谓词μ演算和模态图的语义一致性*

刘 剑+, 林惠民

(中国科学院 软件研究所 计算机科学重点实验室,北京 100080)

Consistency Between the Predicate μ-Calculus and Modal Graphs

LIU Jian+, LIN Hui-Min

(Key Laboratory for Computer Science, Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

+ Corresponding author: Phn: 86-10-62562796, Fax: 86-10-62563894, E-mail: ljian@ios.ac.cn

http://lcs.ios.ac.cn

Received 2003-01-03; Accepted 2003-05-27

Liu J, Lin HM. Consistency between the predicate μ-calculus and modal graphs. *Journal of Software*, 2003, 14(10):1672~1680.

http://www.jos.org.cn/1000-9825/14/1672.htm

Abstract: The modal graphs are effective graph forms for the predicate μ-calculus. The consistency between the predicate μ-calculus and the modal graphs is strictly established. Moreover, the relationship among the predicate μ-calculus, nested predicate equations and the modal graphs is discussed in detail. An optimized transformation algorithm from predicate μ-calculus formulae to nested predicate equations is presented.

Key words: fixed-point; predicate μ-calculus; nested predicate equation; modal graph

摘 要: 模态图是谓词μ演算的一种有效的图形表示形式.证明了谓词μ演算和模态图的语义一致性,详细讨论了谓词μ演算公式、嵌套谓词等式系和模态图之间的关系,并给出了一种优化的从线性公式到嵌套谓词等式系的转换算法.

关键词: 不动点;谓词μ演算;嵌套谓词等式系;模态图

中图法分类号: TP301 **文献标识码**: A

谓词μ演算是一种用来描述传值并发系统性质的逻辑.它是传统命题μ演算的一阶推广.该逻辑的提出源于命题μ演算难以直接地、显式地表达传值进程的逻辑性质.例如,在命题μ演算系统中,要表达传值进程的性质,首先要将各个数据变量的取值实例化,才能把性质转化为等价的不带数据变量的公式.而且这样的转换并不是通过简单代换就能完成的.例如,对于某些性质,要求其中出现的多个变量满足取值相等(不等)关系,此时转换得到的命题μ演算公式繁琐、冗长;相反地,若用谓词μ演算公式来表达则简单、直观.

谓词μ演算的公式分为命题和谓词两种类型.命题是取真假值的断言,谓词是从数据到命题的函数.公式可以含有谓词变量和数据变量,两种变量可以自由出现,也可以受囿出现.数据变量受全称量词、存在量词和抽象算子的约束;谓词变量受最大、最小不动点算子的约束,两种不动点算子又可相互嵌套.所有这些都使得文本表

* Supported by the National Natural Science Foundation of China under Grant No.69833020 (国家自然科学基金)

第一作者简介: 刘剑(1976－),男,云南石屏人,博士生,主要研究领域为并发系统的自动验证,模型检测方法的理论和应用.

示的线性谓词μ演算公式复杂、难懂,不利于机器处理,限制了其在自动处理方面的使用.为了便于应用谓词μ演算对传值进程进行模型检测,文献[1]提出了谓词μ演算的一种图形表示形式——模态图,并给出了从文本公式到模态图的转换算法.但是,该文献主要讨论传值进程模型检测算法,而对谓词μ演算和模态图的语义一致性(即转换过程是否正确)没有展开分析.本文的工作是借助嵌套谓词等式系来建立两者之间的联系,从而证明了转换的正确性.

本文第1节定义谓词μ演算公式的语法和语义,并证明一些相关性质.第2节定义模态图的语法和语义.为了证明谓词μ演算公式和模态图的语义关系,第3节引入嵌套谓词等式系作为一种中间表示形式.第4节详细讨论3种表示形式之间的关系;首先给出谓词闭的命题到嵌套谓词等式系的转换算法,证明该算法保持原命题和对应等式系的语义一致;其次讨论嵌套谓词等式系和模态图表示的对应关系.此外,在转换算法中,我们利用等式系的相关性质对转换过程进行了优化.最后是相关工作的比较,并对全文进行总结.

1 谓词μ演算的语法和语义

本节定义一个面向传值进程的谓词μ演算系统[1],其语义模型基于带赋值的符号迁移图(STGA)[2].限于篇幅,这里不再给出符号迁移图的定义,相关工作和记号请参见文献[1,2].

1.1 谓词μ演算的语法

谓词μ演算是对传统的命题μ演算的扩充,命题μ演算可以看作是谓词μ演算的子集.

令 Act 是抽象动作的集合,其元素用 α 表示;模态算子集 $Modop=\{[\alpha],\langle\alpha\rangle\mid\alpha\in Act\}$,其元素用 β 表示.模态算子的自由变量集和受囿变量集分别定义为 $fv([\alpha])=fv(\langle\alpha\rangle)=fv(\alpha)$, $bv([\alpha])=bv(\langle\alpha\rangle)=bv(\alpha)$.令 X 是谓词变量的集合, $X^{(n)},Y^{(n)},...\in X$ 是谓词变量,其中 $n\in\mathbb{Z}$是一个非负整数,用来标识谓词变量的目数(arity).谓词μ演算公式由如下 BNF 语法生成:

$$\varphi=b\mid\varphi\wedge\varphi\mid\varphi\vee\varphi\mid\exists x\varphi\mid\forall x\varphi\mid\beta\varphi\mid\Lambda\overline{e},$$

$$\Lambda=(\overline{x})\varphi\mid X\mid\mu X.\Lambda\mid\nu X.\Lambda.$$

公式分命题(φ)和谓词(Λ)两种类型.命题由布尔表达式、逻辑连接词∧和∨、一阶量词∀和∃以及模态词 β 构成.谓词是从数据表达式到命题的函数,可以是谓词变量 X、命题抽象 $(\overline{x})\varphi$ 或者最大/最小不动点 $\mu X.\Lambda,\nu X.\Lambda$.将谓词应用于数据表达式就得到命题 $\Lambda\overline{e}$.

在公式 $\forall x\varphi,\exists x\varphi,[c?x]\varphi,\langle c?x\rangle\varphi,(\overline{x})\varphi$ 中, x (或 $\overline{x}$)是受囿的,其辖域为 φ. φ 中出现的非受囿变量称为自由的. φ (或 Λ)中的自由数据变量集和受囿数据变量集分别记为 $fdv(\varphi),bdv(\varphi)$ (或 $fdv(\Lambda),bdv(\Lambda)$).称公式 φ (或 Λ)是数据闭的,当且仅当 $fdv(\varphi)=\varnothing$ (或 $fdv(\Lambda)=\varnothing$).本文只使用数据闭的抽象,即在 $(\overline{x})\varphi$ 中要求 $fdv(\varphi)\subseteq\{\overline{x}\}$.因此,所有的谓词都是数据闭的,且对任意 $\Lambda\overline{e}$,均有 $fdv(\Lambda\overline{e})=fdv(\overline{e})$.

