

基于完备抽象解释的模型检验 CTL 公式研究

钱俊彦¹⁾ 徐宝文^{2),3)}

¹⁾(桂林电子科技大学计算机与控制学院 广西 桂林 541004)

²⁾(南京大学计算机软件新技术国家重点实验室 南京 210093)

³⁾(南京大学计算机科学与技术系 南京 210093)

摘 要 在模型检验中,抽象是解决状态空间爆炸问题的重要方法之一.给定具体 Kripke 结构和时序描述语言 CTL,基于抽象解释框架以及完备抽象解释和性质强保留之间的关系,抽象模型最小精化使得 CTL 性质强保留,可转换为抽象解释中抽象域的最小完备精化,并且总是存在抽象域的最小完备精化.根据状态标签函数确定初始抽象域,然后通过不动点求解,获得对 CTL 标准算子完备的最小抽象域,并依据此抽象域求得 CTL 性质强保留的最优抽象状态划分,最后构造出 CTL 性质强保留且最优的抽象状态转换系统.并指出了抽象域对 CTL 标准算子是完备的当且仅当抽象域对补集和标准前向转换是完备的.

关键词 抽象解释;抽象模型检验;强保留;完备性;精化

中图法分类号 TP311 **DOI号**: 10.3724/SP.J.1016.2009.00992

Model Checking CTL Based on Complete Abstraction Interpretation

QIAN Jun-Yan¹⁾ XU Bao-Wen^{2),3)}

¹⁾(School of Computer and Control, Guilin University of Electronic Technology, Guilin, Guangxi 541004)

²⁾(State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093)

³⁾(Department of Computer Science and Technology, Nanjing University, Nanjing 210093)

Abstract Abstraction plays a fundamental role in combating state-space explosion in model checking. In a complete abstract interpretation-based view, the authors reduce the state space of a Kripke structure in order to obtain a minimal abstract state translation system that strongly preserves a given temporal specification language CTL. According to a relation between completeness of abstract interpretations and strong preservation of abstract model checking, the problem of minimally refining abstract model in order to make it strongly preserving CTL can be formulated as a minimal abstract domain refinement in abstraction interpretation in order to get completeness, and this refined complete abstract domain always exists. Given initial abstract domain, we can obtain a minimal abstract domain which is complete for the standard operators of CTL by the iterative computation of the fixpoint. Moreover, the corresponding partition is the coarsest partition which is strong preserving for CTL. The authors show that the abstract domain is complete for the standard operators of CTL iff it is complete for complement and standard predecessor transformer operators.

Keywords abstract interpretation; abstract model checking; strong preservation; completeness; refinement

1 引言

模型检验^[1]是基于有限状态空间搜索来保证软硬件设计正确性的形式化自动验证技术,能在模型不满足规格时给出反例.在模型检验中,抽象是解决状态空间爆炸的一种重要方法,其目标是构造足够小的抽象模型,使之能有效分析与验证.抽象模型检验的主要思想是通过抽象状态近似具体状态的某些性质,并在抽象状态之间定义抽象转换关系,从而构造出抽象模型,然后在抽象模型中检验时序性质,通过时序性质在抽象模型的结果,推导出在具体模型中是否满足.抽象模型检验已经在硬件验证领域取得巨大成功,特定情况下能验证有 10^{1300} 可达状态的 ALU 电路^[2].

通常抽象主要有两种方式:弱保留抽象和强保留抽象.对于性质弱保留的抽象,可分为两类:向下近似抽象和向上近似抽象.给定具体模型 K 和抽象模型 A 以及性质 ϕ ,如果 A 是 K 按照向下近似抽象方式获得的抽象模型,则 $A \models \phi \Rightarrow K \models \phi$,即性质在抽象模型不满足,推导出在具体模型中也不满足;如果 A 是 K 按照向上近似抽象方式获得的抽象模型,则 $A \models \phi \Rightarrow K \models \phi$,即时序性质在抽象模型满足,推导出在具体模型也满足. Clarke^[3-4] 采用向上近似抽象方式,基于反例精化的思想进行抽象模型检验,即通过模型检验自动验证抽象模型是否满足所期望的性质,如果满足,则报告正确;否则给出抽象反例,检查抽象反例是否合理,若合理,则生成具体反例;否则依据不合理反例精化抽象模型;重复迭代精化直到给出满足性质的肯定回答或给出不满足性质的具体反例.

对于性质强保留的抽象,时序性质在抽象模型满足当且仅当其在具体模型满足,即性质在抽象模型满足(或不满足),则在具体模型中也满足(或不满足).性质强保留是高期望的,且要获得最优的强保留抽象模型是困难的.本文将基于抽象解释的理论以及完备抽象解释和性质强保留之间的关系(即抽象域对 CTL 标准算子是完备的,那么相对应的抽象划分强保留 CTL 性质),通过抽象域的最小完备精化,构造一个抽象状态划分且强保留 CTL 性质的抽象模型.首先根据状态标签函数确定初始抽象域,然后通过不动点求解,得到对补集和标准前向转换完备的最优抽象域,也就是对 CTL 标准算子完备的最

小抽象域,依据此抽象域求解 CTL 性质强保留的最优抽象状态划分.在抽象状态划分之间定义抽象转换关系,构造 CTL 性质强保留的抽象模型.

典型的抽象模型检验是基于状态划分和抽象 Kripke 结构的,给定具体状态转换系统或 Kripke 结构,通过抽象映射获得抽象模型.而在抽象解释框架下,依据具体状态幂集的抽象解释(Galois 连接或闭包操作)导出近似于具体语义的抽象语义. Cousot 首先介绍了用于静态程序分析的抽象解释框架^[5-6],此框架使用 Galois 连接(或闭包操作)在具体性质域和抽象性质域之间建立联系,且证明对任意具体性质,存在唯一的最优抽象近似.

近年来,许多研究人员对抽象模型检验的基础和原则进行了研究,并在许多方面做了改进^[2,7-19]. Graf 和 Saidi^[20] 提出一种特殊的抽象解释形式——谓词抽象,实现了抽象的自动化,然而此方法是不完备的.当前基于谓词抽象的技术已应用到软件系统的模型检验中,诸如验证 Java 程序的 Bandera^[21] 和 Java PathFinder^[22] 以及验证 C 语言的 SLAM^[23-24]、MAGIC^[25-26] 和 BLAST^[27-28].

Cousot^[7] 和 Giacobazzi^[11,29] 的研究表明抽象解释的完备性不依赖抽象语义算子,而仅仅依赖于抽象域,因此是抽象域性质.在此基础上,给出了一个结论,即抽象域的完备核或最小完备精化总是存在,并描述了构造完备抽象解释的方法,也就是抽象解释的完备化可转化为抽象域完备精化. Giacobazzi^[10] 首先给出了性质强保留和完备抽象解释之间的关系,并指出当抽象缺乏完备性,则产生不合理反例.为了消除不合理反例,可通过最小精化使抽象解释完备. Ranzato 和 Tapparo^[13-16] 描述了基于抽象解释的性质强保留模型,在完备抽象解释和强保留抽象模型之间建立了精确的一致性原则,并指出抽象域的完备精化可用不动点对其进行刻画.本文在上述工作的基础上,运用一些已有的结论,证明了抽象域对 CTL 标准算子是完备的当且仅当抽象域对补集和标准前向转换是完备的,并根据完备抽象解释和强保留抽象模型之间的关系以及文献[14]中引理 5.1 和 5.2,改进了文献[14]中求解最优的完备精化闭包算法,从而获得 CTL 性质强保留的最优抽象模型.

2 基础知识

2.1 基础符号

假定 X 是集合, $Fun(X)$ 表示函数 $f: X^n \rightarrow X$ 的

集合, 其中 $n = ar(f) \geq 0$, 当 $n = 0$ 时, f 仅表示 X 的特殊对象. 如果 $F \subseteq Fun(X)$ 且 $Y \subseteq X$, 那么 $F(Y) =_{\text{def}} \{f(y) \mid f \in F, y \in Y^{ar(f)}\}$, 换句话说, $F(Y)$ 就是集合 Y 对于 F 中所有函数的像集. 如果函数 $f: X \rightarrow Y$ 且 $g: Y \rightarrow Z$, 那么 $g \circ f: X \rightarrow Z$ 表示的函数组合 $g \circ f = \lambda x. g(f(x))$. 符号 \mathcal{C} 表示补集, 譬如 $Y \subseteq X$ 且 X 是全集, $\mathcal{C}(Y) = X \setminus Y$. 为了简便起见, 用去除括号和逗号的紧凑格式表示 X 的子集集合 $S \in \mathcal{P}(\mathcal{P}(X))$, 诸如 $\{1, 12, 13, 123\} \in \mathcal{P}(\mathcal{P}(\{1, 2, 3\}))$. 让 $Part(X)$ 表示 X 的划分集, 在划分中的集合称为块. 假如 $\equiv \subseteq X \times X$ 是等价关系, 那么 $P \equiv \in Part(X)$ 是对应的划分. 反之, 如果 $P \in Part(X)$, 则 \equiv_P 表示对应在 X 上的等价关系.

