pi \models_K \psi_1 \mathrm{U} \psi_2$ iff there exists $n \geq 0$ such that $\pi^n \models_K \psi_2$ and for each $i < n, \pi^i \models_K \psi_1$.
- $\pi \models_K \psi_1 \mathrm{V} \psi_2$ iff for all $n \geq 0$ we have that if $\pi^i \not\models_K \psi_1$ for all $i < n$, then $\pi^n \models_K \psi_2$.

We write $K \models \phi$ to mean $q_S \models_K \phi$.

The meaning of most of the operators is straightforward. Note that a state q satisfies $\forall(\psi)$ iff each path emanating from q satisfies the path formula ψ, while q satisfies $\exists(\psi)$ iff some path starting from q satisfies ψ. The X, U and V operators are modalities, with X and U being the usual "next-state" and "until" operators, respectively. V is the dual of U and may be thought of as a "release" operator: a path satisfies $\psi_1 \mathrm{V} \psi_2$ if there is a suffix that simultaneously satisfies both ψ_1 and ψ_2, with ψ_2 holding of all "previous" suffixes, or if ψ_2 holds of all suffixes.

Finally, it should be noted that although negation may only be applied to atomic propositions in our version of CTL*, the duality of $\wedge$ and $\vee$, and V and U, and the self-duality of X, enable the definition of a function NEG mapping state/path formulas to their "negated" counterparts. That is, NEG has the property that $q \models_K \phi$ ($\pi \models_K \psi$) iff $q \not\models_K \mathrm{NEG}(\phi)$ ($\pi \not\models_K \mathrm{NEG}(\psi)$).

3 Democratic Kripke Structures

Our goal in this paper is the following:

> Given a Kripke structure $K = (Q, \delta, q_S, AP, I)$, set A, and surjection $h : Q \rightarrow A$, define an abstract structure K_h with state set A whose CTL* properties are also enjoyed by K.

The motivation for this is that since h is a surjection, A is "smaller" than Q and hence will involve less effort to reason about than K.

The states, start state and atomic propositions of such a K_h are relatively straightforward to define. However, giving a transitions relation and interpretation function is not immediate; the former in particular presents difficulties, since a definition that is appropriate for the $\forall$ path quantifier is not necessarily reasonable for the $\exists$ path quantifier, and vice versa [8]. In this section we approach this problem by presenting a class of structures, which we call Democratic Kripke structures, fashioned after Modal Transition Systems [12], that will serve as approximations for Kripke structures.

Before defining these, we introduce the following useful notions.

Definition 5. Let S be a set, $\perp \notin S$, and $\delta \subseteq S \times S$ be a relation.

- $S_\perp = S \cup \{\perp\}$.
- $\hat{\delta} = \delta \cup \{\langle \perp, \perp \rangle\} \cup \{ \langle s, \perp \rangle \mid \forall s' \in S . \langle s, s' \rangle \notin \delta \}$.

Intuitively, $\perp$ will play the role of an "undefined state" in what follows. The only reason we introduce $\perp$ is to conform to the standard practice of interpreting CTL^* formulae with respect to total relations. We could alternatively have chosen to interpret CTL^* formulae with respect to all relations by ignoring all finite terminating paths, as in [3], and all of the results proved herein would still hold. Note that $\hat{\delta}$ extends δ to a total relation by adding a relationship from elements having no δ-relatives to $\perp$. We also need the following notion.

Definition 6. Let S be a set. Then $\mathcal{B}(S)$ is the complete boolean lattice freely generated from S. In what follows we use $\bigvee$, $\bigwedge$, $\neg$ and $\preceq$ to represent the least upper bound, greatest lower bound, negation and ordering operations, respectively.