在公式 $\mu X.\Lambda$ $\nu X.\Lambda$ 中,谓词变量 X 受囿,其辖域为 Λ. φ (或 Λ)中的自由谓词变量集和受囿谓词变量集分别记为 $fpv(\varphi),bpv(\varphi)$ (或 $fpv(\Lambda),bpv(\Lambda)$).同样地,称公式 φ (或 Λ)是谓词闭的,当且仅当 $fpv(\varphi)=\varnothing$ (或 $fpv(\Lambda)=\varnothing$).

为了叙述方便,用 $\phi,\phi,\phi_1,...$ 表示任意公式,即命题或谓词.用 σ 表示任意不动点算子,即 μ 或 ν.若两个公式仅在某些受囿变量名上不同,则称它们是 α 等价的.对任意公式,通过 α 变换可使其中出现的受囿变量两两不同.

1.2 谓词μ演算的语义

谓词μ演算公式以带赋值的符号迁移图作为语义模型.

令 $\mathcal{G}$ 是给定的带赋值符号迁移图,谓词μ演算公式的语义相对于 $\mathcal{G}$ 的导出标号迁移系统给出.令 S 是该标号迁移系统的状态集,定义谓词变量的取值空间 $\Omega=\{f\mid f:Val^n\to 2^S,n\in\mathbb{Z}\}$.将 2^S 上集合间的包含关系 $\subseteq$ 逐点扩充到 n 元函数空间上,定义 Ω 的偏序关系 $\sqsubseteq$ 为 $f^{(n)}\sqsubseteq g^{(n)}$ 当且仅当对任意 $\overline{v}\in Val^n$ 都有 $f^{(n)}(\overline{v})\subseteq g^{(n)}(\overline{v})$.定义运算 $\sqcap,\sqcup$ 为:对任意 $\overline{v}\in Val^n$,($f^{(n)}\sqcap g^{(n)})(\overline{v})=f^{(n)}(\overline{v})\cap g^{(n)}(\overline{v})$ 和($f^{(n)}\sqcup g^{(n)})(\overline{v})=f^{(n)}(\overline{v})\cup g^{(n)}(\overline{v})$.这样, $\langle\Omega,\sqsubseteq\rangle$ 构成一个完备格.用 $\xi,\xi',\xi_1,...$ 表示 X 上的环境(对谓词变量的赋值),其中 $\xi:X\to\Omega$ 是全函数,将目数

为 n 的谓词变量 X 映射为 $\xi(X):Val^n \to 2^S$.谓词μ演算公式的语义相对于计值 ρ 和环境 ξ 定义如下:

(1) 对命题 φ:

$$[\![b]\!]\ \rho\xi = \begin{cases} S, \rho(b) = \text{true} \\ \varnothing, \rho(b) = \text{false} \end{cases},$$

$$[\![\varphi_1 \wedge \varphi_2]\!]\ \rho\xi = [\![\varphi_1]\!]\ \rho\xi \ \wedge\ [\![\varphi_2]\!]\ \rho\xi,$$

$$[\![\varphi_1 \vee \varphi_2]\!]\ \rho\xi = [\![\varphi_1]\!]\ \rho\xi \ \vee\ [\![\varphi_2]\!]\ \rho\xi,$$

$$[\![\forall x\varphi]\!]\ \rho\xi = \bigcap_{v\in Val} [\![\varphi]\!]\ \rho\{x \mapsto v\}\xi,$$

$$[\![\exists x\varphi]\!]\ \rho\xi = \bigcup_{v\in Val} [\![\varphi]\!]\ \rho\{x \mapsto v\}\xi,$$

$$[\![\langle a\rangle\varphi]\!]\ \rho\xi = \{p \mid \exists p', p \xrightarrow{a} p' \wedge p' \in [\![\varphi]\!]\ \rho\xi\},$$

$$[\![[a]\varphi]\!]\ \rho\xi = \{p \mid \forall p', p \xrightarrow{a} p' \text{蕴含} p' \in [\![\varphi]\!]\ \rho\xi\},$$

$$[\![\langle c!e\rangle\varphi]\!]\ \rho\xi = \{p \mid \exists p', p \xrightarrow{c!\rho(e)} p' \wedge p' \in [\![\varphi]\!]\ \rho\xi\},$$

$$[\![[c!e]\varphi]\!]\ \rho\xi = \{p \mid \forall p', p \xrightarrow{c!\rho(e)} p' \text{蕴含} p' \in [\![\varphi]\!]\ \rho\xi\},$$

$$[\![\langle c?x\rangle\varphi]\!]\ \rho\xi = \{p \mid \exists p', p \xrightarrow{c?y} p' \wedge \forall v \in Val, p'[v/y] \in [\![\varphi]\!]\ \rho\{x \mapsto v\}\xi\},$$

$$[\![[c?x]\varphi]\!]\ \rho\xi = \{p \mid \forall p', p \xrightarrow{c?y} p' \text{蕴含} \forall v \in Val, p'[v/y] \in [\![\varphi]\!]\ \rho\{x \mapsto v\}\xi\},$$

$$[\![\Lambda\overline{e}]\!]\ \rho\xi = [\![\Lambda]\!]\ \rho\xi(\rho(\overline{e})).$$

(2) 对谓词 Λ:

$$[\![(\overline{x})\varphi]\!]\ \rho\xi = \lambda\overline{v}.\ [\![\varphi]\!]\ \rho\{\overline{x} \mapsto \overline{v}\}\xi,$$

$$[\![X]\!]\ \rho\xi = \xi(X),$$

$$[\![\nu X.\Lambda]\!]\ \rho\xi = \sqcup \{f : Val^n \to 2^S \mid f \sqsubseteq [\![\Lambda]\!]\ \rho\xi\{X \mapsto f\}\},$$

$$[\![\mu X.\Lambda]\!]\ \rho\xi = \sqcap \{f : Val^n \to 2^S \mid [\![\Lambda]\!]\ \rho\xi\{X \mapsto f\} \sqsubseteq f\}.$$

直观地,对给定的 $\mathcal{G}$ 和环境 ξ,ρ,语义函数 $[\![\cdot]\!]\ \rho\xi$ 将命题解释为 S 的某一个子集;将谓词解释为 Ω 中的某一个函数.