假定 (P, \leq) 是偏序集, 简称为 P_{\leq} , 函数 $f: P \rightarrow P$, f 是单调升函数当且仅当 $a \leq b \Rightarrow f(a) \leq f(b)$, 同理 f 是单调降函数当且仅当 $a \leq b \Rightarrow f(b) \leq f(a)$. 使用向上箭头 \uparrow 和向下箭头 \downarrow 各自表示单调升和降函数. 当 f 保留最小上界 (lub), 则 f 是连续的. 对任意 $Y \subseteq P$, 如果 $f(\bigvee Y) = \bigvee f(Y)$, 那么 f 是可加的. 用符号 $lfp(f)$ 和 $gfp(f)$ 表示 f 的最小和最大不动点. 假定 $\langle P, \leq, \bigvee, \bigwedge, \top, \perp \rangle$ 是完备格, 通过 Knaster-Tarski 定理, 最小不动点描述为

$lfp(f) = \bigwedge \{x \in P \mid f(x) \leq x\} = \bigvee_{a \in Ord} f^{a, \uparrow}(\perp)$, 其中 f 的向上迭代序列 $\{f^{a, \uparrow}(x)\}_{a \in Ord}$ 通过递归定义如下:

- (1) $\alpha = 0, f^{0, \uparrow}(x) = x$;
- (2) 后继序数 $\alpha + 1, f^{\alpha+1, \uparrow}(x) = f(f^{\alpha, \uparrow}(x))$;
- (3) 有限序数 $\alpha, f^{\alpha, \uparrow}(x) = \bigvee_{\beta < \alpha} f^{\beta, \uparrow}(x)$.

类似地, 最大不动点 $gfp(f) = \bigwedge \{x \in P \mid x \leq f(x)\} = \bigwedge_{a \in Ord} f^{a, \downarrow}(\top)$.

假定转换系统 (Σ, R) , 其中 Σ 表示状态集, $R \subseteq \Sigma \times \Sigma$ 表示转换函数, 通常用 \xrightarrow{R} 表示. 假设 R 是全函数, 即对任意 $s \in \Sigma$, 存在 $t \in \Sigma$, 满足 $s \xrightarrow{R} t$. 给定 Kripke 结构 $\mathcal{K} = (\Sigma, R, AP, L)$ 由转换系统 (Σ, R) , 原子命题集合 AP 和标签函数 $L: \Sigma \rightarrow \mathcal{P}(AP)$ 组成. \mathcal{K} 的路径定义为 $Path(\mathcal{K}) =_{\text{def}} \{\pi. \mathbb{N} \rightarrow \Sigma \mid \forall i \in \mathbb{N}, \pi_i \xrightarrow{R} \pi_{i+1}\}$. 转换关系 R 在 $\mathcal{P}(\Sigma)$ 上的前向/后向转换定义如下:

- (1) $pre_R =_{\text{def}} \lambda Y. \{s \in \Sigma \mid \exists t \in Y. s \xrightarrow{R} t\}$;
- (2) $\widetilde{pre}_R =_{\text{def}} \mathcal{C} \circ pre_R \circ \mathcal{C} = \lambda Y. \{s \in \Sigma \mid \forall t \in \Sigma. s \xrightarrow{R} t \Rightarrow t \in Y\}$;
- (3) $post_R =_{\text{def}} \lambda Y. \{t \in \Sigma \mid \exists s \in Y. s \xrightarrow{R} t\}$;

$$(4) \widetilde{post}_R =_{\text{def}} \mathcal{C} \circ post_R \circ \mathcal{C} = \lambda Y. \{t \in \Sigma \mid \forall s \in \Sigma. s \xrightarrow{R} t \Rightarrow s \in Y\}.$$

直观地, $post_R(Y)$ 表示状态集 Y 的后继集合 $\bigcup_{s \in Y} R(s)$ 与 Σ 的交集, 而 $pre_R(Y)$ 表示状态集 Y 的前趋集合. 当上下文清晰时, 下标 R 被忽略.

2.2 抽象解释

在抽象解释理论中, 抽象域可通过 Galois 连接或闭包操作来获取, 两者是等价的.

第 1 种情况. 具体域 C 和抽象域 A 通过 Galois 连接 (C, α, γ, A) 联系起来, 其中 $\alpha: C \rightarrow A$ 是抽象映射, 而 $\gamma: A \rightarrow C$ 是具体化映射. (C, α, γ, A) 是 Galois 入射 (GI) 的, 当且仅当 α 是满射, 或 γ 是一一映射. 具体域 C 的 GI 对于精确性是拟序的, $\mathcal{G}_1 = (C, \alpha_1, \gamma_1, A_1) \sqsubseteq \mathcal{G}_2 = (C, \alpha_2, \gamma_2, A_2)$ (即 A_1 比 A_2 更精确) 当且仅当 $\gamma_1 \circ \alpha_1 \sqsubseteq \gamma_2 \circ \alpha_2$. 当 $\mathcal{G}_1 \sqsubseteq \mathcal{G}_2$ 且 $\mathcal{G}_2 \sqsubseteq \mathcal{G}_1$, 则 \mathcal{G}_2 和 \mathcal{G}_1 是等价的. 给定 $\mathcal{G} = (C, \alpha, \gamma, A)$ 是 GI, $f: C \rightarrow C$ 是具体语义函数, $f^\# : A \rightarrow A$ 是对应的抽象语义函数, 当满足 $\alpha \circ f \sqsubseteq f^\# \circ \alpha$ (或 $f \circ \gamma \sqsubseteq \gamma \circ f^\#$), $(A, f^\#)$ 是 (C, f) 的安全抽象. 如果 $\alpha \circ f = f^\# \circ \alpha$ (或 $f \circ \gamma = \gamma \circ f^\#$), 那么抽象解释是完备的, 称为后向完备 (或前向完备). Giacobazzi^[12] 指出完备性唯一依赖抽象映射, 即抽象域的性质. 事实上, 存在 $f^\# : A \rightarrow A$ 使得 $\langle A, f^\# \rangle$ 是后向完备 (或前向完备) 当且仅当 $\gamma \circ \alpha \circ f \circ \gamma \circ \alpha = \gamma \circ \alpha \circ f$ (或 $\gamma \circ \alpha \circ f \circ \gamma \circ \alpha = f \circ \gamma \circ \alpha$). 当 $\gamma \circ \alpha \circ f \circ \gamma \circ \alpha = \gamma \circ \alpha \circ f$ (或 $\gamma \circ \alpha \circ f \circ \gamma \circ \alpha = f \circ \gamma \circ \alpha$), 则称 GI \mathcal{G} 是对 f 后向完备 (或前向完备).

第 2 种情况. 在具体域 C 上通过向上闭包操作获得抽象域 A , 简称为 uco 或闭包. 向上闭包操作是在 C 上单调的、幂等的和扩展的操作 $\langle uco(C), \sqsubseteq \rangle$ 表示在 C 上的所有 uco 的偏序集, 假定 $\rho \in uco(C)$ 是通过不动点唯一确定的集合, 即 $\rho(C) =_{\text{def}} \{x \in C \mid \rho(x) = x\}$, 则 $\rho \sqsubseteq \eta$ 当且仅当 $\eta(C) \subseteq \rho(C)$. 当 $\langle C, \leq, \bigvee, \bigwedge, \top, \perp \rangle$ 是完备格, 那么 $\langle uco(C), \sqsubseteq, \sqcup, \sqcap, \lambda x. \top, \lambda x. x \rangle$ 也是完备格, 表示 C 的所有可能抽象域的完备格. 如果当 $X \subseteq C$ 是 meet 封闭时, 则 X 是一个 uco 的不动点集合当且仅当 X 是 C 的 Moore 族, 即 $X = \mathcal{M}(X) =_{\text{def}} \{\bigwedge Y \mid Y \subseteq X\}$ (其中 $\bigwedge \emptyset = \top \in X$). 甚至, 给定 $\rho \in uco(C)$, $\langle \rho(C), \leq \rangle$ 是 C 的完备 meet 子格. 根据抽象域的精确性, 可确定抽象解释的序. 假设 $A_1 \in uco(C), A_2 \in uco(C)$, A_1 比 A_2 更加精确 (或说 A_1 比 A_2 更具体, 或 A_2 比 A_1 更抽象) 当且仅当 $A_1 \sqsubseteq A_2$.

在抽象域中 lub 和 glb 有如下操作: 假定

$\{A_n\}_{n \in \mathbb{N}} \subseteq uco(C)$, (1) $\sqcup_{n \in \mathbb{N}} A_n$ 是在所有 A_n 抽象域最具体的, 即最小公共抽象. (2) $\sqcap_{n \in \mathbb{N}} A_n$ 是在所有 A_n 抽象域中最抽象的.

2.3 计算树 CTL 逻辑的语法和语义

CTL 公式是由原子命题、布尔运算符、时态算子及路径量词构成. 时态算子包括 G、F、R、U 和 X, 路径量词包括 A 和 E, 加在时态算子前. CTL 语法可以描述为

(1) 每个原子命题是一个 CTL 公式;

(2) 如果 φ 和 ψ 是 CTL 公式, 那么 $\neg\varphi$ 、 $\varphi \vee \psi$ 、 $AX\varphi$ 、 $EX\varphi$ 、 $AG\varphi$ 、 $EG\varphi$ 、 $AF\varphi$ 、 $EF\varphi$ 、 $AU(\varphi, \psi)$ 、 $EU(\varphi, \psi)$ 、 $AR(\varphi, \psi)$ 、 $ER(\varphi, \psi)$ 都是 CTL 公式.