Intuitively, for each $S' \subseteq S$ $\mathcal{B}(S)$ will contain elements of the form $\bigvee S'$ representing the "least upper bound" of S' and $\bigwedge S'$ representing the "greatest lower bound" of S'. The ordering relation represents a notion of entailment; so $b \preceq b'$ if b "entails" b'. We use $\mathbf{0}$ and $\mathbf{1}$ to represent $\bigvee \emptyset$ and $\bigwedge \emptyset$. If $b_1, b_2 \in \mathcal{B}(S)$ then we write $b_1 = b_2$ if $b_1 \preceq b_2$ and $b_2 \preceq b_1$ and $b_1 \sharp b_2$ if $b_1 \not\preceq b_2$ and $b_2 \not\preceq b_1$. We also use, e.g., $b_1 \vee b_2$ in lieu of $\bigvee\{b_1, b_2\}$ on occasion.

We may now define Democratic Kripke structures as:

Definition 7. A Democratic Kripke structure (abbrev. DKS) D has the form

$$(Q, \delta_C, \delta_L, q_S, AP, I)$$

where

- Q is a set of states;
- $\delta_C \subseteq Q \times Q$ is called the conservative transition relation;
- $\delta_L \subseteq Q \times Q$ is called the liberal transition relation;
- $\delta_C \subseteq \delta_L$;
- $q_S \in Q$ is the start state;
- AP is a set of atomic proposition; and
- $I : Q \to \mathcal{B}(AP)$ is the interpretation function such that for all $q \in Q$, $I(q) \not\preceq \mathbf{0}$.

Note that we do not require that I be injective.

In the remaining sections we will show that certain democratic Kripke structures may be viewed as abstractions of Kripke structures, in a precisely defined sense. To do so, we must define what it means for a DKS to satisfy a CTL* formula. We first introduce the following notation.

Notations 8. If $I : Q \to \mathcal{B}(AP)$ then $\hat{I} : Q_\perp \to \mathcal{B}(AP)$ is defined as follows.

$$\hat{I}(q) = \begin{cases} \mathbf{1} & if\ q = \perp \\ I(q) & otherwise \end{cases}$$

In the following we will define the meaning of CTL* formulae with respect to democratic Kripke structures. There are two main differences between the traditional interpretation and what follows. The first involves the meaning ascribed to boolean combinations of atomic propositions; the semantics below essentially asserts that such a proposition is satisfied by a state if one can "infer" the formula from the boolean expression labeling the state. This alteration improves on the "obvious" alteration of the traditional approach, as it allows more formulas to be determined to hold of a state. The second involves the use of two different transition relations to interpret formulae with the quantifiers $\forall$ and $\exists$. $\forall$-formulae are interpreted with respect to the liberal transition relation, while $\exists$-formulae refer to the conservative one.

Definition 9. Given a democratic Kripke structure $D = (Q, \delta_C, \delta_L, q_S, AP, I)$, the satisfaction relation $|\models_D \subseteq (Q_\perp \times \Phi) \bigcup ((\Pi(\hat{\delta_L}) \cup \Pi(\hat{\delta_C})) \times \Psi)$ is defined inductively as follows:

- If $p \in \mathcal{B}(AP)$ then $q \mid\models_D p$ iff $\hat{I}(q) \preceq p$.
- If $\phi_1 \wedge \phi_2 \notin \mathcal{B}(AP)$ then $q \mid\models_D \phi_1 \wedge \phi_2$ iff $q \mid\models_D \phi_1$ and $q \mid\models_D \phi_2$.
- If $\phi_1 \vee \phi_2 \notin \mathcal{B}(AP)$ then $q \mid\models_D \phi_1 \vee \phi_2$ iff $q \mid\models_D \phi_1$ or $q \mid\models_D \phi_2$.
- $q \mid\models_D \forall(\psi)$ iff for all $\pi \in \Pi(\delta_L, q), \pi \mid\models_D \psi$.
- $q \mid\models_D \exists(\psi)$ iff there exists $\pi \in \Pi(\delta_C, q)$ such that $\pi \mid\models_D \psi$.
- $\pi \mid\models_D \phi$, where ϕ is a state formula, iff $\pi = s_0 s_1 \ldots$ and $s_0 \mid\models_D \phi$.
- If $\psi_1 \wedge \psi_2 \notin \mathcal{B}(AP)$ then $\pi \mid\models_D \psi_1 \wedge \psi_2$ iff $\pi \mid\models_D \psi_1$ and $\pi \mid\models_D \psi_2$.
- If $\psi_1 \vee \psi_2 \notin \mathcal{B}(AP)$ then $\pi \mid\models_D \psi_1 \vee \psi_2$ iff $\pi \mid\models_D \psi_1$ or $\pi \mid\models_D \psi_2$.
- $\pi \mid\models_D \mathrm{X}\psi$ iff $\pi^1 \mid\models_D \psi$.
- $\pi \mid\models_D \psi_1 \mathrm{U} \psi_2$ iff there exists $n \geq 0$ such that $\pi^n \mid\models_D \psi_2$ and for all $i < n, \pi^i \mid\models_D \psi_1$.