引理 1. 设 ϕ 是公式, $\{\overline{x}\} = fdv(\phi)$; ξ 是环境; ρ,ρ' 是任意计值,

$$\text{若 } \forall x \in \{\overline{x}\}, \rho(x) = \rho'(x), \text{则 } [\![\phi]\!]\ \rho\xi = [\![\phi']\!]\ \rho\xi.$$

由于公式中出现的任意谓词 Λ 都是数据闭的,因此有以下推论成立:

推论 1. 设 Λ 是任意谓词. ξ 是环境, ρ,ρ' 是任意计值,则

$$[\![\Lambda]\!]\ \rho\xi = [\![\Lambda]\!]\ \rho\xi.$$

推论 1 说明,对任意谓词 Λ,其语义和 ρ 的取值无关.在以后的叙述中,当处理到谓词的语义时,常将环境 ρ 省略.

引理 2. 设 Λ 是谓词, φ 是命题; ρ,ξ 是任意环境,则下面的等式成立:

$$[\![((\overline{x})\varphi)\overline{x}]\!]\ \rho\xi = [\![\varphi]\!]\ \rho\xi,$$

$$[\![\Lambda]\!]\ \rho\xi = [\![(\overline{y})(\Lambda\overline{y})]\!]\ \rho\xi.$$

引理 2 说明, $(\overline{x})\varphi\overline{x}$ 和 φ, $(\overline{y})(\Lambda\overline{y})$ 和 Λ 在任意环境 ρ,ξ 下取值都相等.以后我们将不区分 $(\overline{x})\varphi\overline{x}$ 和 φ 以及 $(\overline{y})(\Lambda\overline{y})$ 和 Λ.

引理 3. 设 Λ 是谓词, X 是谓词变量, ξ 是任意环境, Y 不在 $\sigma X.\Lambda$ 中自由出现,则 $[\![\sigma X.\Lambda]\!]\ \xi = [\![\sigma Y.\Lambda[X/Y]]\!]\ \xi$.其中, $\Lambda[X/Y]$ 表示将 Λ 中自由出现的 X 替换为 Y.

2 模态图

本节引入模态图的语法和语义[1].模态图是谓词μ演算闭公式的图形表示形式,目的是为了便于将谓词μ演算应用于传值进程的模型检测中.

2.1 模态图的语法

定义 1(模态图(modal graph)). 模态图是一个五元组 $\mathcal{M}=(N,E,L_N,T,r)$,其中 N 是结点的有穷集,$E\subseteq N\times N$ 是边的有穷集. $L_N=(L_V,L_O,L_D)$ 对结点 $n\in N$ 指定自由变量集 $L_V(n)$,算子 $L_O(n)\in\{\wedge,\vee,\forall x,\exists x,\theta\}\cup Modop\cup BExp$,和自然数 $L_D(n)$ (n 的嵌套深度). $r\in N$ 是 $\mathcal{M}$ 的根结点. T 是从自然数到$\{\mu,\nu\}$的映射,假定 $\{L_D(n)\mid n\in N\}$ 是一个从 0 开始的连续自然数集,且对任意 $i\geq 0, T(i)\neq T(i+1)$.称模态图 $\mathcal{M}$ 是良构的,如果它满足下列限制:

(1) 若 $L_O(n)=b\in BExp$,则 $fv(b)\subseteq L_V(n)$ 且结点 n 没有出边;

(2) 若 $L_O(n)\in\{\forall x,\exists x\}$,则 n 有且仅有一条出边 $n\to n'$ 且 $fv(n')\subseteq fv(n)\cup\{x\}$;

(3) 若 $L_O(n)=\beta\in Modop$,则 n 有且仅有一条出边 $n\to n'$ 且 $fv(n')\subseteq fv(n)\cup bv(\beta), fv(\beta)\subseteq fv(n)$;

(4) 若 $L_O(n)=\theta\equiv\overline{x}:=\overline{e}$,则 n 有且仅有一条出边 $n\to n'$ 且 $fv(e)\subseteq fv(n), fv(n')\subseteq\{\overline{x}\}$;

(5) 若 $L_O(n)\in\{\wedge,\vee\}$,则 n 有两条出边且对任意出边 $n\to n'$, $fv(n')\subseteq fv(n)$.

这里只考虑良构的模态图.

为了叙述方便,在以后的内容中对模态图 $\mathcal{M}$ 常使用简略的记法.例如,对结点 n,若 $L_O(n)=b$,则记为 $n=b$;若 $L_O(n)=\forall x$ 且有出边 $n\to n'$,则记为 $n=\forall xn$,其余情况类似.

2.2 模态图的语义

模态图 $\mathcal{M}$ 的语义相对于由给定的带赋值的符号迁移图导出的标号迁移系统给出,令 S 表示标号迁移系统的状态集.设 n 是 $\mathcal{M}$ 的任意结点, $L_V(n)$ 是 n 的变量集,则构造 n 的取值空间为 $L_V(n)$ 上的所有(partial)计值到 2^S 的映射构成的集合,记为 $\{f_n\mid f_n: L_V(n)\to Val\to 2^S\}$,这里用 f_n 表示 n 的某一取值.令所有 $n\in N$ 的取值空间构成的集合为 $\Theta=\{f_n\mid f_n: L_V(n)\to Val\to 2^S, n\in N\}$,将 2^S 上的集合包含关系 $\subseteq$ 逐点扩充到 n 元计值空间上,定义 Θ 上的偏序关系为 $f_n\sqsubseteq g_n$ 当且仅当对任意 $\rho\in L_V(n)\to Val$ 都有 $f_n(\rho)\subseteq g_n(\rho)$.这样,$\langle\Theta,\sqsubseteq\rangle$ 就构成一个完备格.记 N 上环境的集合为 $\Gamma=\{\zeta\mid\zeta: N\to\Theta\}$,用 $\zeta,\eta,\zeta',\eta',\ldots$ 表示其中的环境.对任意 $\zeta,\zeta'\in\Gamma$, $\zeta\sqsubseteq\zeta'$ 当且仅当对任意 n 和 ρ,都有 $\zeta n\rho\subseteq\zeta' n\rho$.