对于任何 CTL 公式都可以用 \neg 、 \vee 、 EX 、 EU 和 EG 表达, 所以对它语义描述只需考虑这些算子, 假设算子集 $OP = \{\neg, \wedge, EX, EU, EG\}$.

通常 CTL 公式的解释是在 Kripke 结构上. 给定 Kripke 结构是一个四元组 $\mathcal{K} = (\Sigma, R, AP, L)$, 其中 AP 是原子命题集合, Σ 是状态集, 且 $s \in \Sigma$, φ 和 ψ 是 CTL 公式, CTL 公式的语义解释如下:

- (1) $s \models p$ iff $p \in L(s)$;
- (2) $s \models \neg\varphi$ iff $\neg(s \models \varphi)$;
- (3) $s \models \varphi \vee \psi$ iff $(s \models \varphi) \vee (s \models \psi)$;
- (4) $s \models EX\varphi$ iff $\exists \pi \in Path(s) \pi[1] \models \varphi$;
- (5) $s \models EU(\varphi, \psi)$ iff $\exists \pi \in Path(s), (\exists j \geq 0, \pi[j] \models \psi \wedge \forall 0 \leq k < j, \pi[k] \models \varphi)$;
- (6) $s \models EG\varphi$ iff $\exists \pi \in Path(s), (\forall j \geq 0, \pi[j] \models \varphi)$,

其中 $s \models \varphi$ 表示在模型 \mathcal{K} 中状态 s 满足 CTL 公式 φ , 而满足公式 φ 的状态集可描述为 $\llbracket \varphi \rrbracket = \{s \mid s \models \varphi\}$. $Path(s)$ 表示从 s 状态出发的所有路径.

模型检验 CTL 公式 $EX\varphi$ 可通过 $\llbracket \varphi \rrbracket$ 的 pre 计算得到, 即 $EX\varphi = pre(\llbracket \varphi \rrbracket)$; CTL 公式 $EU(\varphi, \psi)$ 和 $EG\varphi$ 可通过最小和最大不动点计算得到, 即 $EU(\varphi, \psi) = \mu Z. \psi \vee (\varphi \wedge EX Z)$, $EG\varphi = \nu Z. [\varphi \wedge EX Z]$. 其它 CTL 公式都可转化为 EX 、 EU 和 EG 表达.

3 抽象语义

假定 $p \in AP$ 属于原子命题集, $f \in OP$ 是算子, 其中 $ar(f) > 0$. CTL 公式的解释采用语义结构来描述 $\mathcal{S} = (\mathcal{P}(\Sigma), I)$, 其中 Σ 是状态集, $\mathcal{P}(\Sigma)$ 为具体语义域, I 是 CTL 公式在具体语义域上的解释函数, $I(p) \in \mathcal{P}(\Sigma)$, $I(f) = \mathcal{P}(\Sigma)^{ar(f)} \rightarrow \mathcal{P}(\Sigma)$. 为了描述简便, 用黑体字 p 和 f 各自表示 $I(p)$ 和 $I(f)$, 即

$p = I(p)$, $f = I(f)$, 且 $AP =_{\text{def}} \{p \in \mathcal{P}(\Sigma) \mid p \in AP\}$, $OP =_{\text{def}} \{f: \mathcal{P}(\Sigma)^{ar(f)} \rightarrow \mathcal{P}(\Sigma) \mid f \in OP\}$, 具体语义函数 $\llbracket \cdot \rrbracket_I: CTL \rightarrow \mathcal{P}(\Sigma)$ 表示语义结构中公式 $\varphi \in CTL$ 为真的状态集, 诸如 $\llbracket p \rrbracket_I = p$, $\llbracket f(\varphi_1, \dots, \varphi_n) \rrbracket_I = f(\llbracket \varphi_1 \rrbracket_I, \dots, \llbracket \varphi_n \rrbracket_I)$. 假定 f 是 n 元的算子, 即 $ar(f) = n$, 当对任意 CTL 公式 $\varphi_1, \dots, \varphi_n \in CTL$, 存在 $\varphi \in CTL$ 使得 $f(\llbracket \varphi_1 \rrbracket_I, \dots, \llbracket \varphi_n \rrbracket_I) = \llbracket \varphi \rrbracket_I$, 则 f 在语言 CTL 是封闭的. 通常在典型的模型检验中, 语义结构常使用 Kripke 结构 $\mathcal{K} = (\Sigma, R, AP, L)$ 来描述, OP 中的算子是定义在 \mathcal{K} 上路径和标准算子的逻辑操作, 例如 $I(\wedge) = \cap$, $I(\neg) = \mathcal{C}$, $I(EX) = pre_R$, $I(EU) = \text{"Existential Until"}$, $I(EG) = \text{"Existential Always"}$.

根据抽象解释方法, 抽象语义结构 $\mathcal{S}^\# = (A, I^\#)$, 其中 A 为抽象语义域, 由抽象解释 $GI(\mathcal{P}(\Sigma), \alpha, \gamma, A)$ 获得, 对 $p \in AP$ 和 $f \in OP$, $p^\# = I^\#(p) \in A$ 且 $f^\# = I^\#(f): A^{ar(f)} \rightarrow A$, 由 A 导出的抽象语义函数 $\llbracket \cdot \rrbracket_I^\#: CTL \rightarrow A$, 其中 $\llbracket p \rrbracket_I^\# = p^\#$, $\llbracket f(\varphi_1, \dots, \varphi_n) \rrbracket_I^\# = f^\#(\llbracket \varphi_1 \rrbracket_I^\#, \dots, \llbracket \varphi_n \rrbracket_I^\#)$. 假定抽象域 $\mu \in uco(\mathcal{P}(\Sigma))$, 对于任意具体语义操作 $f: \mathcal{P}(\Sigma)^{ar(f)} \rightarrow \mathcal{P}(\Sigma)$, 由抽象域 μ 导出的 f 最正确的近似 $\mu \circ f: \mu^{ar(f)} \rightarrow \mu$. 通过抽象域 μ 导出的抽象语义函数 $\llbracket \cdot \rrbracket_I^\#: CTL \rightarrow \mu$, 对于任意公式 $\varphi \in CTL$, 抽象值 $\llbracket \varphi \rrbracket_I^\#$ 属于 μ . 抽象语义函数 $\llbracket \cdot \rrbracket_I^\#$ 可通过具体解释函数最正确近似定义如下:

$\llbracket p \rrbracket_I^\# = \mu(p)$ 和 $\llbracket f(\varphi_1, \dots, \varphi_n) \rrbracket_I^\# = \mu(f(\llbracket \varphi_1 \rrbracket_I^\#, \dots, \llbracket \varphi_n \rrbracket_I^\#))$.

通常时序算子 f 可通过其它时序算子 g 的最小/最大不动点表示, 如 $f = \lambda X. lfp(\lambda Y. g(X, Y))$, 然而最正确近似 $\alpha \circ f \circ \gamma$ 可能不能使用最小/最大不动点特征. 例如 $EU(X, Y) = lfp(\lambda Z. Y \cup (X \cap EX Z))$, 其中 EX 的解释 $EX = I(EX) = pre_R$ 是在具体 Kripke 结构的标准前向转换. EU 在抽象域 A 上的最正确近似是 $\alpha \circ EU \circ \gamma: A \rightarrow A$. 然而此定义无法使用最小不动点来计算. 通过抽象 Kripke 结构 $\mathcal{A} = (P, R^\#, AP, L^\#)$, 可使用最小不动点 $lfp(\lambda Z^\#. Y^\# \cup (X^\# \cap EX^\# Z^\#))$ 去计算. 在抽象状态空间 P 上, $X^\#$ 和 $Y^\#$ 是 P 中的块集, \cup 和 \cap 是块集的并与交, $EX^\# = pre_{R^\#}$ 是在抽象模型上的前向转换. 从而. $\lambda(X, Y). lfp(\lambda Z. Y \cup (X \cap EX Z))$ 的最正确近似能用 $\lambda(X^\#, Y^\#). lfp(\lambda Z^\#. Y^\# \cup (X^\# \cap EX^\# Z^\#))$ 表示, 且保留与具体不动点函数相同的形式. 同理, $EG = \lambda X. gfp(\lambda Z. (X \cap EX Z))$ 的最正确近似为 $\lambda X^\#. gfp(\lambda Z^\#. (X^\# \cap EX^\# Z^\#))$.

4 构造划分且强保留的抽象模型

4.1 划分抽象

假定 Σ 是状态集, $\mu \in uco(\mathcal{P}(\Sigma)_{\subseteq})$, 则 μ 是在 Σ 上识别不能被闭包 μ 区别的某些状态, 即那些状态属于闭包 μ 相同的不动点集合. 形式化定义如下:

$$s \equiv_{\mu} s' \Leftrightarrow \forall S \in \mu. (s \in S \Leftrightarrow s' \in S).$$

显而易见 $s \equiv_{\mu} s'$ 当且仅当 $\mu(\{s\}) = \mu(\{s'\})$. 对任意 $s \in \Sigma$, $[s]_{\mu} =_{\text{def}} \{s' \in \Sigma \mid \mu(\{s\}) = \mu(\{s'\})\}$ 表示由抽象域 μ 导出的含有 s 的划分块.

