对一种公平非否认协议的新改进

雷新锋 刘 军 肖军模

(解放军理工大学通信工程学院 南京 210007)

摘 要 Zhou-Gollmann协议是一种公平非否认协议,近年来得到广泛讨论。Kim发现该协议在时限公平性方面存在缺陷,针对该缺陷提出一种改进的协议,但其改进方法高度依赖于网络时钟的同步。通过详细分析,发现在缺乏时钟同步时Kim的改进协议也可导致不公平。针对此问题,提出一种新的改进方案。新的改进消除了协议对时钟同步的依赖性,保持了协议的公平非否认性,且不会降低协议的效率。

关键词 公平性,非否认性,协议,时钟同步

中图法分类号 TP309 **文献标识码** A

New Improvement on a Fair Non-repudiation Protocol

LEI Xin-feng LIU Jun XIAO Jun-mo

(Institute of Communications Engineering, PLA University of Science and Technology, Nanjing 210007, China)

Abstract Zhou-Gollmann protocol is a fair non-repudiation protocol which is widely discussed in recent years. Kim found a flaw of this protocol on time-limited fairness. Aimming at this flaw, Kim gave an improved protocol, but the improved schema highly depended on synchronization of network clock. We analyzed Kim's protocol in detail and show that, without clock synchronization, Kim's improved protocol is also unfair. Then we proposed a new schema to improve it further. Without reducing the efficiency of the original protocol, our new method avoids depending on synchronization of the clock, and keeps fairness and non-repudiation of the protocol.

Keywords Fairness, Non-repudiation, Protocol, Clock synchronization

在电子商务协议中,不可否认性与公平性是两种必须具备的属性。不可否认性是指主体在协议完成后对自己的发送或接收行为不可否认,公平性是指协议不应当使一方与另一方相比处于不利地位。

Zhou和Gollmann[1]提出的一种公平非否认协议(下面简称ZG协议)近年来得到广泛的分析与讨论。文献[2]基于SVO逻辑[3]验证了ZG协议的不可否认性。文献[4]在不考虑协议主体可能删除相关消息的情况下验证了协议的公平性。考虑到现实中协议主体不可能永久保存相关信息,文献[5]通过分析协议的有限状态发现,如果允许可信第三方在一定时间后删除之前所公布的信息,则协议的发起者可通过打乱协议执行顺序的方式破坏协议的公平性。可见,为了保持公平性,必须考虑时间限制问题。事实上,文献[1]同时给出了一种考虑时间限制的协议版本(下面简称ZG1协议),其目的是允许可信第三方在一个预先商定好的时间删除公布的信息。然而,文献[6]发现该版本的协议还存在另一种与时间限制相关的公平性缺陷,即发送方可通过恶意拖延时间的方式破坏协议的公平性,因此,通过在协议消息中进一步加入时间信息的方法对其进行了改进。改进后的协议(下面简称ZG2)虽然解决了协议主体恶意拖延时间的问题,但该方法高度依赖于网络的时钟同步。本文为了在解决ZG协议在时限公平性方面所存在缺陷的同时,消除协议对时钟同步的依赖,并保持协议的执行效率,提出了一种新的改进协议(下面简称NIZG)。

本文第1节对ZG协议及其主要改进进行介绍;第2节分析ZG2协议并提出一种新的改进协议;第3节对新改进的协议进行分析;第4节对协议的其它性质及其效率作进一步讨论;最后总结全文。

1 ZG协议及其改进

首先对协议中的一些通用术语进行解释。

- $eK(X)$,$dK(X)$:分别表示以密钥K对消息X进行加密或解密;
- $sK(X)$:以私钥K对消息X的数字签名;
- P_A,S_A:主体A的公钥和私钥;
- $A \rightarrow B: X$:主体A发送消息X给主体B;
- $A \leftrightarrow B: X$:主体A使用"*ftp get*"操作从主体B处获取消息X。

