第6卷第2期 软 件 学 报 Vol. 6, No. 2
1995年2月 JOURNAL OF SOFTWARE Feb. 1995

通信协议转换器及其构造*

赵锦蓉

(清华大学计算中心，北京 100084)

摘要 通信协议转换器可以看作是两个协议的信息之间的映射. 这个映射可以用一个重要信息对偶集合 K 来刻划. 本文给出相对于 K 的协议转换器的形式定义. 然后，我们提出用两个有限状态自动机的对偶积来构造协议转换器的方法，并论证了方法的正确性.

关键词 协议转换器，重要信息映射，对偶积，同态.

20多年来，计算机网络技术迅猛发展，建立了大量的多种多样的网络系统，这就带来了一个各种网络之间如何互连的问题. 一个办法是推行国际标准. OSI 网络体系结构及通信协议的国际标准已越来越成熟，但是，要把大量已存在的非 OSI 体系的网络都改造成 OSI 体系，都采用标准协议，那是很困难的. 而且网络技术在不断发展，在进行标准化的同时随时产生多样化，因此考虑异构网络的互连通信大概永远不可避免. 协议转换就是为了解决这个问题而提出来的.

什么是协议转换？Green 在文献[1]中阐明了协议转换的基本概念. 他认为协议转换是一种映射，就是把某一协议的收发信息(或事件)序列映射为另一协议的收发信息序列. 这个一般性的观点已被广泛接受. 但是不可能把一个协议的每一个信息都映射成另一协议的信息，这也不必要. 我们称需要映射的信息为重要信息. 因此协议转换可以看作是两个协议的重要信息之间的映射. 所谓重要信息和非重要信息是相对而言的，要根据具体需要加以确定. 选择不同的重要信息作映射，当然会得到不同的转换器.

怎样建立协议转换器的形式模型？怎样加以描述？当然这首先依赖于通信协议的模型. 关于通信协议，一个使用最广泛的模型是有限状态自动机(fsm)[2]，就是说用一组有限状态自动机来描述协议，这种有限状态自动机称为通信有限状态自动机(cfsm). 例如用$[A,B]$表示协议，其中 A,B 是 cfsm. 设给定两个不同的协议$[A,B]$和$[G,H]$，其中 A,G 是 Sender，B,H 是 Receiver. 那么为了使 Sender A 和 Receiver H 之间通信就需要转换器，协议转换就是在$[A,B]$和$[G,H]$之间进行. Okumura[3]用一个 cfsm C 来表示协议转换器，即协议转换过程用协议$[A,C,H]$来表示，其中 A 和 C 之间遵循协议$[A,B]$，C 和 H 之间遵循协议$[G,H]$. 为了构造 C，在文献[3]中引进所谓转换种子. 转换种子的好坏直接影响到能否用它构造出正确的协议转换器. 然而因为转换种子的概念比较含混，因此构造一个好的转换种子是

* 本文 1991－11－18 收到，1993－06－02 定稿

作者赵锦蓉，女，1941 年生，教授，主要研究领域为通信协议形式描述技术，计算机网络，计算机软件.

本文通讯联系人：赵锦蓉，北京 100084，清华大学计算中心

十分困难的事. 后来 Shu 和 Liu[4]提出了另一种协议转换的模型. 他们构造两个 cfsm B'和 G',而把协议转换用协议$[A,B',G',H]$来表示. B'和 G'是把 B 和 G 扩展后得到的,主要是系统地插入一系列 PUT 和 GET. 他们认为这样做比构造一个通信有限状态自动机 C 作为转换器容易且空间复杂性低. 当然 B'及 G'都比 C 简单,但是这里用两个 cfsm B'和 G'代替一个 cfsm C 是值得斟酌的. 因为在他们的方法中要考虑$[A,B',G',H]$的全局状态图,4 个 cfsm 的全局状态图比 3 个 cfsm 的全局状态图要复杂得多,即使每个 cfsm 要简单一些. 所以,文献[4]中所说的空间复杂性低是要进一步讨论的.

Calvert 和 Lam[5]提出了从服务描述及协议描述用商算法得到转换器描述. 但正如文中所说的,计算是十分困难的.

在本文中,我们采用$[A,C,H]$作为协议转换的模型. 给定两个协议$[A,B]$,$[G,H]$和重要信息对偶集合 K,我们的转换器是 B 和 G 关于 K 的一种对偶积. 既不用转换种子,也无需插进 PUT 和 GET 来对 B 和 G 加以扩展. 在本文中,还将说明可以构造一个 C'使 Shu 和 Liu 的协议转换$[A,B',G',H]$和$[A,C',H]$的全局状态图等价. 因此,在某种意义上 Shu 和 Liu 的协议转换即$[A,C',H]$,然而,这样构造出的 C'比我们构造的$[A,C,H]$中的 C 要复杂得多. 所以,从全局状态图的角度看,我们构造的$[A,C,H]$比他们的$[A,B',G',H]$要简单一些.