定义 1. 给定抽象域 $\mu \in uco(\mathcal{P}(\Sigma)_{\subseteq})$, 如果 $\mu = \mathcal{P}(\mu) =_{\text{def}} \bigcap \{ \eta \in uco(\mathcal{P}(\Sigma)) \mid \equiv_{\eta} = \equiv_{\mu} \}$, 那么 μ 是可划分的.

操作符 \mathcal{P} 表示抽象域的精化, 称为可划分的 shell 精化. 给定 GI $\mathcal{G} = (\mathcal{P}(\Sigma)_{\subseteq}, \alpha, \gamma, A)$, 用 $\mathcal{P}(\mathcal{G})$ 表示抽象域 A 是可划分的抽象解释. 文献[14]证明 $\mathcal{P}(\mathcal{G})$ 是 \mathcal{G} 的最小划分精化, 即 $\mathcal{P}(\mathcal{G}) \sqsubseteq \mathcal{G}$, 且对任意划分 $\mathcal{G}' \sqsubseteq \mathcal{G}$, $\mathcal{G}' \sqsubseteq \mathcal{P}(\mathcal{G})$.

为了便于描述, 设 $uco^P(\mathcal{P}(\Sigma))$ 表示所有可划分的闭包集合, 定义 $par: uco^P(\mathcal{P}(\Sigma)) \rightarrow Part(\Sigma)$ 表示从可划分的闭包获得一个划分, $par(\mu) = \{[s]_{\mu} \mid s \in \Sigma\}$. 因此 $par(\mu) \in Part(\Sigma)$ 表示与抽象域 $\mu \in uco^P(\mathcal{P}(\Sigma))$ 相对应的划分. 参考文献[15], 可得如下性质.

性质 1. 让 $\mu \in uco(\mathcal{P}(\Sigma))$, 那么 $\mu \in uco^P(\mathcal{P}(\Sigma))$ 当且仅当 μ 是可加的, 且 $\{\mu(\{s\})\}_{s \in \Sigma}$ 是 Σ 的一个划分, 即 $par(\mu) = \{[s]_{\mu} \mid s \in \Sigma\} = \{\mu(\{s\})\}_{s \in \Sigma}$.

例如 $\Sigma = \{1, 2, 3\}$, $\mu_1 = \{1, 12, 13, 123\}$, $\mu_2 = \{\emptyset, 1, 2, 123\}$, $\mu_3 = \mathcal{P}(\{1, 2, 3\})$, μ_1 和 μ_2 不是可划分的闭包, 因为 $\{\mu_1(\{s\})\}_{s \in \Sigma} = \{1, 12, 13\}$ 和 $\{\mu_2(\{s\})\}_{s \in \Sigma} = \{1, 2, 123\}$ 不是状态集 Σ 的一个划分. 根据定义 1, μ_i 可划分的 shell 精化 $\mathcal{P}(\mu_i) = \mathcal{P}(\{1, 2, 3\})$, 其中 $1 \leq i \leq 3$. 然而对于闭包 $\{\emptyset, 12, 3, 123\}$ 是可划分的, 其划分为 $\{12, 3\}$.

对偶地, 定义 $pcl: par(\Sigma) \rightarrow uco^P(\mathcal{P}(\Sigma))$ 表示从一个划分映射到可划分的闭包, $pcl(P) =_{\text{def}} \mathcal{P}(\mathcal{M}(P)) = \lambda S \in \mathcal{P}(\Sigma) \cup \{B \in P \mid S \cap B \neq \emptyset\}$. 考虑闭包的不动点集合, 通过划分 P 中块的所有可能的并, 则可获得可划分的闭包, 即 $pcl(P) = \{\bigcup_{n \in N} B_n \mid \{B_n\}_{n \in N} \subseteq P\}$.

定理 1. $(uco^P(\mathcal{P}(\Sigma))_{\subseteq}, par, pcl, Part(\Sigma)_{\subseteq})$ 是一个 GI.

证明. 为了证明 $(uco^P(\mathcal{P}(\Sigma))_{\subseteq}, par, pcl, Part(\Sigma)_{\subseteq})$ 是一个 GI, 首先需证明 $(uco^P(\mathcal{P}(\Sigma))_{\subseteq}, par, pcl, Part(\Sigma)_{\subseteq})$ 是一个 Galois 连接, 即对任意 $\mu \in uco^P(\mathcal{P}(\Sigma))$ 和 $P \in Part(\Sigma)$, $P \times par(\mu) \Leftrightarrow pcl(P) \sqsubseteq \mu$.

(\Rightarrow) 对 $S \in \mathcal{P}(\Sigma)$, 证明 $pcl(P)(S) \subseteq \mu(S)$, 即证明 $\forall s \in pcl(P)(S), s \in \mu(S)$. 假定 $s \in pcl(P)(S)$, 则存在 $B \in P$ 使得 $s \in B$ 且 $S \cap B \neq \emptyset$. 让 $q \in S \cap B$, 由于 $P \leq par(\mu)$, 存在块 $[r]_{\mu} \in par(\mu)$ 使得 $B \subseteq [r]_{\mu}$. 那么对任意 $x \in B, \mu(\{x\}) = \mu(\{r\})$, 从而 $\mu(\{s\}) = \mu(\{q\}) \subseteq \mu(S)$, 证得 $s \in \mu(S)$.

(\Leftarrow) 考虑块 $B \in P$ 且 $s \in B$. 证明 $B \subseteq [s]_{\mu}$, 即 $s', s'' \in B$, 则 $\mu(\{s'\}) = \mu(\{s''\})$. 由于 $pcl(P) \mu$. 对 $s', s'' \in B, pcl(P)(s') = B \subseteq \mu(\{s'\})$, 则 $s'' \in \mu(\{s'\})$, 因此 $\mu(\{s''\}) \subseteq \mu(\mu(\{s'\})) = \mu(\{s'\})$, 同样地, $\mu(\{s'\}) \subseteq \mu(\{s''\})$, 所以 $\mu(\{s'\}) = \mu(\{s''\})$.

由于 pcl 是 1-1 的映射, 所以 $(uco^P(\mathcal{P}(\Sigma))_{\subseteq}, par, pcl, Part(\Sigma)_{\subseteq})$ 是一个 GI. 证毕.

根据定理 1, 可知 $pcl \circ par$ 是 $uco^P(\mathcal{P}(\Sigma))_{\subseteq}$ 的向上闭包. 通过定义 $\mu \in uco^P(\mathcal{P}(\Sigma))$ 当且仅当 $\mu = pcl(P)$, 其中 $P \in Part(\Sigma)$ 是一个划分, 依据 GI 的性质, 可表示为 $pcl(par(\mu)) = \mu$. 从而可得 $\mu \in uco^P(\mathcal{P}(\Sigma))$ 当且仅当 $pcl(par(\mu)) = \mu$.

定理 2. 如果 $\mu \in uco(\mathcal{P}(\Sigma))$, 那么 $\mu \in uco^P(\mathcal{P}(\Sigma))$ 当且仅当 μ 是 \mathcal{C} -完备.

证明.

(\Rightarrow) 假定 $\mu \in uco^P(\mathcal{P}(\Sigma))$, 根据性质 1, μ 是可加的, 即如果 $S \in \mu$, 那么 $S = \bigcup_{s \in S} \mu(\{s\})$, 可得 $\mathcal{C}(S) = \bigcap_{s \in S} \mathcal{C}(\mu(\{s\}))$. 并且对于 $s, s' \in \Sigma, s \notin \mu(\{s'\}) \Leftrightarrow \mu(\{s'\}) \cap \mu(\{s\}) = \emptyset$. 因此可得, $\mu(\mathcal{C}(\mu(\{s\}))) = \mu(\{s' \in \Sigma \mid s' \notin \mu(\{s\})\}) = \bigcup \{\mu(\{s'\}) \mid s' \notin \mu(\{s\})\} = \bigcup \{\mu(\{s'\}) \mid \mu(\{s'\}) \cap \mu(\{s\}) = \emptyset\} = \bigcup \{\mu(\{s'\}) \mid \mu(\{s'\}) \subseteq \mathcal{C}(\mu(\{s\}))\} \subseteq \mathcal{C}(\mu(\{s\}))$.

(\Leftarrow) 假定 μ 是 \mathcal{C} -完备. (1) 如果 $\{S_i\}_{i \in I} \subseteq \mu$, 则 $\bigcup_i S_i = \mathcal{C}(\bigcap_i \mathcal{C}(S_i)) \in \mu$, 故 μ 是可加的. (2) 假定 $s, r \in \Sigma, \mu(\{r\}) \cap \mu(\{s\}) \neq \emptyset, \mu(\{s\}) \setminus \mu(\{r\}) = \mu(\{s\}) \cap \mathcal{C}(\mu(\{r\})) \subset \mu(\{s\})$. 可知 $s \in \mu(\{s\})$, 令 $s \notin \mu(\{r\})$, 则 $s \in \mu(\{s\}) \setminus \mu(\{r\})$, 故 $\mu(\{s\}) \subseteq \mu(\{s\}) \setminus \mu(\{r\}) \subset \mu(\{s\})$ 自相矛盾, 所以 $\mu(\{r\}) \cap \mu(\{s\}) \neq \emptyset \Leftrightarrow \mu(\{r\}) = \mu(\{s\})$. 由(1)可知 μ 是可加的, $\Sigma = \bigcup_{s \in \Sigma} \mu(\{s\})$. 最终可得 $\{\mu(\{s\})\}_{s \in \Sigma} \in Part(\Sigma)$ 是一个划分. 综合(1)(2), 根据性质 1, 可得 $\mu \in uco^P(\mathcal{P}(\Sigma))$. 证毕.