- $\pi \mid\models_D \psi_1 \mathrm{V} \psi_2$ iff for all $n \geq 0$, if $\pi^i \not\mid\models_D \psi_1$ for all $i < n$ then $\pi^n \mid\models_D \psi_2$.

We write $D \mid\models \phi$ to mean $q_S \mid\models_D \phi$.

The following establishes that Kripke structures may be viewed as certain kinds of democratic Kripke structures. We first introduce the following notation.

Notations 10. Let AP be a set, with $S \subseteq AP$. Then the characteristic boolean value for S, $\chi_{AP}(S) \in \mathcal{B}(AP)$, is defined by $\chi_{AP}(S) = \bigwedge_{s \in S} s \wedge \bigwedge_{s \notin S} \neg s$.

Lemma 11. *Let $K = (Q, \delta, q_S, AP, I)$ be a Kripke structure, and define $D_K = (Q, \delta, \delta, q_S, AP, \chi_{AP} \circ I)$. Then for any CTL* formula ϕ, we have that $K \models \phi$ iff $D_K \mid\models \phi$.*

4 Ordering on DKS

In this section we will introduce an abstraction ordering on the states of a DKS. The abstraction ordering is similar to (more precisely, an inverse of) the refinement ordering defined in [12].

Intuitively a state a_1 will be said to be an abstraction of a_2 provided every conservative edge from a_1 is matched by a conservative edge from a_2 and every liberal edge from a_2 is matched by some liberal edge from a_1. More formally we have:

Definition 12. Let $D = (A, \delta_C, \delta_L, a_S, AP, I_D)$ be a democratic Kripke structure. A relation R is an abstraction relation provided for all $a_1, a_2 \in A$ we have:

$a_1 R a_2$ iff
- $I_D(a_2) \preceq I_D(a_1)$,
- If $(a_1, a_1') \in \hat{\delta_C}$ then there exists $a_2' \in A$ such that $(a_2, a_2') \in \hat{\delta_C}$ and $a_1' R a_2'$, and
- If $(a_2, a_2') \in \hat{\delta_L}$ then there exists $a_1' \in A$ such that $(a_1, a_1') \in \hat{\delta_L}$ and $a_1' R a_2'$.

We say $a_1 \trianglelefteq a_2$ provided there exists an abstraction relation R which includes the pair (a_1, a_2).

Note that states of two DKSs D_1 and D_2 can be compared by constructing a new DKS D containing a (discriminated) union of states, transitions, and the interpretation functions of D_1 and D_2. Furthermore, states of a DKS D and a Kripke structure K can be compared by first lifting K to be a DKS K_D and then taking the discriminated union of D and D_K.

We would now like to recall some results about the abstraction ordering, from [12], adapted to our setting. But before we can do so, we need to establish the following conditions about paths in the democratic Kripke structure, all of which follow from the definition:

Lemma 13. *Let $D = (A, \delta_C, \delta_L, a_S, AP, I_D)$ be a democratic Kripke structure, and let $a_0 \trianglelefteq b_0$ where $\{a_0, b_0\} \subseteq A$. We then have*

$$
\begin{tikzcd}
2^{CTL^*} \arrow[r, "id"] & 2^{CTL^*} \\
 & \rotatebox[origin=c]{90}{$\subseteq$} \\
\mathcal{K}(Q, AP) \arrow[uu, "\models"] \arrow[r, "\mathcal{F}_h"'] & \mathcal{D}(A, AP) \arrow[u, "||\!\!\models"']
\end{tikzcd}
$$

Fig. 1. The commuting diagram.