首先由下列规则定义结点的语义函数 $\|\cdot\|: N\to Eval\to\Gamma\to 2^S$.

$$\|b\|\rho\zeta=\begin{cases}S, & \rho(b)=\text{true}\\ \varnothing, & \rho(b)=\text{false}\end{cases},$$

$$\|\theta n'\|\rho\zeta=\zeta n'(\theta\rho),$$

$$\|\wedge_{i\in\{1,2\}} n_i\|\rho\zeta=\bigcap_{i\in\{1,2\}}\zeta n_i\rho,$$

$$\|\vee_{i\in\{1,2\}} n_i\|\rho\zeta=\bigcup_{i\in\{1,2\}}\zeta n_i\rho,$$

$$\|\forall xn'\|\rho\zeta=\bigcap_{v\in Val}\zeta n'\rho\{x\mapsto v\},$$

$$\|\exists xn'\|\rho\zeta=\bigcup_{v\in Val}\zeta n'\rho\{x\mapsto v\},$$

$$\|\langle a\rangle n'\|\rho\zeta=\{s\mid\exists s', s\xrightarrow{a}s'\text{且}s'\in\zeta n'\rho\},$$

$$\|[a]n'\|\rho\zeta=\{s\mid\forall s', s\xrightarrow{a}s'\text{蕴含}s'\in\zeta n'\rho\},$$

$$\|\langle c!e\rangle n'\|\rho\zeta=\{s\mid\exists s', s\xrightarrow{c!\rho(e)}s'\text{且}s'\in\zeta n'\rho\},$$

$$\|[c!e]n'\|\rho\zeta=\{s\mid\forall s', s\xrightarrow{c!\rho(e)}s'\text{蕴含}s'\in\zeta n'\rho\},$$

$$\|\langle c?x\rangle n'\|\rho\zeta=\{s\mid\exists s', s\xrightarrow{c?y}s'\text{且}\forall v\in Val, s'[v/y]\in\zeta n'\rho\{x\mapsto v\}\},$$

$$\|[c?x]n'\|\rho\zeta=\{s\mid\forall s', s\xrightarrow{c?y}s'\text{蕴含}\forall v\in Val, s'[v/y]\in\zeta n'\rho\{x\mapsto v\}\}.$$

设 $\mathcal{M}$ 是给定的模态图,令 $\mathcal{M}$ 的第 i 块 $B_i=\{n\in N\mid L_D(n)=i\}$.若 $j<i$,则称 B_j 是 B_i 的外部块或 B_i 是 B_j 的内部块.假定 $\mathcal{M}$ 有 $k+1$ 块 $\{B_i\mid 0\leq i\leq k\}$,其中 $T(i)=\sigma_i\in\{\mu,\nu\}$.且对任意 $0\leq i\leq k$,令 $\mathcal{B}_i=\{B_j\mid i\leq j\leq k\}$,是 B_i 到 B_{k+1} 块构成的集合.

对给定的环境 ζ 和 i,构造函数 $\boldsymbol{F}_{\mathcal{B}_i,\zeta}:\Gamma\to\Gamma$ 如下:

- 当 $i=k+1$ 时,$\boldsymbol{F}_{\mathcal{B}_{k+1},\zeta}(\eta)=\eta$.
- 当 $0\le i\le k$ 时,

$$\boldsymbol{F}_{\mathcal{B}_i,\zeta}(\eta)n\rho=\begin{cases}\|n\|\rho(\llbracket\mathcal{B}_{i+1}\rrbracket\zeta[\eta\lceil B_i]), & n\in B_i\\ (\llbracket\mathcal{B}_{i+1}\rrbracket\zeta[\eta\lceil\mathcal{B}_i])n\rho, & \text{否则}\end{cases}.$$

其中,$\eta\lceil B_i$ 表示由 η 在 B_i 中的结点上的取值构成的(partial)环境;$\zeta[\eta\lceil B_i]$表示按 $\eta\lceil B_i$ 的取值对 ζ 进行修改.由 $\|\cdot\|$ 的定义可得 $\boldsymbol{F}_{\mathcal{B}_i,\zeta}$ 是单调的,记其最大、最小不动点分别为 $\nu\boldsymbol{F}_{\mathcal{B}_i,\zeta}$ 和 $\mu\boldsymbol{F}_{\mathcal{B}_i,\zeta}$ 则定义 $\llbracket\mathcal{B}_i\rrbracket\ \zeta$ 为,若 $\sigma_i=\nu$,则 $\llbracket\mathcal{B}_i\rrbracket\ \zeta=\nu\boldsymbol{F}_{\mathcal{B}_i,\zeta}$;否则,若 $\sigma_i=\mu$,则 $\llbracket\mathcal{B}_i\rrbracket\ \zeta=\mu\boldsymbol{F}_{\mathcal{B}_i,\zeta}$.

对给定的环境 ζ,定义模态图 $\mathcal{M}$ 在 ζ 上的语义 $\llbracket\mathcal{M}\rrbracket\ \zeta$ 为 $\llbracket\mathcal{M}\rrbracket\ \zeta\stackrel{\Delta}{=}\llbracket\mathcal{B}_0\rrbracket\ \zeta$.注意到 $\mathcal{M}$ 是谓词闭的,因此,$\llbracket\mathcal{M}\rrbracket\ \zeta$ 是计值 ρ 到状态集 2^S 的函数.

3 嵌套谓词等式系

为了建立谓词μ演算和模态图之间的语义关系,本节引入嵌套谓词等式系的概念.谓词μ演算公式可以转换为语义上等价的嵌套谓词等式系,而模态图则可以视为嵌套谓词等式系的图形表示.

3.1 嵌套谓词等式系的语法语义

(带不动点类型的)谓词等式形如

$$X(\bar{x})=_\sigma\varphi.$$

其中,X 是一个谓词变量,$\bar{x}$ 是一列互不相同的数据变量,$\sigma\in\{\mu,\nu\}$;φ 是不含不动点算子的命题且 $fdv(\varphi)=\{\bar{x}\}$.

嵌套谓词等式系满足如下的 BNF 语法:

$$\mathcal{E}::=\varepsilon\mid(X(\bar{x})=_\sigma\varphi)\mathcal{E},$$

其中 ε 表示空序列.通常地,用 $\mathcal{E}\ \mathcal{E}'\ \mathcal{E}_1\ldots$表示嵌套谓词等式系,并约定在同一等式系中,任意两个等式的左边出现的谓词变量两两不同.对 $\mathcal{E}$ 其左边出现的谓词变量集记为 $lhs(\mathcal{E})$,即 $lhs((X(\bar{x})=_\sigma\varphi)\mathcal{E})=\{X\}\cup lhs(\mathcal{E})$.右边出现的谓词变量集记为 $rhs(\mathcal{E})$.称 $lhs(\mathcal{E})$中的变量为 $\mathcal{E}$ 的受囿谓词变量;称 $rhs(\mathcal{E})-lhs(\mathcal{E})$中的变量为 $\mathcal{E}$ 的自由谓词变量.对 $\mathcal{E}$ 和 $\mathcal{E}'$,当 $lhs(\mathcal{E})\cap lhs(\mathcal{E}')=\varnothing$ 时,用 $\mathcal{E}::\mathcal{E}'$表示将两个等式系联结在一起而得到的新等式系.若 $X(\bar{x})=_\sigma\varphi$ 是 $\mathcal{E}$ 中的等式,$Y\in fpv(\varphi)-\{X\}$,则称变量 X 依赖于 Y,依赖关系是可传递的;X 的依赖变量闭包集记为 $Dep(X)$.