4.2 强保留抽象

4.1 节描述了通过状态划分 $P \in Part(\Sigma)$ 获得可划分的闭包 $pcl(P) \in uco^P(\mathcal{P}(\Sigma))$, 在抽象域 $pcl(P)$ 导出的抽象语义函数为 $\llbracket \cdot \rrbracket_I^{pcl(P)}: CTL \rightarrow pcl(P)$. 定义划分 P 导出的抽象语义函数 $\llbracket \cdot \rrbracket_I^P = \llbracket \cdot \rrbracket_I^{pcl(P)}: CTL \rightarrow pcl(P)$, CTL 公式的抽象语义值是划分 P 中块的并, 即 $\llbracket \varphi \rrbracket_I^P = \{ \cup B \mid B \in P \text{ 且 } B \models \varphi \}$. 如果划分 P 对 CTL 是强保留的, 则抽象语义 $\llbracket \cdot \rrbracket_I^P$ 与具体语义 $\llbracket \cdot \rrbracket_I$ 是一致的.

定理 3. 如果 $P \in Part(\Sigma)$ 对 CTL 是强保留的, 那么 $\forall \varphi \in CTL, \llbracket \varphi \rrbracket_I = \llbracket \varphi \rrbracket_I^{pcl(P)}$.

证明. 假设 $\mu = pcl(P)$, 令 $\varphi \in CTL$, 根据抽象语义函数 $\llbracket \cdot \rrbracket_I^\mu$ 的最正确近似, 采用结构归纳证明如下:

- (1) $\varphi \equiv p \in AP: \llbracket p \rrbracket_I^\mu = \mu(\llbracket p \rrbracket_I) = \llbracket p \rrbracket_I$;
- (2) $\varphi \equiv \neg \varphi: \llbracket \neg \varphi \rrbracket_I^\mu = \mu(\mathcal{C}(\llbracket \varphi \rrbracket_I^\mu)) = \mu(\mathcal{C}(\llbracket \varphi \rrbracket_I)) = \mathcal{C}(\llbracket \varphi \rrbracket_I) = \llbracket \neg \varphi \rrbracket_I$;
- (3) $\varphi \equiv \varphi \wedge \psi: \llbracket \varphi \wedge \psi \rrbracket_I^\mu = \mu(\llbracket \varphi \rrbracket_I^\mu \cap \llbracket \psi \rrbracket_I^\mu) = \mu(\llbracket \varphi \rrbracket_I \cap \llbracket \psi \rrbracket_I) = \llbracket \varphi \wedge \psi \rrbracket_I$;
- (4) $\varphi \equiv EX \varphi: \llbracket EX \varphi \rrbracket_I^\mu = \mu(pre_R(\llbracket \varphi \rrbracket_I^\mu)) = \mu(pre_R(\llbracket \varphi \rrbracket_I)) = pre_R(\llbracket \varphi \rrbracket_I) = \llbracket EX \varphi \rrbracket_I$;
- (5) $\varphi \equiv EU(\varphi, \psi): \llbracket EU(\varphi, \psi) \rrbracket_I^\mu = \mu(lfp(\lambda Z. \llbracket \varphi \rrbracket_I^\mu \cup (\llbracket \psi \rrbracket_I^\mu \cap \llbracket EX Z \rrbracket_I^\mu))) = \mu(lfp(\lambda Z. \llbracket \varphi \rrbracket_I \cup (\llbracket \psi \rrbracket_I \cap \llbracket EX Z \rrbracket_I))) = lfp(\lambda Z. \llbracket \varphi \rrbracket_I \cup (\llbracket \psi \rrbracket_I \cap \llbracket EX Z \rrbracket_I)) = \llbracket EU(\varphi, \psi) \rrbracket_I$;
- (6) $\varphi \equiv EG \varphi: \llbracket EG \varphi \rrbracket_I^\mu = \mu(gfp(\lambda Z. \llbracket \varphi \rrbracket_I^\mu \cap \llbracket EX Z \rrbracket_I^\mu)) = \mu(gfp(\lambda Z. \llbracket \varphi \rrbracket_I \cap \llbracket EX Z \rrbracket_I)) = GFP(\lambda Z. \llbracket \varphi \rrbracket_I \cap \llbracket EX Z \rrbracket_I) = \llbracket EG \varphi \rrbracket_I$. 证毕.

推论 1. 假定 $P \in Part(\Sigma)$, 则 P 对 CTL 公式是强保留的当且仅当 $\forall \varphi \in CTL, s \in \Sigma, s \in \llbracket \varphi \rrbracket_I \Leftrightarrow pcl(P)(\{s\}) \subseteq \llbracket \varphi \rrbracket_I^P$.

从推论 1 可得, 划分 $P \in Part(\Sigma)$ 对 CTL 和 I 是强保留的当且仅当具体状态 $s \in \Sigma$ 满足公式 $\varphi \in CTL$, 即 $s \in \llbracket \varphi \rrbracket_I$, 等价于 s 对应 P 中的抽象状态, 即容纳 s 的块 $pcl(P)(\{s\})$ 是抽象域 $pcl(P)$ 的一个元素, 且小于或等于 CTL 公式的抽象语义值 $\llbracket \varphi \rrbracket_I^P$. 给定 $P \in Part(\Sigma)$, 通过抽象 Kripke 结构 (P, R^P, AP, L^P) 导出抽象语义 $\llbracket \cdot \rrbracket_I^P: CTL \rightarrow \mathcal{P}(P)$. 当对任何 $s \in \Sigma$ 和 $\varphi \in CTL, s \in \llbracket \varphi \rrbracket_I$ 当且仅当 $\alpha_P(\{s\}) \in \llbracket \varphi \rrbracket_I^P$, 那么划分 $P \in Part(\Sigma)$ 是对 CTL 强保留. 文献[8, 9, 14]通过等价关系 \sim_{CTL} 给出最粗糙强保留划分 $P_{CTL}, s_1 \sim_{CTL} s_2$ 当且仅当 $\forall \varphi \in CTL, s_1 \in \llbracket \varphi \rrbracket_I \Leftrightarrow s_2 \in \llbracket \varphi \rrbracket_I$. 通过具体语义函数 $\llbracket \cdot \rrbracket_I: CTL \rightarrow \mathcal{P}(\Sigma)$ 导出状态

划分 $P_{CTL} \in Part(\Sigma)$, P_{CTL} 是最优的强保留抽象划分, 即最粗糙的强保留抽象划分.

定义 2. 给定 CTL 公式和 Kripke 结构 $\mathcal{K} = (\Sigma, R, AP, L)$, 假设 $\mathcal{S} = (\mathcal{P}(\Sigma), I)$ 是具体语义结构, $\llbracket \cdot \rrbracket_I$ 是由 I 导出的具体语义函数. 让 $\mathcal{S}^\mu = (\mu, I^\mu)$ 是抽象语义结构, 其中 $\mu \in uco(\mathcal{P}(\Sigma))$, $\llbracket \cdot \rrbracket_I^\mu: CTL \rightarrow \mu$ 是对应的抽象语义函数. 对任意 $S \in \mathcal{P}(\Sigma)$ 和 $\varphi \in CTL$, 如果 $S \subseteq \llbracket \varphi \rrbracket_I \Leftrightarrow \mu(S) \subseteq \llbracket \varphi \rrbracket_I^\mu$, 那么 $\llbracket \cdot \rrbracket_I^\mu$ 对 CTL 公式是强保留的.

当 $\llbracket \cdot \rrbracket_I^\mu$ 对 CTL 和 I 是强保留的, 则闭包 $\mu \in uco(\mathcal{P}(\Sigma))$ 对 CTL 和 I 是强保留的.

定义 3. $\llbracket \cdot \rrbracket_I: CTL \rightarrow \mathcal{P}(\Sigma)$ 是具体语义函数, 对任意 $S \subseteq \Sigma, S \models \varphi$ 当且仅当 $\forall s \in S, s \models \varphi$ 当且仅当 $S \subseteq \llbracket \varphi \rrbracket_I$, 定义对 CTL 最优的强保留闭包 $\mu_{CTL}: \mathcal{P}(\Sigma) \rightarrow \mathcal{P}(\Sigma)$ 如下:

$$\mu_{CTL}(S) =_{\text{def}} \bigcup \{ T \in \mathcal{P}(\Sigma) \mid \forall \varphi \in CTL, S \models \varphi \Leftrightarrow T \models \varphi \}.$$

性质 2. $\mu \in uco(\mathcal{P}(\Sigma))$, 那么 μ 对 CTL 和 I 是强保留的当且仅当 $\mu \sqsubseteq \mu_{CTL}$.

实际上 μ_{CTL} 是最优的强保留抽象域, 即最抽象的强保留闭包. $\mu_{CTL} = \sqcup \{ \mu \in uco(\mathcal{P}(\Sigma)) \mid \mu \text{ 是对 CTL 和 } I \text{ 强保留} \}$, 对应最优的强保留抽象划分 $P_{CTL} = par(\mu_{CTL})$.