对偶积的概念在文献[6]中曾用来从两个交换信息的子模块构造合成系统.

本文的内容如下:第 1 节给出协议转换器的概念,第 2 节给出协议转换器的构造方法并证明其正确性,第 3 节给出协议转换器的一些例子,第 4 节是结论.

1 协议转换的概念

一个通信协议通常用几个通信有限状态自动机(cfsm)来描述[2],而信息则通过先进先出队列进行交换.

一个通信有限状态自动机 A 是四元组$\langle S,\Sigma,o,\delta\rangle$,其中:$S$ 是 A 的有限状态集合;Σ 是 A 的收发信息集合,$\Sigma=\Sigma^-\cup\Sigma^+$,其中 Σ^- 是发送信息(用"$-m$"表示)集合,Σ^+是接收信息(用"$+m$"表示)集合;o 是 A 的初始状态;δ 是转移函数,即从 $S\times\Sigma$ 到 S 的部分函数.

一个协议是$[A_1,A_2,\cdots,A_n]$,其中 A_i 是 cfsm $\langle S_i,\Sigma_i,o_i,\delta_i\rangle$,$\Sigma_i=\bigcup_{j=1}^{n}\Sigma_{ij}$,$\Sigma_{ij}=\Sigma^+{}_{ij}\cup\Sigma_{ij}^-$,而且 $\Sigma_{ij}^-=\overline{\Sigma_{ji}^+}$. 这里,$\Sigma_{ij}^-$是从 A_i 到 A_j 的发送信息集合,Σ_{ji}^+是 A_j 从 A_i 的接收信息集合,它们也可能为空. 注意,我们用 $\overline{A}=\overline{\{a_1,a_2,\cdots,a_n\}}$来表示$\{-a_1,-a_2,\cdots,-a_n\}$且$-(-m)=+m$. 因此 $\bar{\bar{A}}=A$.

现在来说明协议$[A_1,A_2,\cdots,A_n]$的全局状态及其转移→:

(1)$o=\langle(o_1,o_2,\cdots,o_n),(\lambda,\cdots,\lambda)\rangle$是全局状态,称为初始全局状态,其中 λ 表示空队列.

(2)若 $s=\langle(s_1,\cdots,s_i,\cdots,s_n),(q_{11},\cdots,q_{ij},\cdots,q_{nn})\rangle$是全局状态,且 $\delta_i(s_i,-m)=t_i$,其中 $-m\in\Sigma_{ij}^-$,则 $s'=\langle(s_1,\cdots,t_i,\cdots,s_n),(q_{11},\cdots,q_{ij}m,\cdots,q_{nn})\rangle$是全局状态且有转移 $s\to s'$,与转移有关的收发信息是 $\mu=(\varepsilon,\cdots,\varepsilon,-m,\varepsilon,\cdots,\varepsilon)$,它是 n 个元素的向量,其中只有第 i 个元素不是 ε,ε 表示无收发. 因此转移及有关的收发信息可记为 $s\xrightarrow{\mu}s'$.

(3)若 $s=\langle(s_1,\cdots,s_i,\cdots,s_n),(q_{11},\cdots,q_{ji},\cdots,q_{mn})\rangle$是全局状态，且 $\delta_i(s_i,+m)=t_i$和 $q_{ji}=mq'$，其中$+m\in\Sigma_{ij}^+$，则 $s'=\langle(s_1,\cdots,t_i,\cdots,s_n),(q_{11},\cdots,q',\cdots,q_{mn})\rangle$是全局状态且有转移 $s\to s'$，有关的收发信息是 $\mu=(\varepsilon,\cdots,\varepsilon,+m,\varepsilon,\cdots,\varepsilon)$，它的第 i 个元素是$+m$. 可以记为 $s\xrightarrow{\mu}s'$.

因此全局状态及其转移构成一个转移系统，结点表示全局状态，根是初始全局状态，从全局状态 s 到全局状态 s'的转移用从 s 到 s'的边表示，有关的收发信息作为边的标号.