- *For every path $\pi = a_0 a_1 \ldots$ over the conservative relation δ_C there is a path $\pi' = b_0 b_1 \ldots \in \Pi(\delta_C, b_0)$ such that for $\forall i \geq 0: \ a_i \trianglelefteq b_i$.*
- *For every path $\pi = b_0 b_1 \ldots$ over the liberal relation there exists a path $\pi' = a_0 a_1 \ldots \in \Pi(\delta_L, a_0)$ such that $\forall i \geq 0: \ a_i \trianglelefteq b_i$.*

The lemma about paths would now allow us to show, by induction on the structure of formulae, that the $\trianglelefteq$ ordering is a sufficient condition for an inclusion ordering on the sets of formulae provable about a state:

Theorem 14 [12]. *Let $D = (A, \delta_C, \delta_L, a_S, AP, I_D)$ be a democratic Kripke structure, and let $a \trianglelefteq b$ where $\{a, b\} \subseteq A$. We then have:*

$$\forall \phi \in \Phi: \ a \mid\models_D \phi \Rightarrow b \mid\models_D \phi$$

It turns out that the converse of Theorem 14 holds for the class of *image-finite* democratic Kripke structures. Intuitively, a transition relation is image finite provided there are only a finite number of transitions out of every state, and a democratic Kripke structure is image-finite provided both of its transition relations are image finite.

Theorem 15 [12]. *Let $D = (A, \delta_C, \delta_L, a_S, AP, I_D)$ be a image-finite democratic Kripke structure, and let a and b be two states of D. We then have:*

$$(\forall \phi \in \Phi: \ a \mid\models_D \phi \Rightarrow b \mid\models_D \phi) \Rightarrow a \trianglelefteq b$$

The theorem can be proved by showing that if $a \not\trianglelefteq b$ then there has to be a CTL^* formula that is true of a and not of b.

In the next two sections we will make use of Theorem 14 to show soundness of our abstract interpretation scheme and Theorems 14 and 15 to show optimality of a canonical abstraction.

5 Approximating Kripke Structures

In the remainder of this paper we consider the following question.

> Given a Kripke structure $K = (Q, \delta, q_S, AP, I)$ and surjection $h : Q \to A$, when is a democratic Kripke structure with state set A a "good" approximation of K?

This question can be formulated using notions from abstract interpretation as follows. Consider the complete lattice $\langle 2^{\Phi}, \supseteq \rangle$, i.e. the set of sets of CTL* formulas ordered by reverse inclusion. Also let $\mathcal{K}(Q, AP)$ be the class of Kripke structures with state set Q and atomic propositions AP, and let $\mathcal{D}(A, AP)$ be the similarly defined class of democratic Kripke structures. Then the relations $\models$ and $|\!\models$ may be seen as mappings with types $\mathcal{K}(Q, AP) \to 2^{\Phi}$ and $\mathcal{D}(Q, AP) \to 2^{\Phi}$, respectively: e.g. $\models (K) = \{\phi \mid K \models \phi\}$. Given $h : Q \to A$, we would like to do the following.

1. Determine when a transformation $\mathcal{F}_h : \mathcal{K}(Q, AP) \to \mathcal{D}(S, AP)$ is safe, i.e. satisfies: for all $K \in \mathcal{K}(Q, AP)$, $\models (K) \supseteq (|\!\models \circ \mathcal{F}_h)(K)$.
2. Devise an optimal safe transformation $\mathcal{C}_h$, i.e., which satisfies the following property: for any safe $\mathcal{F}_h$, $K \in \mathcal{K}(Q, AP)$, $(|\!\models \circ \mathcal{C}_h)(K) \supseteq (|\!\models \circ \mathcal{F}_h)(K)$.