5 闭包完备性与强保留

假定 $f: C^n \rightarrow C$ 表示具体语义函数, 其中 $n \geq 0$, C 是完备格, 表示具体语义域 $\langle C, f \rangle$ 的抽象解释为 $\langle A, f^\# \rangle$, 其中 A 通过 GI (C, α, γ, A) 获得, $f^\#: A^n \rightarrow A$ 表示抽象语义函数. 文献[11]指出 $f^\#$ 是 f 的正确近似, 即 $\alpha \circ f \sqsubseteq f^\# \circ \alpha \Leftrightarrow \alpha \circ f \circ \gamma \sqsubseteq f^\#$, 换句话说, $\langle A, f^\# \rangle$ 是 $\langle C, f \rangle$ 的安全抽象, 而 $\langle A, f^\# \rangle$ 是完备的当且仅当 $\alpha \circ f = f^\# \circ \alpha$. 由于 $\alpha \circ f \sqsubseteq f^\# \circ \alpha \Leftrightarrow \alpha \circ f \circ \gamma \sqsubseteq f^\#$, 那么 f 相对于抽象域 A 最正确的近似 $f^b = \alpha \circ f \circ \gamma: A^n \rightarrow A$, 则 f^b 为 f 在抽象域 A 中最正确的近似抽象. 也就是说, 给定抽象域 A , 存在抽象函数 $f^\#$ 使得 $\langle A, f^\# \rangle$ 是完备的当且仅当 $\langle A, f^b \rangle$ 是完备的. 由于 f^b 依赖抽象域 A , 故完备性依赖于抽象域的性质. 采用闭包操作描述抽象解释 $\langle A, f^\# \rangle$ 的完备性, 假定 $\mu = \gamma \circ \alpha \in uco(C)$ 表示抽象域, 则 μ 是完备抽象当且仅当 $\mu \circ f = \mu \circ f \circ \mu$. 给定任意闭包 $\mu \in uco(C)$ 是 Moore 封闭的, 那么 μ 总是前向 meet 完备.

定义 4. 让 C 是完备格, $f: C^n \rightarrow C$ 是单调语义函数, $\mu \in uco(C)$, 如果 $\mu \circ f = \mu \circ f \circ \mu$, 那么 μ 是 f -

完备的. 如果 $F \subseteq \text{Fun}(C)$, $\forall f \in F, \mu \circ f = \mu \circ f \circ \mu$, 那么 μ 是 F -完备的.

定义 5. 给定任意函数 $F \subseteq \text{Fun}(C)$, 定义 $\Gamma(C, F) =_{\text{def}} \{\mu \in \text{uco}(C) \mid \forall f \in F, \mu \circ f = \mu \circ f \circ \mu\}$ 表示所有 F -完备的闭包. 当 $F = \{f\}$ 时, 简写为 $\Gamma(C, f)$.

性质 3. (1) $\lambda x. x \in \Gamma(C, F)$; (2) 如果 $\mu \in \Gamma(C, F), \rho \sqsubseteq \mu$, 那么 $\rho \in \Gamma(C, F)$.

性质 4. 假定 $F \subseteq \text{Fun}(C), \mu \in \text{uco}(C), \varepsilon_F(\mu) =_{\text{def}} \sqcup \{\rho \in \text{uco}(C) \mid \rho \sqsubseteq \mu, \rho \in \Gamma(C, F)\}$ 和 $\kappa_F(\mu) =_{\text{def}} \sqcap \{\eta \in \text{uco}(C) \mid \mu \sqsubseteq \eta, \eta \in \Gamma(C, F)\}$ 是 F -完备的.

性质 4 是文献[14]中引理 5.1 的扩展. 操作符 $\varepsilon_F: \text{uco}(C) \rightarrow \text{uco}(C)$ 表示最小 F -完备精化, $\varepsilon_F(\mu)$ 是比 μ 更具体域 ρ 的最小公共抽象. 操作符 $\kappa_F: \text{uco}(C) \rightarrow \text{uco}(C)$ 表示求解 F -完备核, $\kappa_F(\mu)$ 是相对于 μ 的最抽象完备域.

定义 6. 如果 $\kappa_F(\mu) \in \Gamma(C, F)$, 那么 $\kappa_F(\mu)$ 称为 μ 的 F -完备核. 对偶地, 如果 $\varepsilon_F(\mu) \in \Gamma(C, F)$, 那么 $\varepsilon_F(\mu)$ 称为 μ 的最小 F -完备精化.

性质 5^[29]. $F \subseteq \text{Fun}(C)$, 如果 $\mu \in \text{uco}(C)$, μ 的最小 F -完备精化 $\varepsilon_F(\mu)$ 总是存在. 如果 $\mu \in \Gamma(C, F)$, μ 的 F -完备核 $\kappa_F(\mu)$ 总是存在.

对于 $\mu \in \Gamma(C, F)$ 是完备闭包, 则如何求解完备核, 本文不作考虑. 本文主要描述给定任意一个闭包, 如何求解最小完备精化. 假定 $\mu \in \text{uco}(\mathcal{P}(\Sigma))$, 为了证明 μ 是对 CTL 完备的当且仅当 μ 是 $\{\neg, \text{EX}\}$ -完备的, 先给出如下引理.

引理 1. 假定 $\mu \in \text{uco}(\mathcal{P}(\Sigma)), f: \mathcal{P}(\Sigma) \rightarrow \mathcal{P}(\Sigma)$ 是单调函数, 如果 μ 对 f 是完备, 那么 $\text{gfp}(\mu \circ f) = \mu(\text{gfp}(f)) = \text{gfp}(f)$, 且 $\text{lfp}(\mu \circ f) = \mu(\text{lfp}(f)) = \text{lfp}(f)$.

引理 2. 假定 $\mu \in \text{uco}(\mathcal{P}(\Sigma)), \mu$ 对 CTL 是完备的当且仅当 μ 对 CTL 上的标准算子集是完备的.

根据引理 2 可知, μ 对 CTL 是完备的, 只需满足 μ 对 CTL 上的标准算子集是完备的. 对 CTL 上的标准算子集 $OP_{\text{CTL}} = \{\neg, \wedge, \vee, \text{EX}, \text{AX}, \text{EU}, \text{AU}, \text{AG}, \text{EG}, \text{AF}, \text{EF}, \text{AR}, \text{ER}\}$, 然而任何 CTL 公式都可用 $\neg, \vee, \text{EX}, \text{EU}$ 和 EG 表达, 故仅需考虑这些算子 $OP = \{\neg, \vee, \text{EX}, \text{EU}, \text{EG}\}$.

推论 2. 假定 $\mu \in \text{uco}(\mathcal{P}(\Sigma)), \mu$ 是 OP -完备的, 则 μ 是对 CTL 性质强保留的.

定理 4. 假定算子集 $OP = \{\neg, \vee, \text{EX}, \text{EU}, \text{EG}\}, \mu \in \text{uco}(\mathcal{P}(\Sigma)), \mu$ 是 OP -完备的当且仅当 μ 是 $\{C, \text{pre}_R\}$ -完备的.

证明. 为了证明 μ 是 OP -完备的当且仅当 μ

是 $\{C, \text{pre}_R\}$ -完备的, 只需证明 μ 对 OP 中其它算子 $\{\vee, \text{EU}, \text{EG}\}$ 的完备性可用 μ 对算子 $\{C, \text{pre}_R\}$ 的完备性表示.

(1) \vee : 根据定理 2, μ 是 C -完备的, 则 $\mu \in \text{uco}^P(\mathcal{P}(\Sigma))$, 根据性质 1, μ 是可加的, 即 $S = \bigcup_{s \in S} \mu(\{s\}) \mu(\bigcup_{s \in S} \mu(\{s\})) = \mu(S) = \mu(\bigcup_{s \in S} \{s\})$. 故 μ 是 \bigcup -完备的.

(2) EU : 通过不动点特征表示 $\text{EU} = \lambda(X, Y). \text{lfp}(\lambda Z. Y \cup (X \cap \text{EX } Z)) = \lambda(X, Y). \text{lfp}(\lambda Z. Y \cup (X \cap \text{pre}_R(Z)))$. 为了证明 μ 是 EU -完备的, 则需证明 $\mu \circ \text{EU} \circ \mu = \text{EU} \circ \mu$, 即 $\mu(\text{lfp}(\lambda Z. \mu(Y) \cup (\mu(X) \cap \text{pre}_R(Z)))) = \text{lfp}(\lambda Z. \mu(Y) \cup (\mu(X) \cap \text{pre}_R(Z)))$. 考虑 $\mu(\mu(Y) \cup (\mu(X) \cap \text{pre}_R(\mu(Z)))) = \mu(\mu(Y) \cup (\mu(X) \cap \mu(\text{pre}_R(\mu(Z)))) = \mu(Y) \cup \mu(\mu(X) \cap \mu(\text{pre}_R(\mu(Z)))) = \mu(Y) \cup (\mu(X) \cap \mu(\text{pre}_R(\mu(Z)))) = \mu(Y) \cup (\mu(X) \cap \text{pre}_R(\mu(Z)))$. 根据引理 1, 可得 $\mu(\text{lfp}(\lambda Z. \mu(Y) \cup (\mu(X) \cap \text{pre}_R(Z)))) = \text{lfp}(\lambda Z. \mu(Y) \cup (\mu(X) \cap \text{pre}_R(Z)))$.