在协议$[A_1,A_2,\cdots,A_n]$ 的全局状态图中，从初始全局状态出发的任一路径上的收发信息序列称为协议的行迹. 行迹 π 可以用 $\pi^1\pi^2\cdots$表示，其中 π^i 是与一个转移有关的收发信息，即 $\pi^i=(\pi_1^i,\pi_2^i,\cdots,\pi_n^i)$，$\pi_j^i$ 是 A_j 的收发信息. 定义向量的连接运算如下：$(a_1,a_2,\cdots,a_n)(b_1,b_2,\cdots,b_n)=(a_1b_1,a_2b_2,\cdots,a_nb_n)$，则 π 可以写成$(\pi_1,\pi_2,\cdots,\pi_n)$，其中 $\pi_j=\pi_j^1\pi_j^2\cdots$是 A_j 的收发信息序列，称为行迹 π 的 j 分量. 以上是一些基本概念，下面我们定义行迹段.

定义 1.1. 给定协议$[A_1,A_2,\cdots,A_n]$，在其全局状态图中从初始全局状态 o 到全局状态 s 的任一路径段上的收发信息序列称为协议对 s 的行迹段，协议对 s 的行迹段的全体记为 $trace_s([A_1,A_2,\cdots,A_n])=\{\pi\mid\pi$ 是协议$[A_1,A_2,\cdots,A_n]$对 s 的行迹段$\}$，它的 j 分量记为 $trace_s(A_j,[A_1,A_2,\cdots,A_n])=\{\pi_j\mid\pi_j$ 是协议对 s 的行迹段 π 中 A_j 的收发信息序列$\}$.

协议对初始全局状态的行迹段的全体(即全局状态图中从初始全局状态返回初始全局状态的所有路径上的标号序列)记为 $trace([A_1,A_2,\cdots,A_n])$，这些行迹段的 j 分量的全体记为 $trace(A_j,[A_1,A_2,\cdots,A_n])$.

作为例子，我们考虑图 1 的协议. 图 2 是图 1 中协议$[A,B]$的全局状态图，在图中只标出边的标号，o 是根. 由有限状态自动机理论不难得到：

$$trace([A,B])=\{(((-m_e,\varepsilon)(\varepsilon,+m_e)(\varepsilon,-n)(+n,\varepsilon))^i(-m,\varepsilon)(\varepsilon,+m)(\varepsilon,-a),(+a,\varepsilon))^j\mid i,j=0,1,2,\cdots\},$$

$$trace(A,[A,B])=\{(((-m_e)(+n))^i(-m)(+a))^j\mid i,j=0,1,2,\cdots\},$$

$$trace(B,[A,B])=\{(((+m_e)(-n))^i(+m)(-a))^j\mid i,j=0,1,2,\cdots\}.$$

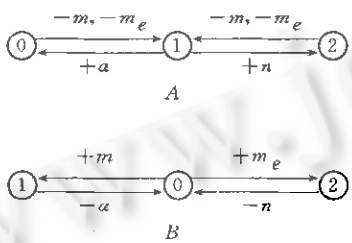


图1　协议$[A,B]$

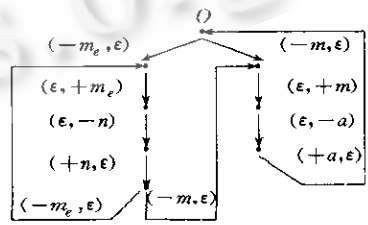


图2　$[A,B]$的全局状态图

所以协议$[A,B]$可以看作映射 α：

$$(((-m_e)(+n))^i(-m)(+a))^j\to(((+m_e)(-n))^i(+m)(-a))^j.$$

我们知道通信有限状态自动机 A 和 B 作为有限状态自动机分别定义了语言 L_A 和 L_B，它们可以由正则表达式$(((-m)+(-m_e))((+n)((-m)+(-m_e)))^*(+a))^*$和$((+m)(-a)+(+m_e)(-n))^*$表示，所以 $\alpha\subset L_A\times L_B$. 也就是说 α 是 L_A 到 L_B 的部分映射.

图 3 是另一协议的例子，我们有

$$trace(G,[G,H])=\{((+p)(-d)^k(-e))^l\mid k,l=0,1,2,\cdots\},$$

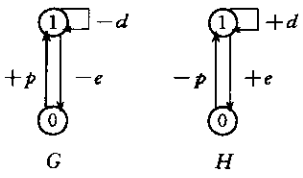


图3 协议[G,H]

$trace(H,[G,H])=\{((-p)(+d)^k(+e))^l|k,l=0,1,2,\cdots\}$. 则协议[G,H]给出映射 β:

$$((+p)(-d)^k(-e))^l\rightarrow((-p)(+d)^k(+e))^l$$

正如例子所示,任何协议可以考虑为两个语言之间的映射. 这样一来[A,B]及[G,H]的协议转换是从 L_A 到 L_H 的映射 φ. 注意 L_A 和 L_B 是在同一字母集上的语言,L_G 和 L_H 也是,但 L_A 和 L_H 却不是. 所以 φ 是在不同字母集上两个语言之间的映射,由此可见协议转换和语言翻译之间有某种类似.