1.1 ZG协议

ZG协议主要包括3个主体:发送方A、接收方B及可信第三方T。其主要思想是:发送方A将消息加密后发送给接收方B,然后在收到B发回的接收不可否认证据后,将解密密

到稿日期:2008-08-24 返修日期:2008-12-15 本文受江苏省自然科学基金(KB2008090)资助。

雷新锋(1973—),男,博士生,主要研究方向为信息安全、形式化技术,E-mail:leixinfeng@163.com;刘 军(1969—),男,副教授,主要研究方向为信息安全、软件工程;肖军模(1947—),男,教授,博士生导师,主要研究方向为信息安全、软件工程。

钥提交给 T，由 T 在其公开目录公布这一密钥。B 通过访问公开目录获得解密密钥后，对之前所收到的密文进行解密，从而获得所需信息。协议的具体描述如下：

(1) $A\rightarrow B: f_{NRO}, B, L, C, NRO$

(2) $B\rightarrow A: f_{NRR}, A, L, NRR$

(3) $A\rightarrow T: f_{SUB}, B, L, K, sub\_K$

(4) $B\leftrightarrow T: f_{CON}, A, B, L, K, con\_K$

(5) $A\leftrightarrow T: f_{CON}, A, B, L, K, con\_K$

其中，

- M：A 欲发送给 B 的消息；
- L：将特定协议轮次中所有消息联系起来的独一无二的标签；
- C：用密钥 K 对消息 M 进行加密的密文；
- K：由 A 产生的密钥；
- $f_{NRO}, f_{NRR}, f_{SUB}, f_{CON}$：协议消息的意图标志；
- $NRO=sS_A(f_{NRO}, B, L, C)$：消息 M 的发送不可否认证据；
- $NRR=sS_B(f_{NRR}, A, L, C)$：消息 M 的接收不可否认证据；
- $sub\_K=sS_A(f_{SUB}, B, L, K)$：密钥 K 的提交证据；
- $con\_K=sS_T(f_{CON}, A, B, L, K)$：可信第三方发布密钥 K 的确认。

在该协议中，A 要证明 B 收到了消息 M，需要 NRR 和 con_K 两个证据，B 要证明 A 发送了消息 M，也需要 NRO 和 con_K 两个证据；从理想情况出发，假设主体不会删除相关消息，则该协议满足公平性。因为在 B 获得消息 M 时，A 与 B 也都获得了对方不可否认的相关证据。

1.2 ZG1 协议

ZG 协议在理想情况下可获得公平性，但从实际应用的角度来看，主体不可能做到对消息的永久保存，这一点可导致协议的不公平。以下是两种可能的情形：

(1) B 删除加密消息 C。如果 A 恶意拖延提交密钥 K 的时间，B 将在很长时间内无法从 T 处获得 K，会误以为 A 单方面结束了协议，从而删除加密消息 C，但 A 却在随后提交了密钥 K。这样，A 通过从 T 处获取 con_K，连同之前收到的 NRR，便可证明 B 收到了 M。而此时 B 由于已将 C 删除，即使能从 T 处获得 K，也无法恢复消息 M。这一情形对 B 来说是不公平的。

(2) T 删除密钥 K。文献[5]认为，在该协议中，A 其实可以先执行第 3 步，即首先提交密钥 K 给 T，并在 T 公布相关信息后从 T 处获取 con_K。T 在认为时间足以使 A，B 获取所公布的信息后将公布信息删除，而事实上 B 对此一无所知。这时如果 A 再向 B 发送密文 C，则 B 为了获得 K，将向 A 发送 NRR。从而使得 A 能够证明 B 获得了 M，而 B 由于得不到 K 而无法获得 M，造成对 B 的不公平。