(3) EG : 通过不动点特征表示 $\text{EG} = \lambda X. \text{gfp}(\lambda Z. X \cap \text{EX } Z) = \lambda X. \text{gfp}(\lambda Z. X \cap \text{pre}_R(Z))$. 证明类似于 EU . 证毕.

性质 6. 假定 $\mu \in \text{uco}(\mathcal{P}(\Sigma)), \mathcal{S}'' = (\mu, I'')$ 是抽象语义结构, $\llbracket \cdot \rrbracket_t'': \text{CTL} \rightarrow \mu$ 是对应的抽象语义函数, \mathcal{S}'' 对 CTL 性质强保留当且仅当 $(\mu, \llbracket \cdot \rrbracket_t'')$ 是完备的.

性质 6 给出了性质强保留与抽象完备性之间的关系. 根据定理 2 可得, 如果 μ 是可划分的闭包, 则 μ 是 C -完备的. 根据引理 2 和定理 4, 如果 μ 是 $\{C, \text{pre}_R\}$ -完备的, 则 μ 对 CTL 是完备的. 因此, 如果 μ 是可划分的闭包, 只需精化 μ 使 pre_R -完备, 那么精化的 μ 对 CTL 是完备的, 且对 CTL 性质强保留.

假设 $\mu \in \text{uco}(\mathcal{P}(\Sigma))$, 根据性质 5 可得 $\varepsilon_{\{\text{pre}_R\}}(\mu)$ 是 μ 的最优 pre_R -完备精化. 参考文献[14]中引理 5.1 和 5.2, 可得 μ 的最优可划分且完备精化 $\varepsilon_{\{C, \text{pre}_R\}}(\mu) = \text{gfp}(\lambda \rho. \mathcal{M}(\mathcal{P}(\rho) \sqcap \text{pre}_R(\rho)))$. 在有限状态系统中, 算子 $\lambda \rho. \mathcal{M}(\mu \sqcap \mathcal{P}(\rho) \sqcap \text{pre}_R(\rho))$: $\text{uco}(\mathcal{P}(\Sigma)) \rightarrow \text{uco}(\mathcal{P}(\Sigma))$ 是平凡的 ω 连续, 其不动点能通过 Kleene 迭代序列计算. 由于状态集 Σ 是有限的, 则其递归定义为 $\mu_0 =_{\text{def}} \mu, \mu_1 =_{\text{def}} \mathcal{M}(\mathcal{P}(\mu_0) \sqcap \text{pre}_R(\mu_0))$, 对 $i \in \mathbb{N}, \mu_{i+2} =_{\text{def}} \mathcal{M}(\mathcal{P}(\mu_{i+1}) \sqcap \text{pre}_R(\mu_{i+1} \setminus \mu_i))$, 那么存在 $n \in \mathbb{N}$, 使得 $\varepsilon_{\{C, \text{pre}_R\}}(\mu) = \mu_n$.

性质 7. 让 $\eta \in uco(\mathcal{P}(\Sigma))$, OP 是 CTL 上的算子, 则闭包 η 的最小 OP -完备精化 $\epsilon_{OP}(\eta)$ 就是对 CTL 最优的强保留闭包 μ_{CTL} , 即 $\epsilon_{OP}(\eta) = \mu_{CTL}$.

推论 2 和性质 7 从抽象解释角度描述了对 CTL 最优的强保留闭包精化算法. 给定原子命题 AP , 其相对应的初始闭包 $\mu_{AP} =_{\text{def}} \mathcal{M}(\{p \mid p \in AP\})$, 通过性质 7, 从而可得对 CTL 最优的强保留闭包 $\mu_{CTL} = \epsilon_{OP}(\mu_{AP})$. 令 $P \in \text{Part}(\Sigma)$, P 对 CTL 强保留的最优划分精化 $P_{CTL} = \text{par}(\epsilon_{OP}(\mathcal{M}(P)))$, 并且与划分 P 相关的可划分闭包 $pcl(P)$ 的最小 OP -完备精化 $\epsilon_{OP}(pcl(P))$ 等价于其划分的最优划分精化 P_{CTL} 的可划分闭包 $pcl(P_{CTL})$, 即 $pcl(P_{CTL}) = \epsilon_{OP}(pcl(P))$. 假定 $\eta \in uco(\mathcal{P}(\Sigma))$, 通过定理 4, 对 CTL 强保留的最优可划分闭包 $\mu_{CTL} = \epsilon_{OP}(\eta) = \epsilon_{\{C, pre_R\}}(\eta)$, 其相应的最优抽象划分 $P_{CTL} = \text{par}(\mu_{CTL}) = \text{par}(\epsilon_{\{C, pre_R\}}(\eta))$.

给定初始闭包 μ_{AP} , 可求得最优的抽象划分 $P_{CTL} = \text{par}(\epsilon_{\{C, pre_R\}}(\mu_{AP}))$. 为了求解抽象转换关系, 给定抽象 Kripke 结构 $\mathcal{A} = (P_{CTL}, R^\#, AP, L^\#)$ 强保留 CTL. 设 $B_1, B_2 \in P_{CTL}$, $B_1 \in pre_R^\#(\{B_2\})$ 当且仅当 $B_1 \subseteq pre_R\{B_2\}$, 也就是 $B_1 \subseteq pre_R\{B_2\}$ 当且仅当 $B_1 \xrightarrow{R^\#} B_2$, 由于 $B_1 \subseteq pre_R\{B_2\}$ 当且仅当 $B_1 \xrightarrow{R^{33}} B_2$, 故可得 $R^\# = R^{33}$, 且 $L^\#(B) = \bigcup_{s \in B} L(s) = L^3(B)$. 从而可得抽象 Kripke 结构 $\mathcal{A} = (P_{CTL}, R^{33}, AP, L^3)$ 是强保留 CTL 的.

6 实例分析

假设原子命题集 $AP = \{p, q\}$, Kripke 结构 $\mathcal{K} = (\Sigma = \{1, 2, 3, 4, 5\}, R, AP, L)$, 如图 1(a) 所示. 通过状态上的标签, 获得初始闭包 $\mu_{AP} = \mathcal{M}(\{p \mid p \in AP\}) = \{\emptyset, 123, 45, 12345\}$. 由定理 4 可知, 求解最优的可划分闭包 μ_{CTL} 仅需考虑算子 $\{C, pre_R\}$, 换句话说, 就是只考虑 μ_{AP} 在 $\{C, pre_R\}$ 上的最小完备精化. 假定 $\mu_0 = \mu_{AP}$, 通过 Kleene 不动点计算 $\epsilon_{\{C, pre_R\}}(\mu_{AP})$.

$$(1) \mu_0 = \mu_{AP} = \{\emptyset, 123, 45, 12345\},$$

$$(2) \mathcal{P}(\mu_0) = \{\emptyset, 123, 45, 12345\},$$

$$pre_R(\{\emptyset\}) = \emptyset, pre_R(\{123\}) = 12,$$

$$pre_R(\{45\}) = 345, pre_R(\{12345\}) = 12345,$$

$$pre_R(\mu_0) = \{\emptyset, 12, 345, 12345\},$$

则 $\mu_1 = \mathcal{M}(\mathcal{P}(\mu_0) \cup pre_R(\mu_0)) = \mathcal{M}(\{\emptyset, 12, 45, 123, 345, 12345\}) = \{\emptyset, 3, 12, 45, 123, 345, 12345\}$.

$$(3) \mathcal{P}(\mu_1) = \{\emptyset, 3, 12, 45, 123, 345, 1245, 12345\},$$

$$pre_R(\{12\}) = 12, pre_R(\{3\}) = 12,$$

$$pre_R(\{345\}) = 12345,$$

$$pre_R(\mu_1 \setminus \mu_0) = \{12, 12345\},$$

则 $\mu_2 = \mathcal{M}(\mathcal{P}(\mu_1) \cup pre_R(\mu_1 \setminus \mu_0)) = \mathcal{M}(\{\emptyset, 3, 12, 45, 123, 345, 1245, 12345\}) = \{\emptyset, 3, 12, 45, 123, 345, 1245, 12345\}$.

$$(4) \mathcal{P}(\mu_2) = \{\emptyset, 3, 12, 45, 123, 345, 1245, 12345\},$$

$$pre_R(\{1245\}) = 12345,$$

$$pre_R(\mu_2 \setminus \mu_1) = \{12345\},$$

则 $\mu_3 = \mathcal{M}(\mathcal{P}(\mu_2) \cup pre_R(\mu_2 \setminus \mu_1)) = \mathcal{M}(\{\emptyset, 3, 12, 45, 123, 345, 1245, 12345\}) = \mu_2$ (不动点达到).