The associated commuting diagram appears in Figure 1.

We begin by defining when a DKS is an approximant with respect to surjection h of a KS.

Definition 16. Let $K = (Q, \delta, q_S, AP, I_K)$ be a Kripke structure, $h : Q \to A$ a surjection, and $D = (A, \delta_C, \delta_L, a_S, AP, I_D)$ a democratic Kripke structure. Then D is an approximation of K with respect to h (written $D \propto_h K$) provided that the following conditions are met:

- (Preservation of initial state) $h(q_S) = a_S$.
- (Preservation of transitions) $\forall q \in Q$ we have $h(q) \trianglelefteq q$.

Given this definition of approximation it is easy to see that all approximants of a Kripke structure K satisfy only properties that K itself satisfies; consequently, any $\mathcal{F}_h$ that returns only approximants of its argument is guaranteed to be safe in the sense of abstract interpretation. Formally, we have:

Theorem 17. *Let D, h and K be such that $D \propto_h K$. Then for any CTL* formula ϕ, $D |\!\models \phi$ implies $K \models \phi$.*

6 An Optimal Abstraction

The previous section established that if $\mathcal{F}_h$ is such that $\mathcal{F}_h(K) \propto_h K$ then $\mathcal{F}_h$ is a safe abstraction for Kripke structures. In this section we turn to the question of optimality; in particular, we wish to determine under what conditions a "most precise" safe abstraction over Kripke structures may be defined. We would like to approach this question by defining a particular transformation $\mathcal{C}_h$ and showing that it yields the most precise approximant among all the approximants (as defined in the previous section). This turns out to be infeasible, for reasons we go into later. Instead, we establish a more restricted optimality result by focusing on a reasonable subclass of approximants.

We begin by defining $\mathcal{C}_h$ as follows:

Definition 18. Given a Kripke structure $K = (Q, \delta, q_S, AP, I_K)$ and a surjection $h : Q \to A$ define $\mathcal{C}_h(K)$ to be $(A, \delta_C, \delta_L, a_S, AP, I_D)$, where

- $(a_1, a_2) \in \delta_C$ iff for each $q_1 \in h^{-1}(a_1)$ there is a $q_2 \in h^{-1}(a_2)$ with $(q_1, q_2) \in \delta$;
- $(a_1, a_2) \in \delta_L$ iff there exist $q_1 \in h^{-1}(a_1)$ and $q_2 \in h^{-1}(a_2)$ with $(q_1, q_2) \in \delta$;
- $a_0 = h(q_0)$; and
- $I_D(a) = \bigvee_{q \in h^{-1}(s)} \chi_{AP}(I_K(s))$.

The definition of δ_C and δ_L are similar to those found in [4, 8, 11]. The next lemma indicates that $\mathcal{C}_h$ is safe.

Lemma 19. *Let $K = (Q, \delta, q_S, AP, I_K)$ be a Kripke structure and $h : Q \to A$ a surjection. Then $\mathcal{C}_h(K)$ is a DKS and $\mathcal{C}_h(K) \propto_h K$.*

We will sometimes refer to $\mathcal{C}_h(K)$ as the canonical approximation to K under h.

We are interested in showing that $\mathcal{C}_h(K)$ is the "best approximation" to K among all of the democratic Kripke structures over the set of states A and the atomic propositions AP. However, two obstacles prevent us from being able to establish this result for arbitrary KS's and DKS's.

1. Arbitrary KS's may have an infinite number of transitions emanating from a particular state. In general, our technique for constructing formulas will in some cases require taking conjunctions over the transitions from a state, and if this number is infinite we are left with an infinite conjunction, which cannot be expressed in the logic. To cope with this problem we will restrict our attention to image-finite KS's.
2. Arbitrary DKS's may have overly imprecise interpretation functions. To see the problem that this poses, consider a DKS of the form

$$D = (A, \delta_C, \delta_L, a_S, AP, I)$$

where $I(a) = \mathbf{1}$ for any $a \in A$. Then regardless of what δ_C, δ_L and a_S are,

$$\forall \phi : |\models_D \phi \Rightarrow \models_K \phi$$

for any $K \in \mathcal{K}(Q, AP)$ and $h : Q \to A$. The intuitive reason for this is that the (extreme) imprecision of I ensures that the only CTL* formulas satisfied by D are satisfied by any KS. To handle this difficulty we will make use of a class of DKS's that have "precise" interpretation functions.