可见，要达到真正的公平，必须有明确的时间限制。基于这一点，文献[1]进一步提出一种包含时间限制的协议，即 ZG1 协议。

(1) $A\rightarrow B: f_{NRO}, B, L, t, C, NRO$

(2) $B\rightarrow A: f_{NRR}, A, L, NRR$

(3) $A\rightarrow T: f_{SUB}, B, L, t, K, sub\_K$

(4) $B\leftrightarrow T: f_{CON}, A, B, L, t, t_0, K, con\_K$

(5) $A\leftrightarrow T: f_{CON}, A, B, L, t, t_0, K, con\_K$

其中，t 为密钥 K 可被公开访问的最终时刻，t_0 为密钥 K 实际被 T 公布的时刻；

$NRO=sS_A(f_{NRO}, B, L, t, C)$；

$NRR=sS_B(f_{NRR}, A, L, t, C)$；

$sub\_K=sS_A(f_{SUB}, B, L, t, K)$；

$con\_K=sS_T(f_{CON}, A, B, L, t, t_0, K)$。

协议约定由 A 确定一个时限 t，并将该时限通过协议第(1)、(3)步通知 B 和 T，使得可信第三方可在 t 时之后删除 K 及 con_K。如果 A 在 t 之后提交密钥，T 可拒绝公布该密钥。如果 B 对此时限的设定有异议，可以选择结束协议。该方案保证了主体不必永久保存相关信息，也使得 A 不能无期限拖延提交密钥，避免了(1)、(2)两种情形下的不公平。

1.3 ZG2 协议

虽然 ZG1 协议比 ZG 协议更为实用，但 Kim[6] 发现 ZG1 协议并不能完全防止 A 的恶意拖延行为。具体来说，如果 A 在 t 到来的前一刻才提交密钥 K，那么 K 及 con_K 将在公布之后的很短时间内就被删除。期间，由于 A 占据主动，它可以随时监控以获取 con_K，同时设法干扰 B，使 B 无法获取密钥 K。这样 A 就占据了优势，破坏了公平性。针对以上缺陷，Kim 在 ZG1 的基础上提出 ZG2 协议。ZG2 协议可描述如下：

(1) $A\rightarrow B: f_{NRO}, B, L, t, C, NRO$

(2) $B\rightarrow A: f_{NRR}, A, L, t_1, NRR$

(3) $A\rightarrow T: f_{SUB}, B, L, t, K, sub\_K$

(4) $B\leftrightarrow T: f_{CON}, A, B, L, t, t_0, K, con\_K$

(5) $A\leftrightarrow T: f_{CON}, A, B, L, t, t_0, K, con\_K$

其中，t_1 ($<t$) 用以约束 NRR 的有效性，即如果 $t_0 > t_1$，则 NRR 失效；

$NRO=sS_A(f_{NRO}, B, L, t, C)$；

$NRR=sS_B(f_{NRR}, A, L, t, t_1, C)$；

$sub\_K=sS_A(f_{SUB}, B, L, t, K)$；

$con\_K=sS_T(f_{CON}, A, B, L, t, t_0, K)$。

协议 ZG2 假定任何主体对其它主体的干扰都是有限的，只要时间足够，主体总能获取可信第三方公布的信息。因此，约定由 B 确定一个时限 t_1，以确保在 t_1 到 t 这一时间段内 B 能够获取公布信息。在协议第 2 步，B 将该时限通知 A，如果 A 的拖延使得密钥的公布时间超过了 t_1，则 A 之前所收到的 NRR 将会失效，从而限制了 A 的恶意拖延行为。如果 B 在 t_1 至 t 这段时间内没有从 T 处获取密钥 K，则 B 可以放心地删除 C 及 NRO。T 在 t 之后也可以放心地删除所公布的密钥信息了。

在该协议中，Kim 假设各协议主体的时钟是严格同步的，并证明了协议的公平性，有效地防止了 A 的恶意拖延行为。

2 ZG 协议的新改进(NIZG)

Kim 的方案假设各协议主体的时钟是严格同步的，这一点限制了其在现实中的应用。因此，对于 ZG2 协议来说，人们关心在缺乏专门时钟同步机制的情况下，协议能否保持公平非否认性。如果不能，是否可设计一种新的方案以消除协议对时钟同步的依赖性。

2.1 ZG2 协议存在的问题

Kim 本人认识到 ZG2 需要高精度时钟同步机制的支持，甚至建议使用 GPS 全球定位系统提供同步机制，但没有对如何消除协议对时钟同步的依赖性做进一步分析。文献[8,9]致力于消除 ZG2 协议对时钟同步的依赖性，但也没有给出在缺乏时钟同步机制的情况下会导致什么样的结果。为了使对协议的改进更具针对性，以下在取消时钟同步假设的情况下对 ZG2 可能存在的问题进行分析。