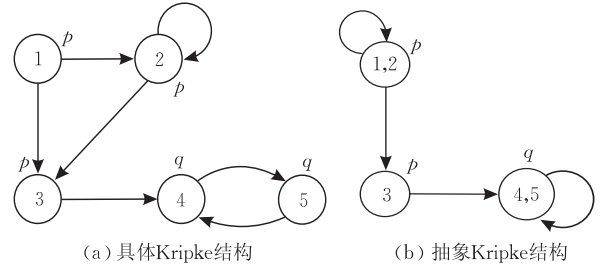


图 1

通过上述计算, μ_{AP} 的最小完备精化 $\epsilon_{OP}(\mu_{AP}) = \mu_2 = \{\emptyset, 3, 12, 45, 123, 345, 1245, 12345\}$. 根据性质 7, 可得最优的可划分闭包 $\mu_{CTL} = \{\emptyset, 3, 12, 45, 123, 345, 1245, 12345\}$ 以及 $P_{CTL} = \text{par}(\mu_{CTL}) = \{12, 3, 45\}$. 通过上节, 使用 R^{33} 求得抽象转换关系, 从而获得一个对 CTL 性质强保留的抽象 Kripke 结构, 如图 1(b) 所示.

7 结 论

抽象模型检验是一种实用且成功解决状态空间爆炸问题的验证方法^[18]. 典型的抽象模型检验是通过合并一些无法用时序语言区分的状态, 获得近似具体模型的抽象 Kripke 结构. 给定 Kripke 结构 $\mathcal{K} = (\Sigma, R, AP, L)$ 表示具体模型, 抽象模型用抽象 Kripke 结构 $\mathcal{A} = (A, R^\#, AP, L^\#)$ 描述, 其中抽象状态集合 A 由映射 $h: \Sigma \rightarrow A$ 获得, $R^\#$ 表示抽象状态转换关系. A 确定了具体状态空间 Σ 的一个划分 P_A . 本文基于抽象解释的理论, Σ 的幂集 $\mathcal{P}(\Sigma)$ 表示具体语义域, 通过 Galois 连接或闭包操作, 获得相对应的抽象域来描述状态空间 Σ 的划分. 并且基于完备抽象解释和性质强保留之间的联系, 即 A 对 CTL 算子是完备的, 那么 A 对 CTL 是性质强保留的, 从而精化抽象模型满足 CTL 性质强保留, 可转换为抽象解释中抽象域的完备精化. 并且指出了 A

对 CTL 标准算子是完备的当且仅当 A 对 $\langle \mathcal{C}, pre_R \rangle$ 是完备的。下一步工作将主要研究依赖所验证时序公式的抽象模型检验。

参 考 文 献

- [1] Clarke E M, Grumberg O, Peled D A. Model Checking. Massachusetts: MIT Press, 1999
- [2] Clarke E M, Grumberg O, Long D E. Model checking and abstraction. ACM Transactions on Programming Languages and Systems (TOPLAS), 1994, 16(5): 1512-1542
- [3] Clarke E M, Grumberg O, Jha S, Lu Y, Veith H. Counterexample-guided abstraction refinement//Proceedings of the 12th International Conference on Computer Aided Verification (CAV). LNCS 1855. Chicago, 2000: 154-169
- [4] Clarke E M, Jha S, Lu Y, Veith H. Tree-like counterexamples in model checking//Proceedings of the IEEE Symposium on Logic in Computer Science (LICS). Copenhagen, 2002: 19-29
- [5] Cousot P, Cousot R. Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints//Proceedings of the 6th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL). Los Angeles, California, 1977: 238-252
- [6] Cousot P, Cousot R. Abstract interpretation frameworks. Journal of Logic and Computation, 1992, 2(4): 511-547
- [7] Cousot P, Cousot R. Temporal abstract interpretation//Proceedings of the 27th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Programming Languages (POPL), Boston, USA, 2000: 12-25
- [8] Dams D. Abstract interpretation and partition refinement for model checking [D]. The Netherlands: Eindhoven University of Technology, 1996
- [9] Dams D. Flat Fragments of CTL and CTL * : Separating the expressive and distinguishing powers. Logic Journal of the IGPL, 1999, 7(1): 55-78
- [10] Giacobazzi R, Ranzato F. Incompleteness, counterexamples and refinements in abstract model checking//Proceedings of the 8th International Symposium on Static Analysis (SAS). LNCS 2126. Paris, 2001: 356-373
- [11] Giacobazzi R, Ranzato F, Scozzari F. Making abstract interpretations complete. Journal of the Association for Computing Machinery (ACM), 2000, 47(2): 361-416
- [12] Loiseaux C, Graf S, Sifakis J, Bouajjani A, Bensalem S. Property preserving abstractions for the verification of concurrent systems. Formal Methods in System Design, 1995, 6(1): 11-44
- [13] Ranzato F, Tapparo F. Making abstract model checking strongly preserving//Proceedings of the 8th International Symposium on Static Analysis (SAS). LNCS 2477. Madrid, Spain, 2002: 411-427
- [14] Ranzato F, Tapparo F. Strong preservation as completeness in abstract interpretation//Proceedings of the 13th European Symposium on Programming (ESOP). LNCS 2986. Barcelona, Spain, 2004: 18-32
- [15] Ranzato F, Tapparo F. An abstract interpretation-based refinement algorithm for strong preservation//Proceedings of the 11th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS). LNCS 3440. Edinburgh, UK, 2005: 140-156
- [16] Ranzato F, Tapparo F. Strong preservation of temporal fixpoint-based operators by abstract interpretation//Proceedings of the 7th International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI). LNCS 3855. Charleston, SC, USA, 2006: 332-347
- [17] Schmidt DA. Closed and logical relations for over- and underapproximation of powersets//Proceedings of the 11th International Symposium on Static Analysis (SAS). LNCS 3148. Verona, Italy, 2004: 22-37
- [18] Clarke E M, Grumberg O, Jha S, Lu Y, Veith H. Progress on the state explosion problem in model checking//Informatics-10 Years Back, 10 Years Ahead, Dagstuhl, 2001. LNCS 2000, London: Springer-Verlag, 2001: 176-194
- [19] Cousot P, Cousot R. Refining Model Checking by Abstract Interpretation. Automated software engineering Journal, 1999, 6(1): 69-95
- [20] Graf S, Saidi H. Construction of abstract state graphs with PVS//Proceedings of the 9th International Conference on Computer Aided Verification (CAV). LNCS 1254. Haifa, Israel, 1997: 72-83
- [21] Corbett J C, Dwyer M B, Hatcliff J, Laubach S, Pasareanu C S, Robby, Zheng H. Bandera: Extracting finite-state models from java source code//Proceedings of the 22nd International Conference on Software Engineering (ICSE). Limerick, Ireland, 2000: 439-448
- [22] Havelund K, Pressburger T. Model checking java programs using Java Pathfinder. International Journal on Software Tools for Technology Transfer, 2000, 2(4): 366-381
- [23] Ball T, Majumdar R, Millstein T D, Rajamani S K. Automatic predicate abstraction of C programs//Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI). Snowbird, USA, 2001: 203-213
- [24] Ball T, Rajamani S K. Automatically validating temporal safety properties of interfaces//Proceedings of the 8th International SPIN Workshop on Model Checking Software. LNCS 2057. Toronto, Canada, 2001: 103-122
- [25] Chaki S, Clarke E M, Groce A, Jha S, Veith H. Modular verification of software components in C//Proceedings of the 25th International Conference on Software Engineering (ICSE). Portland, Oregon, 2003: 385-395
- [26] Chaki S, Clarke E M, Groce A, Ouaknine J, Strichman O, Yorav K. Efficient verification of sequential and concurrent C

programs. *Formal Methods in System Design*, 2004, 25(2-3): 129-166

[27] Henzinger T A, Jhala R, Majumdar R, Sutre G. Software verification with BLAST//*Proceedings of the 10th International SPIN Workshop on Model Checking Software*. LNCS 2648. Portland, Oregon, 2003: 235-239

[28] Henzinger T A, Jhala R, Majumdar R, Sutre G. Lazy abstraction//*Proceedings of the 29th SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL)*. Portland, 2002: 58-70

[29] Giacobazzi R, Ranzato F. Completeness in abstract interpretation: A domain perspective//*Proceedings of the 6th International Conference on Algebraic Methodology and Software Technology (AMAST)*. LNCS 1349. Sydney, Australia, 1997: 231-245



QIAN Jun-Yan, born in 1973, Ph.D. , professor. His research interests include software engineering, formal verification and embedded real-time system.

XU Bao-Wen, born in 1961, Ph.D. , professor and Ph.D. supervisor. His research interests include program language, software engineering, concurrent and network software, etc.

Background

Model checking is an automatic formal verification technique for a finite state system, where all the states of the system are exhaustively enumerated and the correctness condition checked at each state, and also yields extremely useful counter examples if model checking fails. As key issue in model checking, abstraction is a method for reducing the state space of the checked system. When model checking using abstraction, the main concern is that the abstractions must be property-preserving. There are two forms of property preservation: Weak Preservation and Strong Preservation.

Strong preservation is related to completeness in abstract interpretation by studying the relationship between complete abstract interpretations, strongly preserving and spurious counterexamples of abstract model checking. In order to eliminate spurious counterexamples, a method is designed for minimally making abstract interpretations complete. It is

showed that completeness of an abstract interpretation is a property which does not depend on the abstract semantic operations but on the underlying abstract domains only. This opened up the question of making abstract interpretations complete by least refinements of abstract domains. However, this least refinement always exists and can be constructively characterized as fixed point solution of abstract domain equations.

This work is supported by the National Natural Science Foundations for Distinguished Young Scholar under grant No. 60425206, the National Natural Science Foundation of China under grant Nos. 90818027, 60633010, 60663005 and the National High Technology Research and Development Program (863 Program) of China under grant No. 2009AA01Z147.