We now define the notion of an abstraction being *precise*.

Definition 20. Let $K = (Q, \delta, q_S, AP, I)$ be a KS with $h : Q \to A$ and $I_A : A \to \mathcal{B}(AP)$. Then I_A is precise with respect to K and h if, whenever $a_1 \neq a_2$, then the following holds: for all $q_1 \in h^{-1}(a_1)$, $\chi_{AP}(I(q_1)) \not\preceq I_A(a_2)$. A DKS is said to be precise with respect to K and h iff its interpretation function is precise with respect to K and h. A transformation $\mathcal{F}_h : \mathcal{K}(Q, AP) \to \mathcal{D}(A, AP)$ is precise for h iff $\mathcal{F}_h(K)$ is precise with respect to h and K, for every K.

Intuitively, an interpretation function $I_A : A \to \mathcal{B}(AP)$ is precise with respect to K and h if for any q, the only $a \in A$ for which q "satisfies" $I(a)$ is $h(q)$. Also note that if AP is finite, then $\mathcal{B}(AP)$ may be seen as the semantic correspondents of propositions consisting of boolean combinations of AP.

Our aim in this section is to show that $\mathcal{C}_h$ is the "best" transformation that is precise with respect to h. To this end, and also to provide motivation for this definition of precise, we prove the following lemmas. The first establishes that if one "strengthens" an interpretation function in a DKS, then one is able to establish that more CTL* formulas hold.

Lemma 21. *Let $D_1 = (A, \delta_C, \delta_L, a_S, AP, I_1)$, and let $D_2 = (A, \delta_C, \delta_L, a_S, AP, I_2)$ be such that for any $a \in A$, $I_2(a) \preceq I_1(a)$. Then for any $\phi \in \Phi$ and $a \in A$, if $a \models_{D_1} \phi$ then $a \models_{D_2} \phi$.*

The next result states in essence that any DKS D may be "strengthened" into a precise one. This may be viewed as a justification for considering precise DKS's rather than arbitrary ones, since the latter are essentially "weaker" versions of the former.

Lemma 22. *Let $K = (Q, \delta, q_S, AP, I_K)$ be a KS, $D = (A, \delta_C, \delta_L, a_S, AP, I_D)$ be a DKS, and let $h : Q \to A$ be a surjection such that $K \propto_h K$. Then there exists a D' that is precise with respect to K and h and satisfies the following: for any $\phi \in \Phi$ and $a \in A$, $a \models_D \phi$ implies $a \models_{D'} \phi$.*

Proof. Define $D' = (A, \delta_C, \delta_L, a_S, AP, I'_D)$, where

$$I'_D(a) = \bigvee_{q \in h^{-1}(a)} (\chi_{AP} \circ I_K)(q).$$

That I'_D is precise with respect to K and h follows from the definition of χ_{AP} and the fact that I_K is injective. The rest of the result follows from the interpretation-preservation property of approximants and Lemma 21.

We now come to main theorem of the paper.

Theorem 23. *Suppose $D \propto_h K$, where D is a image-finite DKS and K is a image-finite Kripke structure such that D is precise with respect to K and h. We then have that every state a of D is related by $\trianglelefteq$ to the same state in $\mathcal{C}_h(K)$.*

Proof. Given that D is precise with respect to K and h we can easily establish that every conservative edge in D is also a conservative edge in $C_h(K)$ and that every liberal edge of $C_h(K)$ is a liberal edge of D. The proof then follows from a routine application of Theorems 14 and 15.