设 $\mathcal{E}$ 是嵌套谓词等式系,ζ 是环境,则 $\mathcal{E}$ 在 ξ 上的语义(解)【$\mathcal{E}$】ξ 定义如下:

- 若 $\mathcal{E}=\varepsilon$,【$\mathcal{E}$】$\xi=\xi$;
- 若 $\mathcal{E}=(X(\bar{x})=_\upsilon\varphi)\mathcal{E}'$,【$\mathcal{E}$】$\zeta=$【$\mathcal{E}'$】$\xi\ [X/\upsilon X.(\bar{x})\varphi$【$\mathcal{E}'$】$\xi\ ]$;
- 若 $\mathcal{E}=(X(\bar{x})=_\mu\varphi)\mathcal{E}'$,【$\mathcal{E}$】$\zeta=$【$\mathcal{E}'$】$\xi\ [X/\mu X.(\bar{x})\varphi$【$\mathcal{E}'$】$\xi\ ]$.

其中:

$$\mu X.(\bar{x})\varphi\ 【\mathcal{E}'】\ \xi\stackrel{\text{def}}{=}\sqcap\{f\mid f\sqsupseteq\llbracket(\bar{x})\varphi\rrbracket\ 【\mathcal{E}'】\ \xi[X\mapsto f]\},$$

$$\upsilon X.(\bar{x})\varphi\ 【\mathcal{E}'】\ \xi\stackrel{\text{def}}{=}\sqcup\{f\mid f\sqsubseteq\llbracket(\bar{x})\varphi\rrbracket\ 【\mathcal{E}'】\ \xi[X\mapsto f]\}.$$

在上述定义中,$\llbracket(\bar{x})\varphi\rrbracket$【$\mathcal{E}'$】$\xi[X\mapsto f]$ 表示先求得环境【$\mathcal{E}'$】$\xi[X\mapsto f]$,再按 $\llbracket(\bar{x})\varphi\rrbracket$ 的语义定义求 $\llbracket(\bar{x})\varphi\rrbracket$ 在【$\mathcal{E}'$】$\xi[X\mapsto f]$ 下的值.注意到,嵌套谓词等式系是数据闭的,其语义与 ρ 无关.下面给出几个引理,其正确性容易由上述定义得证.

引理 4. 设 $\mathcal{E}_1,\mathcal{E}_2$ 是嵌套谓词等式系,满足:

$$lhs(\mathcal{E}_1)\cap lhs(\mathcal{E}_2)=\varnothing,$$
$$lhs(\mathcal{E}_1)\cap rhs(\mathcal{E}_2)=\varnothing,$$

$$lhs(\mathcal{E}_2)\cap rhs(\mathcal{E}_1)=\varnothing,$$

则【$\mathcal{E}_1$】【$\mathcal{E}_2$】ξ=【$\mathcal{E}_1::\mathcal{E}_2$】$\xi$,且对任意$X\in lhs(\mathcal{E}_1)$,(【$\mathcal{E}_1::\mathcal{E}_2$】$\xi$)($X$)=(【$\mathcal{E}_1$】$\xi$)($X$);对任意$X\in lhs(\mathcal{E}_2)$,(【$\mathcal{E}_1::\mathcal{E}_2$】$\xi$)($X$)=(【$\mathcal{E}_2$】$\xi$)($X$).

引理 5. 设ξ_1=【$\mathcal{E}_1::(X_1(\bar{x}_1)=_\sigma\varphi_1)::(X_2(\bar{x}_2)=_\sigma\varphi_2)::\mathcal{E}_2$】$\xi$,$\xi_2$=【$\mathcal{E}_1::(X_2(\bar{x}_2)=_\sigma\varphi_2)::(X_1(\bar{x}_1)=_\sigma\varphi_1)::\mathcal{E}_2$】$\xi$,则$\xi_1=\xi_2$.

引理 6. 设$X_1\notin Dep(X_2)$或$X_2\notin Dep(X_1)$,ξ_1=【$\mathcal{E}_1::(X_1(\bar{x}_1)=_{\sigma_1}\varphi_1)::(X_2(\bar{x}_2)=_{\sigma_2}\varphi_2)::\mathcal{E}_2$】$\xi$,$\xi_2$=【$\mathcal{E}_1::(X_2(\bar{x}_2)=_{\sigma_2}\varphi_2)::(X_1(\bar{x}_1)=_{\sigma_1}\varphi_1)::\mathcal{E}_2$】$\xi$,则$\xi_1=\xi_2$.

3.2 嵌套谓词等式系的语义刻画

定义环境间的偏序关系$\sqsubseteq$为:$\xi_1\sqsubseteq\xi_2$当且仅当对任意$X\in\mathcal{X},\xi_1(X)\subseteq\xi_2(X)$.所有环境在$\sqsubseteq$下构成完备格.设变量集$\mathcal{X}'\subseteq\mathcal{X},\zeta\lceil\mathcal{X}'$表示将$\xi$的值限制到$\mathcal{X}'$上得到的(partial)环境,即对任意$X\in\mathcal{X}',\xi\lceil\mathcal{X}'(X)=\xi(X)$.特别地,当$\mathcal{X}'$是只含有一个变量的集合$\{X\}$时用$\xi\lceil X$表示$\xi\lceil\{X\}$.$\xi[x\mapsto f]$表示将$\xi$在$X$处的取值换为$f$,其他保持不变.若$\mathcal{X}'\subseteq\mathcal{X}$,且$\xi'$是$\mathcal{X}'$上的环境,则$\xi\lceil\xi'$表示将$\xi$在$\mathcal{X}'$上的取值换为$\xi'$的相应取值,其他保持不变,即

$$\xi\lceil\xi'(X)=\begin{cases}\xi'(X), X\in\mathcal{X}'\\ \xi(X), X\notin\mathcal{X}'\end{cases}.$$

下面给出嵌套谓词等式系的一种语义刻画.