在语言翻译中,重要词(关键词)的翻译起核心作用. 类似地,在协议转换中重要信息的映射是核心的. 例如,为了描述$(((-m_e)(+n))^i(-m)(+a))^j$和$((-p)(+d)^j(+e))$之间的映射,我们只要选$-m$和$+d$为重要信息,要求$-m$被映射为$+d$就行. 我们称这种$-m$和$+d$为重要信息对偶,或简称对偶. 下面仔细分析一下 φ.

一般说来,对给定的协议[A,B]和[G,H],假设 A,B,G,H 的收发信息集合分别为 Σ_A,Σ_B,Σ_G,Σ_H. 给定一个重要信息对偶集合 K

$$K=\{\langle a_i,h_i\rangle|a_i\in\Sigma_A,h_i\in\Sigma_H,i=1,2,\cdots,k\},$$

令 $$M=\{a_1,a_2,\cdots,a_k\}\subseteq\Sigma_A,\quad D=\{h_1,h_2,\cdots,h_k\}\subseteq\Sigma_H.$$

K 确定了从集合 M 到集合 D 的一个映射 φ,即 $\varphi(a_i)=h_i$. 用 M^* 表示由 M 产生的自由半群(含单位元),即 M 上有限字的集合以连接为运算. 类似地,D^* 是由 D 产生的自由半群. 则 φ 可以扩展为从 M^* 到 D^* 的一个同态,即从集合 M^* 到 D^* 的保持连接运算的映射. 下面我们将用同样的 φ 既表示映射 $M\rightarrow D$,又表示同态 $M^*\rightarrow D^*$. 对任意 $\alpha\in\Sigma_A^*$,设 $\tilde{\alpha}$ 表示从 α 删去所有$(\Sigma_A\backslash M)$中的字母后得到的,则 $\tilde{\alpha}\in M^*$ 且 $\varphi(\tilde{\alpha})\in D^*$.

应注意到,对偶集合 K 也确定了从 $\overline{M}=\{b_1,\cdots,b_k\}$ 到 $\overline{D}=\{g_1,\cdots,g_k\}$ 的一个映射 ψ 如下:若 $\varphi(a_i)=h_i$ 而 $b_i=-a_i$(即当 $a_i=+m$ 时 $b_i=-m$,当 $a_i=-m$ 时 $b_i=+m$),$g_i=-h_i$,则 $\psi(b_i)=g_i$. 同样,ψ 可以扩展为 $\overline{M}^*$ 和 $\overline{D}^*$ 之间的同态.

这两个有关的映射 φ 和 ψ 叫做与 K 关联的同态.

定义 1.2. 给定协议[A,B],[G,H]和集合 $K=\{\langle a_i,h_i\rangle|a_i\in\Sigma_A,h_i\in\Sigma_H,i=1,2,\cdots,K\}$,设 φ 表示与 K 关联的同态,且$\{(\alpha,\delta)|\alpha\in trace(A,[A,B]),\delta\in trace(H,[G,H]),$而 $\varphi(\tilde{\alpha})=\delta\}\neq\varnothing$,$\varnothing$表示空集. 若存在 cfsm C 满足下列条件:

$(\exists\ \omega)(\alpha,\omega,\delta)\in trace([A,C,H])$充要条件是

$\alpha\in trace(A,[A,B])$,$\delta\in trcae(H,[G,H])$且 $\varphi(\tilde{\alpha})=\delta$.

则 C 称为[A,B]和[G,H]关于 K 的协议转换器.

2 协议转换器的构造

首先,我们给出两个 cfsm 的对偶积运算.