Corollary 24. *Let $K = (Q, \delta, q_S, AP, I_D)$ be an image-finite Kripke structure and $h : Q \rightarrow A$ be a surjection. Then $C_h(K)$ is the optimal approximation to K among all precise approximations of K under h.*

Note that the previous theorem is restricted to image-finite systems; this restriction is necessary as a corresponding claim for infinitary transition systems would require us to construct a formula with an infinite number of terms in the proof of Theorem 15.

Discussion. We close this discussion with some comments about $\mathcal{C}_h$. It should be noted that there is some subtlety involved in generating appropriate finitary propositions from the infinitary interpretations that the DKS's (generated by $C_h(K)$) may have. Also, in practice $\mathcal{C}_h(K)$ may not be much smaller than K, since the interpretations assigned to abstract states will in general be more complex that those assigned to states in K. However, smaller abstract systems can be obtained by weakening the interpretations function in $\mathcal{C}_h(K)$: one reasonable possibility, for example, is to assign to each abstract state a conjunction of atomic propositions found in every state corresponding to it in Q.

7 Conclusion

In this paper we have developed a framework for abstracting Kripke structures on the basis of abstraction functions given for their states. In particular, we introduced the class of democratic Kripke structures and defined a relation indicating when a particular democratic Kripke structure is a "safe approximation" to a given Kripke structure. This relation may be used as a basis for establishing the safety of abstraction transformations that generate democratic Kripke structures from traditional ones: one need only show that the result of applying the transformation to a Kripke structure yields a "safe approximation." We also presented a particular transformation that we showed was optimal, in a particular sense.

As future work we would like to investigate the development of automated support for applying abstractions and to apply the formalism developed in this paper to a series of case studies.

References

1. S. Bensalem, A. Bouajjani, C. Loiseaux and J. Sifakis. Property prerving simulations. In *CAV 92*, LNCS 663, 1992.
2. J. R. Burch, E. M. Clarke and D. E. Long. Symbolic model checking with partitioned transition relations. In *VLSI 91*, Edinburgh, Scotland, 1990.

3. E. M. Clarke, O. Grumberg and D. Long. Model checking and abstraction. In Proc of *XIX POPL*, Jan 1992.
4. R. Cleaveland, S. P. Iyer, D. Yankalevich. Abstractions for preserving all CTL^* formulae Tech Report 94-03, NCSU Computer Science Department, 1994.
5. E. M. Clarke, E. A. Emerson and A. P. Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. *ACM TOPLAS*, 8(2):244–263, 1986.
6. P. Cousot and R. Cousot. Systematic Design of Program Analysis Frameworks. In *VI POPL*, 1979.
7. P. Cousot and R. Cousot. Comparing the Galois connection and the widening/narrowing approaches to abstract interpretation. In Proc of *Conference on Programming Language Implementation and Logic Programming*, LNCS 631, August, 1992.
8. D. Dams, O. Grumberg and R. Gerth. Abstract Interpretation of Reactive Systems: Abstractions Preserving ∀CTL*, ∃CTL* and CTL*. In IEEE PROCOMONET. Nov, 1994.
9. E. A. Emerson and J. Y. Halpern. "Sometimes" and "not never" revisited: on branching time versus linear time temporal logic. *JACM*, 33(1):151–178, 1986.
10. N. Jones and F. Nielsen. Abstract Interpretation: a Semantic-Based Tool for Program Analysis. To appear in *Handbook of Logic in Computer Science*, (ed) S. Abramsky, D. Gabbay and T. S. E. Maibaum.
11. P. Kelb. Model Checking and Abstraction: A framework approximating both truth and failure information. University of Oldenburg, March 30, 1994.
12. K. Larsen. Modal Specifications. In Proc. of *CAV '89*, (ed) J. Sifakis, LNCS Vol. 407, pp: 232–246, 1989.
13. Z. Manna and A. Pnueli. *Temporal Logic of Reactive and Concurrent Systems* Springer-Verlag, 1992.
14. J. P. Quielle and J. Sifakis. Specification and Verification of Concurrent Systems in Cesar. In Proc of *V International Symposium on Programming*, LNCS Vol 137, 1981.