设$\mathcal{E}$是嵌套谓词等式系,ξ是环境,则$\mathcal{E}$在ξ上的语义可按以下方式求解:

(1) 当$\mathcal{E}=\varepsilon$时,《$\mathcal{E}$》$\xi=\xi$;

(2) 当$\mathcal{E}=(X(x)=_\sigma\varphi)\mathcal{E}'$时,构造函数$\boldsymbol{F}_{\mathcal{E},\xi}:(\mathcal{X}\to\Omega)\to(\mathcal{X}\to\Omega)$为

$$\boldsymbol{F}_{\mathcal{E},\xi}(\xi')(Y)=\begin{cases}[\![(\bar{x})\varphi]\!]\xi[《\mathcal{E}'》\xi[\xi'\lceil X]], & Y\equiv X\\ 《\mathcal{E}'》\xi[\xi'\lceil X](Y), & \text{否则}\end{cases}.$$

由$\boldsymbol{F}_{\mathcal{E},\xi}$的定义容易验证$\boldsymbol{F}_{\mathcal{E},\xi}$是$\mathcal{X}\to\Omega$上的单调函数,Tarski 定理保证了其存在最大不动点和最小不动点,分别记为$\nu\boldsymbol{F}_{\mathcal{E},\xi}$和$\mu\boldsymbol{F}_{\mathcal{E},\xi}$,则定义《$\mathcal{E}$》$\xi$为:若$\sigma=\nu$,则《$\mathcal{E}$》$\xi\overset{\text{def}}{=}\nu\boldsymbol{F}_{\mathcal{E},\xi}$;否则,若$\sigma=\mu$,则《$\mathcal{E}$》$\xi\overset{\text{def}}{=}\mu\boldsymbol{F}_{\mathcal{E},\xi}$.

定理 1. $\mathcal{E}$是嵌套谓词等式系,ξ是环境,则

$$《\mathcal{E}》\xi=【\mathcal{E}】\xi.$$

定理 1 说明对嵌套谓词等式系$\mathcal{E}$,语义函数《$\mathcal{E}$》ξ和【$\mathcal{E}$】ξ一致.称$\mathcal{E}$中具有相同不动点类型的连续的等式构成的子等式系为$\mathcal{E}$的块(block).$\mathcal{E}$可表示为块的序列,即$\mathcal{E}=\langle B_1,B_2,...,B_n\rangle$(或$\mathcal{E}=B_1::\mathcal{E}',\mathcal{E}'=\langle B_2,...,B_n\rangle$).其中,$B_i$是$\mathcal{E}$的块,且对任意$i\neq j$,$lhs(B_i)\cap lhs(B_j)=\varnothing$.$B_i$中等式的不动点类型记为$\sigma(B_i)$,且对任意$0<i<n,\sigma(B_i)\neq\sigma(B_j)$.在$\mathcal{E}$块表示的基础上,由 Bekic 定理[3]可将函数$\boldsymbol{F}_{\mathcal{E},\xi}$和《$\mathcal{E}$》$\xi$推广到块划分上,即

(1) 当$\mathcal{E}=\varepsilon$时,《$\mathcal{E}$》$\xi=\xi$.

(2) 当$\mathcal{E}=B_1::\mathcal{E}'$时,设$B_1\equiv\langle X_1(\bar{x}_1)=_\sigma\varphi_1,...,X_m(\bar{x}_m)=_\sigma\varphi_m\rangle$,变量集$\mathcal{X}'=\{X_1,...,X_m\}$构造函数$\boldsymbol{F}_{\mathcal{E},\xi}:(\mathcal{X}\to\Omega)\to(\mathcal{X}\to\Omega)$为

$$\boldsymbol{F}_{\mathcal{E},\xi}(\xi')(Y)=\begin{cases}[\![(\bar{x})\varphi_i]\!]\xi[《\mathcal{E}'》\xi[\xi'\lceil X']], & Y\equiv X_i\\ 《\mathcal{E}'》\xi[\xi'\lceil X'](Y), & \text{否则}\end{cases}.$$

定义《$\mathcal{E}$》ξ为:若$\sigma=\nu$,则《$\mathcal{E}$》$\xi\overset{\text{def}}{=}\nu\boldsymbol{F}_{\mathcal{E},\xi}$;否则,若$\sigma=\mu$,则《$\mathcal{E}$》$\xi\overset{\text{def}}{=}\mu\boldsymbol{F}_{\mathcal{E},\xi}$.也就是说,$\mathcal{E}$的语义可以在等式系块表示的基础上"逐块"地进行求解.通常地,在模型检测过程中,等式系的块划分越少,检测效率越高.在对$\mathcal{E}$中等式的顺序不作调整的前提下,块表示是$\mathcal{E}$的极大划分,因此,按这样的方式进行语义求解具有更高的效率.

4 谓词μ演算公式和模态图的对应关系

本节给出谓词μ演算公式到嵌套谓词等式系的转换,并建立嵌套谓词等式系和模态图的对应关系,从而得

到谓词μ演算公式和模态图的语义一致性.

4.1 谓词μ演算公式到嵌套谓词等式系的转换算法

给定命题 φ,可以通过算法 trans 转换为嵌套谓词等式系.其中,$\oplus$ 表示二目算子 $\oplus=\{\wedge,\vee\}$;$\ominus$表示一目算子 $\ominus\in Modop\bigcup\{\forall x,\exists x\}$.

命题转换算法: $trans:\varphi\to\mathcal{E}$

Procedure proc (φ,σ,i,k) (在下面的计算中记 $fdv(\varphi)=\{\bar{x}_i\}$)

Begin

case φ of

p $\Rightarrow$ return $\langle X_i(\bar{x}_i),\{X_i(\bar{x}_i),=_\sigma,p,k\},i+1\rangle$

$\varphi_1\oplus\varphi_2$ $\Rightarrow$ $(Y(\bar{y}),\mathcal{E}_1,i_1):=proc(\varphi_1,\sigma,i+1,k)$

$(Z(\bar{z}),\mathcal{E}_2,i_2):=proc(\varphi_2,\sigma,i_1,k)$

return $\langle X_i(\bar{x}_i),\{X_i(\bar{x}_i),=_\sigma,Y\bar{y}\oplus Z\bar{z},k\}::\mathcal{E}_1::\mathcal{E}_2,i_2\rangle$

$\ominus\varphi'$ $\Rightarrow$ $(Y(\bar{y}),\mathcal{E}_1,i_1):=proc(\varphi',\sigma,i+1,k)$

return $\langle X_i(\bar{x}_i),\{X_i(\bar{x}_i),=_\sigma,\ominus Y\bar{y},k\}::\mathcal{E}_1,i_1\rangle$