给定两个 cfsm $P_1=\langle S_1,\Sigma_1,o_1,\delta_1\rangle$ 和 $P_2=\langle S_2,\Sigma_2,o_2,\delta_2\rangle$,其中 $\Sigma_1\cap\Sigma_2=\varnothing$. 令 $K=\{\langle m_i,d_i\rangle|m_i\in\Sigma_1,d_i\in\Sigma_2,i=1,2,\cdots,k\}$是对偶集合,$M=\{m_1,m_2,\cdots,m_k\}$,$D=\{d_1,d_2,\cdots,d_k\}$. cfsm $C=\langle S,\Sigma,o,\delta\rangle$定义如下:

(1)$S=S_1\times S_2$

(2)$o=\langle o_1,o_2\rangle$

(3)$\Sigma=\{\Sigma_1\backslash M\}\cup\{\Sigma_2\backslash D\}\cup\{\langle m_i\&d_i\rangle\mid\langle m_i,d_i\rangle\in K\}$

(4)$\delta(\langle x,y\rangle,a)=\langle x',y'\rangle$ 表示

$$(((a\in\Sigma_1\backslash M)\wedge\delta_1(x,a)=x'\wedge y'=y)\vee((a\in\Sigma_2\backslash D)\wedge\delta_2(y,a)=y'\wedge x'=x)\vee(a=\langle m_i\&d_i\rangle\wedge\delta_1(x,m_i)=x'\wedge\delta_2(y,d_i)=y')).$$

C 称为 P_1 和 P_2 关于 K 的对偶积,可记为 $P_1\otimes^K P_2$. 若 K 空,我们记 C 为 $P_1\cdot P_2$,称作积.

图 1 中的 B 和图 3 中的 G 关于 $K=\{\langle+m,-d\rangle\}$ 的对偶积见图 4.

任意给定 $K=\{\langle b_i,g_i\rangle\mid b_i\in\Sigma_B,g_i\in\Sigma_G,\ i=1,2,\cdots,k\}$,$B\otimes^K G$ 当然不一定是协议$[A,B]$和$[G,H]$的转换器. 为了搞清楚对偶积和协议转换器的关系,首先我们分析一下 $B\cdot G$ 和 $B\otimes^K G$. 设对于 $(\alpha,\omega,\delta)\in trace_s([A,B\otimes^K G,H])$,令 ω_B 是从 ω 删去 Σ_G 的字母且用 b_i 代替 $b_i\&g_i$ 得到的. 类似地,ω_G 是从 ω 删去 Σ_B 的字母且用 g_i 代替 $b_i\&g_i$ 得到的.

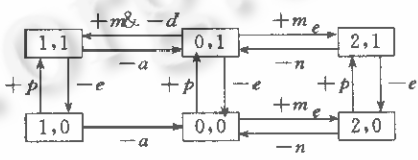


图4　$B\otimes^K G$

从 $B\otimes^K G$ 及 $B\cdot G$ 的定义容易证明下面两个引理. 我们用 x,y,u,v 表示 A,B,G,H 的状态.

引理 2.1. 设 $s=\langle(x,\langle y,u\rangle,v),(q_{12},q_{21},q_{23},q_{32})\rangle$是$[A,B\otimes^K G,H]$的全局状态且$(\alpha,\omega,\delta)\in trace_s([A,B\otimes^K G,H])$,则 $r=\langle(x,y),(q_{12},q_{21})\rangle$和 $t=\langle(u,v),(q_{23},q_{32})\rangle$分别为$[A,B]$和$[G,H]$的全局状态,且$(\alpha,\omega_B)\in trace_r([A,B])$,$(\omega_G,\delta)\in trace_t([G,H])$.

引理 2.2. 给定协议$[A,B]$和$[G,H]$,则$(\alpha,\omega,\delta)\in trace([A,B\cdot G,H])$的充要条件是存在 β 和 γ 使得$(\alpha,\beta)\in trace([A,B])$,$(\gamma,\delta)\in trace([G,H])$,且 ω 是 β 和 γ 的一个任意穿插.

引理 2.3. 给定协议$[A,B]$和$[G,H]$,设 $M=\{b_1,\cdots,b_k\}\subseteq\Sigma_B$,且 $D=\{g_1,\cdots,g_k\}\subseteq\Sigma_G$. 若$(\alpha,\omega,\delta)\in trace([A,B\cdot G,H])$,且 $\psi(\tilde{\omega}_B)=\tilde{\omega}_G$,则存在 ω',$(\alpha,\omega',\delta)\in trace([A,B\otimes^K G,H])$,其中 $K=\{\langle b_i,g_i\rangle\mid b_i\in M,g_i\in D\}$.

证明:由引理 2.1,$(\alpha,\omega_B)\in trace([A,B])$且$(\omega_G,\delta)\in trace([G,H])$. 设 $\tilde{\omega}_B=b_{i_1}b_{i_2}\cdots b_{i_j}\in M^*$,因为 $\psi(\tilde{\omega}_B)=\tilde{\omega}_G$,所以 $\tilde{\omega}_G=g_{i_1}g_{i_2}\cdots g_{i_j}\in D^*$. 现作 ω_B 和 ω_G 的穿插 ω'' 使得 b_i 和 g_i 在 ω'' 相邻,$t=i_1,i_2,\cdots,i_j$. 由引理 2.2,$(\alpha,\omega'',\delta)\in trace([A,B\cdot G,H])$. 用 $b_i\&g_i$ 代替 b_ig_i(及 g_ib_i),从 ω'' 得到 ω'. 显然$(\alpha,\omega',\delta)\in trace([A,B\otimes^K G,H])$. 证毕.