$(\bar{x}')\varphi'\bar{e}$ $\Rightarrow$ $(Y(\bar{y}),\mathcal{E}_1,i_1):=proc(\varphi',\sigma,i+1,k)$

return $\langle X_i(\bar{x}_i),\{X_i(\bar{x}_i),=_\sigma,[(\bar{y})Y\bar{y}]\bar{e},k\}::\mathcal{E}_1,i_1\rangle$

$Y\bar{e}$ $\Rightarrow$ 若 Y 是某个 $X_j(y<i)$ 且 X_j 的自由变量集为 $\bar{x}_j$

return $\langle X_i(\bar{x}_i),\{X_i(\bar{x}_i),=_\sigma,[(\bar{x}_j)X\bar{x}_j]\bar{e},k\},i+1\rangle$

否则,return $\langle X_i(\bar{x}_i),\{X_i(\bar{x}_i),=_\sigma,Y\bar{e},k\},i+1\rangle$

$(\sigma X.(\bar{x})\varphi')\bar{e}$ $\Rightarrow$ 若 $(\sigma'\neq\sigma$ 则 $\{\sigma:=\sigma';k:=k+1;X_{i+1}.d:=k;\}$

$X_{i+1}.\sigma:=\sigma';Dep(X_{i+1}):=\bigcup_{Y\in fpv(\varphi'[X/X_{i+1}])-X_{i+1}}Dep(Y);$

$l:=\max\{Y.d\mid Y.\sigma\neq X.\sigma\wedge Y\in Dep(X)\wedge X\in Dep(Y)\}$

return $\langle X_i(\bar{x}_i),\{X_i(\bar{x}_i),=_\sigma,[(\bar{y})X_{i+1}\bar{y}]\bar{e},l+1\}::\{X_{i+1}(\bar{y}),=_\sigma,Y\bar{y},l+1\}::\mathcal{E}',i_1\rangle$

End

Procedure *trans*(φ)

Begin

对φ作α变换,使其中出现的受囿变量两两不同;

$(X(\bar{x}),E,j):=proc(\varphi,\upsilon,1,0);$

Return $(X(\bar{x}),\mathcal{E})$;

end

Trans 将命题 φ 转换为嵌套谓词等式系 $\mathcal{E}$,主要思想是用形如 $X\bar{x}$ 的式子来代表 φ 中的各个子命题,这一过程通过递归地调用 *proc* 来实现.*proc* 包含 3 个参数: φ 是要处理的命题; σ 当前的不动点类型;整数 i 用来指示将要引入的下一个谓词变量的下标; k 是当前的嵌套深度.*proc* 的返回值是一个三元组: $X(\bar{x})$ 对应处理的命题 φ; 等式系 $\mathcal{E}$ 规定 X 的语义; j 指示下一个可用的谓词变量的下标.通常地, X 是等式系 $\mathcal{E}$ 中的第 1 个谓词变量.在 *proc* 的计算中,为了使得到的等式系具有较优的块分化,对公式的每个谓词变量 X 引入两个属性: $X.\sigma$ 是 X 的不动点类型; $X.d$ 是 X 的嵌套深度.这样,通过求解 l 值就可以将等式的嵌套深度尽量提升从而得到较优的块分化结果.例如,对公式 $\mu X.\upsilon Y.[-]Y\wedge\mu Z.[-](X\vee Z)$,直观上看,化为等式系后是 $\mu/\upsilon/\mu$ 三层嵌套形式.但是,注意到 $Y\notin Dep(X)$,本质上可以将该公式化为 μ/υ 两层嵌套的形式.

4.2 转换算法的正确性证明

定理 2. 设 φ 是命题; ξ 是环境; $i\geq 1$ 是一个整数,满足对任意 $k\geq i$, X_k 都不在 φ 中自由出现.若 $Proc(X(\bar{x}),\mathcal{E},j)=(X(\bar{x}),\mathcal{E},j)$,则 $[\![(\bar{x})\varphi]\!]\ \xi=《\mathcal{E}》\ \xi\ (X)$.

由定理 2 和定理 1 有如下推论:

推论 2. 设 φ 是谓词闭的命题,$(X(\bar{x}),\mathcal{E})=trans(\varphi)$;$\xi$ 是任意环境,且 $\{X_i \mid i\geq 1\}$ 不在 φ 中自由出现,则

$$[\![(\bar{x})\varphi]\!]\ \xi=【\mathcal{E}】\ \xi\ (X)=《\mathcal{E}》\ \xi\ (X).$$

特别地,若 φ 是谓词闭的命题,则转换得到的等式系 $\mathcal{E}$ 是谓词闭的,其语义不依赖于环境 ζ.最后,按等式的嵌套深度可以对 $\mathcal{E}$ 作块划分,即将嵌套深度为 i 的等式集合到块 B_i 中,引理 5 和引理 6 保证这样的顺序调整不影响 $\mathcal{E}$ 的语义,最终可以将 $\mathcal{E}$ 表示为 $\langle B_0,B_1,...,B_n\rangle$ 的形式.

定理 3. 设 φ 是谓词闭的命题,$(X(\bar{x}),E)=trans(\varphi)$,则按等式的嵌套深度对等式系作块划分后得到的等式系保持原语义.

4.3 嵌套谓词等式系的模态图表示

设 φ 是谓词闭的命题,$(X_0,\mathcal{E})$是 $trans(\varphi)$ 的返回值.$\mathcal{E}$ 按嵌套深度表示为块序列 $\langle B_0,B_1,...,B_n\rangle$,则按下列规则可以构造模态图 $\mathcal{M}=(N,E,L_N,T,r)$:

- 设 $X(\bar{x})=b$ 是块 B_i 中的等式,则在 $\mathcal{M}$ 中存在结点 v 与 X 相对应,$L_V(v)=\{\bar{x}\};L_O(v)=b;L_D(v)=i$.
- 设 $X(\bar{x})=\ominus Y\bar{y}$ 是块 B_i 中的等式,则在 $\mathcal{M}$ 中存在结点 v 与 X 相对应,$L_V(v)=\{\bar{x}\};L_O(v)=\ominus;L_D(v)=i$ 且若 Y 对应结点 u,则 $(v,u)\in\mathcal{E}$.
- 设 $X(\bar{x})=[(\bar{y})Y\bar{y}](\bar{e})$ 是块 B_i 中的等式,则在 $\mathcal{M}$ 中存在结点 v 与 X 相对应,$L_V(v)=\{\bar{x}\};L_O(v)=\bar{y}:=\bar{e}$;$L_D(v)=i$;且若 Y 对应结点 u,则 $(v,u)\in\mathcal{E}$.
- 设 $X(\bar{x})=Y_1(\bar{y}_1)\oplus Y_2(\bar{y}_2)$ 是块 B_i 中的等式,则在 $\mathcal{M}$ 中存在结点 v 与 X 相对应,$L_V(v)=\{\bar{x}\};L_O(v)=\oplus$;$L_D(v)=i$;且若 Y_1 对应结点 u_1,Y_2 对应结点 u_2,则 $(v,u_1)\in\mathcal{E}$ 且 $(v,u_2)\in\mathcal{E}$.
- 构造映射 T:对 $0\leq i\leq n, T(i)=\sigma(B_i)$.
- $\mathcal{M}$ 的根结点 r 是变量 X_0 对应的结点.