定义 2.1. 给定协议$[A,B]$和$[G,H]$,设 $K=\{\langle a_i,h_i\rangle\mid a_i\in\Sigma_A,h_i\in\Sigma_H,i=1,2,\cdots,k\}$,$\varphi$ 和 ψ 是与 K 关联的同态. 若对任何$(\alpha,\beta)\in trace([A,B])$和$(\gamma,\delta)\in trace([G,H])\varphi(\tilde{\alpha})=\tilde{\delta}$的充要条件是 $\varphi(\tilde{\beta})=\tilde{\gamma}$,就说 K 是适当的.

定理 2.1. 给定协议$[A,B]$,$[G,H]$和集合 $K=\{\langle a_i,h_i\rangle\mid a_i\in\Sigma_A,h_i\in\Sigma_H,i=1,2,\cdots,k\}$,$\overline{K}=\{\langle b_i,g_i\rangle\mid b_i=-a_i\in\Sigma_B,g_i=-h_i\in\Sigma_G,i=1,2,\cdots,k\}$. 设 φ 表示与 K 关联的同态,且$\{(\alpha,\delta)\mid\alpha\in trace(A,[A,B]),\delta\in trace(H,[G,H])$而 $\varphi(\tilde{\alpha})=\tilde{\delta}\}\neq\varnothing$. 若 K 是适当的,则 $C=B\otimes^{\overline{K}}G$ 是$[A,B]$和$[G,H]$关于 K 的协议转换器.

证明:由定义1.2,我们需要证明$(\exists\ \omega)(\alpha,\omega,\delta)\in trace([A,C,H])$的充要条件是$\alpha\in trace(A,[A,B])$,$\delta\in trace(H,[G,H])$且$\varphi(\tilde{\alpha})=\tilde{\delta}$.

(1)必要性

若$(\alpha,\omega,\delta)\in trace([A,B\otimes^{K}G,H])$,根据引理2.1,我们有$(\alpha,\omega_B)\in trace([A,B])$且$(\omega_G,\delta)\in trace([G,H])$,即$\alpha\in trace(A,[A,B])$和$\delta\in trace(H,[G,H])$.由$B\otimes^{K}G$的定义,$\overline{M}=\{b_1,\cdots,b_k\}$和$\overline{D}=\{g_1,\cdots,g_k\}$中信息只能以成对形式$b_i\&g_i$出现,所以$\psi(\tilde{\omega}_B)=\tilde{\omega}_G$.而$K$为适当,则$\varphi(\tilde{\alpha})=\tilde{\delta}$.

(2)充分性

若$\alpha\in trace(A,[A,B])$,$\delta\in trace(H,[G,H])$且$\varphi(\tilde{\alpha})=\tilde{\delta}$,那么有$\beta$和$\gamma$,$(\alpha,\beta)\in trace([A,B])$,$(\gamma,\delta)\in trace([G,H])$.由引理2.2,若$\omega$是$\beta$和$\gamma$的一个任意穿插,则$(\alpha,\omega,\delta)\in trace([A,B\cdot G,H])$.因$\varphi(\tilde{\alpha})=\tilde{\delta}$,而$K$为适当,则$\psi(\tilde{\beta})=\tilde{\gamma}$.由引理2.3,存在$\omega'$,$(\alpha,\omega',\delta)\in trace([A,B\otimes^{K}G,H])$.

推论.给定协议$[A,B]$,$[G,H]$和集合$K=\{\langle a,h\rangle\}$,其中$a\in\Sigma_A$,$h\in\Sigma_H$,$\overline{K}=\{\langle b,g\rangle\}$,其中$b=-a\in\Sigma_B$,$g=-h\in\Sigma_G$.设$\varphi$表示与$K$关联的同态,且$\{(\alpha,\delta)\mid\alpha\in trace(A,[A,B])$,$\delta\in trace(H,[G,H])$而$\varphi(\tilde{\alpha})=\tilde{\delta}\}\neq\varnothing$.则$C=B\otimes^{K}G$是$[A,B]$和$[G,H]$关于$K$的协议转换器.

证明:因为K只包含一个对偶,则K为适当.由定理2.1推论成立.