注意到,按上述规则得到的模态图 $\mathcal{G}$ 是良构的.

定理 4. 设 $\mathcal{M}$ 是嵌套谓词等式系 $\mathcal{E}$ 对应的模态图.X 是 $\mathcal{E}$ 中出现的谓词变量的集合.构造完备格 $X\to\Omega$ 到 $N\to\Theta$ 的一一映射 I 为

$$若\ \zeta=I(\xi),n\ 和\ X\ 对应,则对任意\ \bar{v}\ 和计值\ \rho\equiv L_V(n)\to\bar{v},使得\ \zeta n\rho=\xi X\bar{v}.$$

这样,对任意环境 ξ 和 ζ,若 $\zeta=I(\xi)$,$\zeta'=《\mathcal{E}》\ \xi$,$\zeta'=[\![\mathcal{M}]\!]\ \zeta$,则 $\zeta'=I(\zeta')$.

设 φ 是谓词闭的命题,$\mathcal{M}$ 是 φ 的模态图.由推论 2、定理 4 及公式的语义性质即可得到如下推论,它说明 φ 及其模态图 $\mathcal{E}$ 的语义是一致的.

推论 3. 设 φ 是谓词变量闭的命题,φ 表示为模态图 $\mathcal{M},r$ 是 $\mathcal{M}$ 的根结点,则对任意环境 ξ,ζ 和任意计值 ρ,

$$[\![\varphi]\!]\ \rho\xi=[\![\mathcal{M}]\!]\ \zeta r\rho.$$

5 结论和相关工作

对命题μ演算,文献[3~5]讨论了μ演算公式和嵌套不动点等式系之间的关系,特别是在文献[3,4]中给出了公式到嵌套不动点等式系的转换方法.在等式系表示的基础上,人们提出了一些效率较好的模型检测算法[4~9]等.本文的工作可以看作是上述工作在一阶情况下的扩展.如引言中所述,文献[1]提出了基于模态图的传值进程模型检测算法.该算法具有较好的时空效率.因此,讨论谓词μ演算公式、嵌套谓词等式系和模态图之间的语义关系及转换过程是很有必要的.特别是在一阶情况下,由于引入了数据,使得公式更为复杂、难懂,有必要将公式转换为嵌套谓词等式系或模态图.此外,通过对嵌套谓词等式系的性质加以发掘利用,算法 trans 在对部分公式作处理时所生成的谓词等式系具有较优的块划分.

References:

[1] Lin HM. Model checking value-passing processes. In: Proceedings of the 8th Asia-Pacific Software Engineering Conference. Macao: IEEE Press, 2001. 3~10.

[2] Lin HM. Symbolic transition graphs with assignment. In: Montanari U, Sassone V, eds. Proceedings of the CONCUR'96. LNCS 1119, Heidelberg: Springer-Verlag, 1996. 50~65.

[3] Mader, A. Verification of modal properties using Boolean equation systems [Ph.D. Thesis]. TMU, 1997.

[4] Cleaveland R, Klein M, Steffen B. Faster model checking for the modal Mu-calculus. In: Bochmann GV, Probst DK, eds. Proceedings of the CAV'92. LNCS 663, Heidelberg: Springer-Verlag, 1992. 410~422.

[5] Andersen HR. Model checking and Boolean graphs. Theoretical Computer Science, 1994,126(1):3~30.

[6] Andersen HR. Verification of temporal properties of concurrent systems [Ph.D. Thesis]. Aarhus: Aarhus University, 1993.

[7] Liu X, Ramakrishnan CR, Smolka SA. Fully local and efficient evaluation of alternating fixed points. In: Steffen B, ed. Proceedings of the TACAS'98. LNCS 1384, Heidelberg: Springer-Verlag, 1998. 5~19.

[8] Bhat GS, Cleaveland R. Efficient model checking via the equational μ-calculus. In: Clarke EM, ed. Proceedings of the 11th Annual Symposium on Logic in Computer Science. New Jersey: IEEE Computer Society Press, 1996. 304~312.

[9] Vergauwen B, Lewi J. Efficient local correctness checking for single and alternating Boolean equation systems. In: Abiteboul S, Shamir E, ed. Proceeding of the ICALP'94. LNCS 820, Heidelberg: Springer-Verlag, 1994. 304~315.

Call for Papers for GCC 2003

December 7~10 , 2003, Shanghai, China

http://www.cs.sjtu.edu.cn/gcc2003/index.htm

The Second International Workshop on Grid and Cooperative Computing (GCC2003) is to be held from December 7~10, 2003 in Shanghai, China. GCC2003 is the follow-up of the highly successful GCC2002 Workshop held in SanYa, HaiNan. It will serve as a forum to present current and future work as well as to exchange research ideas by researchers, developers, practitioners, and users in Grid computing, Web services and cooperative computing.

TOPICS OF INTEREST

The main topics of interest include, but not limited to:

Grid Computing and Grid Security

Grid Information Services

Grid Middleware and Toolkits

Grid Monitoring, Management and Organization Tools

Grid Applications

Information Grid and Knowledge Grid

Advance Resource Reservation and Scheduling

Performance Evaluation and Modeling

Web Services and Web Security

P2P Computing

Cooperative Middleware

Software Integration Technologies

Software Engineering Support for Cooperative Computing

Computer-Supported Cooperative Work

SUBMISSION

GCC 2003 invites authors to submit original and unpublished work. Papers should not exceed 10 pages of text using 10 point size type on 8.5×11 inch paper. Papers will be refereed and accepted on the basis of their scientific merit and relevance to the conference topics. Submission of full technical papers is called. Papers should be written in English and submitted in PDF format. All papers are to be electronically submitted to (Please also include the address, telephone, FAX, and email of the primary contact person in a separate sheet): gcc2003@cs.sjtu.edu.cn

IMPORTANT DATES

Electronic submissions due:	September 15, 2003
Acceptance notification due:	October 15, 2003
Camera-ready due:	November 5, 2003
GCC2003 Workshop:	December 7~10, 2003