因此,图4是$[A,B]$和$[G,H]$关于$K=\{\langle+m,-d\rangle\}$的转换器.

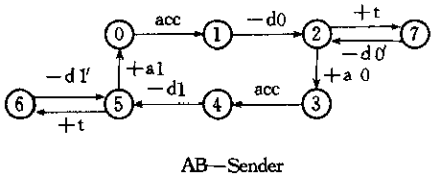
3 一些例子

图5和图6分别是简化的交替位协议和无序号协议.*acc*和*del*分别表示从用户接受数据和向用户递交数据(*ACC*和*DEL*也类似),它们都是和用户交换的信息或称服务原语,可以略去.$+t$和$+T$用来模拟超时,而$-t$和$-T$可分别看作$-a0$(或$-a1$)和$-A$的延迟.图中表示了超时重传的机制.为了简化讨论,没有考虑信息丢失,只是延迟引起超时重传.

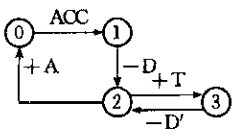
我们在协议中区分第一次收发的信息和重传信息.即$+d0$是第一次收到的$d0$信息,而$+d0'$是收到的重传信息,它们分别对应$-d0$和$-d0'$.$\pm d1$和$\pm d1'$,$\pm D$和$\pm D'$等的意义类似.很自然,我们可选择$\overline{K}=\{\langle+d0,-D\rangle,\langle+A,-a0\rangle,\langle+d1,-D\rangle,\langle+A,-a1\rangle\}$,即重传信息不作映射.这样做是合理的.我们删去AB—Receiver中的*del*和NS—Sender中的*ACC*得图7.*del*和*ACC*是和用户交换的信息,在协议转换器中不考虑.对图7中的AB—Receiver和NS—Sender作关于$\overline{K}$的对偶积见图8.容易验证对图5、图6的交替位协议和无序号协议,$K=\{\langle-d0,+D\rangle,\langle-A,+a0\rangle,\langle-d1,+D\rangle,\langle-A,+a1\rangle\}$是适当的,所以图8是这两个协议关于$K$的协议转换器.

若我们选择对偶集合$\overline{K'}=\{\langle+d0,-D\rangle,\langle+d1,-D\rangle\}$,AB—Receiver和NS—Sender关于$\overline{K'}$的对偶积也是交替位协议和无序号协议之间的一个转换器.这个转换器在收到$d0$并映射为D发送给NS—Receiver后,不必等待NS—Receiver的确认A即可发送$a0$给AB—Sender,所以这里确认$a0$和$a1$没有端到端的意义.而用图8中的转换器,确认有端到端的意义.

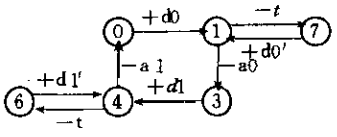
作为另一个例子,我们考虑协议*TCP*和*ISO/TP*之间的转换.若我们选择对偶集合



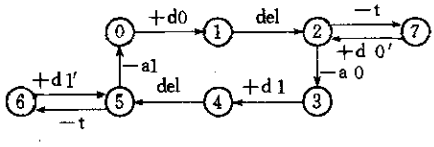
AB—Sender



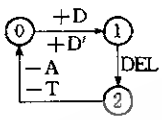
NS—Sender



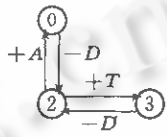
AB—Receiver



AB—Receiver



NS—Receiver



NS—Sender

图5　交替位协议

图6　无序号协议

图7　修改后的*AB—Receiver*和*NS—Sender*

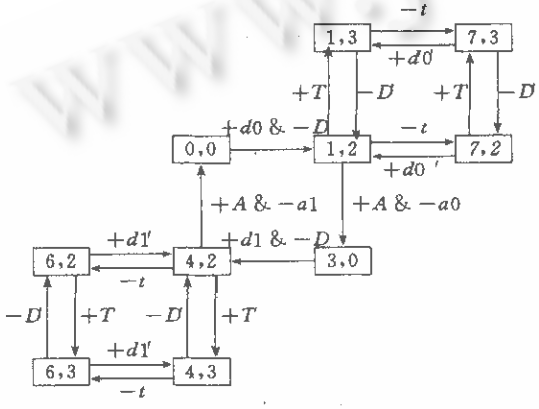


图8　交替位协议和无序号协议关于K的转换器

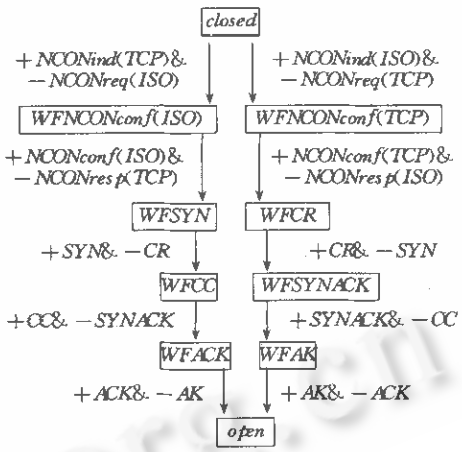


图9　*TCP*和*ISO/TP*连接建立部分的转换器

如下:

$$\begin{aligned}\overline{K}=\{&\langle+NCONind(TCP),-NCONreq(ISO)\rangle,\\&\langle+NCONconf(ISO),-NCONresp(TCP)\rangle,\\&\langle+NCONind(ISO),-NCONreq(TCP)\rangle,\\&\langle+NCONconf(TCP),-NCONresp(ISO)\rangle,\\&\langle+SYN,-CR\rangle,\langle+CC,-SYNACK\rangle,\langle+ACK,-AK\rangle,\\&\langle+CR,-SYN\rangle,\langle+SYNACK,-CC\rangle,\langle+AK,-ACK\rangle\}.\end{aligned}$$

则协议 *TCP* 和 *ISO/TP* 的连接建立部分的对偶积为图 9 所示,它类似于文献[7]中的转换器图 6.

现在我们讨论一下 Shu 和 Liu 的转换器. 在他们的模型中,$[A,B]$和$[G,H]$的转换表示为$[A,B',G',H]$. B'和G'是在B和G中插入 *PUTs* 和 *GETs* 得到的. 在B'和G'之间除了成对的 *PUT* 和 *GET* 外无其它信息交换. 这时协议转换器实际上是$C'=B'\otimes^{\overline{K}}G'$,其中$\overline{K}=\{\langle PUT_i,GET_i\rangle|i=1,2,\cdots,k\}$,即$\overline{K}$是出现在$B'$和$G'$中的所有 *PUT*—*GET* 对偶集合,

而[A,C',H]与[A,B',G',H]的全局状态图一样. 这样构造的C'比我们从B和G构造的C复杂. 而[A,C',H](或[A,B',G',H])的全局状态图就更复杂了.

4 结 论

我们给出了两个协议的转换器的形式定义. 给定两个协议和两个协议的重要信息对偶集合,我们可以容易地用两个 cfsm 的对偶积来构造协议转换器. 不同的重要信息对偶集合对应不同的协议转换器. 在某些转换器中确认有端到端的意义,在另一些转换器中确认只是逐段的,没有端到端的意义. 重要信息对偶的选择根据需要确定.

关于协议的性质在文献中有过很多讨论,如有无死锁. 协议转换是在两个协议[A,B]和[G,H]的基础上构造一个新的协议[A,C,H]. 如果原来的协议无死锁,[A,C,H]能否保持原来协议的这个基本性质呢? 需要什么条件? 关于这些问题我们将另文讨论.

参考文献

1 Green P E. Protocol conversion. IEEE Trans. Commun., 1986, **COM—34**: 257—268.

2 Brand D, Zafiropulo P. On communicating finite state machines. J. ACM, 1983, **30**: 323—342.

3 Okumura K. A formal protocol conversion method. Proc. ACM SIGCOMM'86 Symposium, 1986. 30—37.

4 Shu J C, Liu M T. A synchronization model for protocol conversion. Proc. IEEE INFOCOM'89 Symposium, 1989. 276—284.

5 Calvert K L, Lam S S. Formal methods for protocol conversion. IEEE J. Select. Areas Commun., 1990, **SAC—8**: 127—142.

6 Merlin P, Bochmann G V. On the construction of submodule specification and communication protocols. ACM Trans. Programm. Lang. Syst. 1983, **5**: 1—25.

7 Groenback I. Conversion between the TCP and ISO transport protocols as a method of achieving interoperability between data communications systems. IEEE J. Select. Areas Commun., 1986, **SAC—4**: 288—296.

PROTOCOL CONVERTER AND ITS CONSTRUCTION

Zhao Jinrong

(*Computer Center, Tsinghua University, Beijing* 100084)

Abstract A protocol converter can be regarded as a mapping between messages of two protocols. This mapping can be characterized by a set K of coupled significant messages. In the present paper a formal definition of the protocol converter with respect to K is given, and proposes a method for constructing the converter which is based on the coupled product of two finite state machines. The correctness of the method is proved.

Key words Protocol converters, significant message mapping, coupled product, homomorphism.