在缺乏时钟同步时，同一个时间值在不同的环境下可能对应于不同的时间。这种不同步可能会对某些主体在一些关键的时刻产生不利的影响。我们采用排除法来定位时钟同步对 ZG2 协议的影响：

(1) 协议中，t_0 仅起到宣示的作用，不具备约束力，A 与 B 只关心其标称值，而不关心其是否受同步的影响。

(2) B 不用担心时钟同步的影响。协议中 B 只关心 t_1 与 t 之间的相对时间间隔是否足以使其获取密钥。当 t_1 和 t 分别确定后，即使时钟不同步，其时间间隔也可得到确定。

(3) T 在该协议中没有利益瓜葛，不用担心时钟同步的影响。

(4) A 不用担心与 B 的时钟不同步，因为密钥的发布是由 T 进行的，而 B 无法影响这一行为。

(5) A 不用担心与 T 在 t 上的不同步，因为 t 主要影响对 T 所公布消息的获取，而 T 所公布的消息依赖于 A 所提交的消息，因此 A 可在 T 公布消息的第一时间获取该消息。如果受时钟同步的影响，A 不能在 T 删除所公布消息前获取该消息，则 B 也将无法获取，不会对 A 造成不利。

至此，只剩下 A 对其自身与可信第三方 T 在时间 t_1 上是否同步的担心了。以 T 的时间为基准，设 t_1 在 A 的本地环境中所对应的时间为 t_{1A}，则有两种可能：$t_{1A}\leqslant t_1$ 或 $t_{1A}>t_1$。

(6) 在 $t_{1A}\leqslant t_1$ 这种情况下，T 有足够的时间在 t_1 之前发布密钥信息，即保持 $t_0\leqslant t_1$，从而不会使 A 收到的 NRR 失效。对于 A 来说，在随后获得 con_K 后便可证明 B 收到了消息 M。这种情况对 A 并无不利。

(7) 在 $t_{1A}>t_1$ 这种情况下，A 的估计受到了时钟不同步的影响，可造成 T 的实际公布时间 $t_0>t_1$，并使得 A 之前收到的 NRR 失效。但对 B 来说，只要 T 公布了密钥信息，便有足够的时间获得密钥 K，并通过解密得到消息 M。如果之后 B 声称未收到过 M，由于 A 所掌握的 NRR 已失效，因此无法对 B 提出反驳，造成对 A 的不公平。

由此可见，在缺乏时钟同步的情况下，ZG2 协议不能保持对 A 的公平性。为此，提出 ZG 协议的一种新的改进(NIZG)，以期消除协议对时钟同步的依赖性。

2.2 NIZG 协议

在正式给出新的改进之前，需要声明的是，NIZG 的主要目的是消除协议对时钟同步的依赖性，但并不意味着时间可以处于任意失序状态。假定同一时间值在不同环境之间的误差存在一个上限 t_u。这一假定从现实的角度来说是合理的，且防止了将时钟误差任意扩大化的趋势。

对新改进的协议描述如下：

(1) $A\rightarrow B: f_{NRO}, B, L, t, C, NRO$

(2) $B\rightarrow A: f_{NRR}, A, L, t_1, NRR$

(3) $A\rightarrow T: f_{SUB}, B, L, t, t_1, K, sub\_K$

(4) $B\leftrightarrow T: f_{CON}, A, B, L, t, t_1, t_0, K, con\_K$

(5) $A\leftrightarrow T: f_{CON}, A, B, L, t, t_1, t_0, K, con\_K$

其中，t_1 为 T 公布密钥相关信息的最终时刻；

$NRO=sS_A(f_{NRO}, B, L, t, C)$；

$NRR=sS_B(f_{NRR}, A, L, t, t_1, C)$；

$sub\_K=sS_A(f_{SUB}, B, L, t, t_1, K)$；

$con\_K=sS_T(f_{CON}, A, B, L, t, t_1, t_0, K)$。

在 NIZG 协议中，t_1 不再用以约束 NRR 的有效性，而直接表示可信第三方 T 提交密钥信息的时限。主体 A 在协议第 3 步发送给 T 的消息中包含了 t_1，同时在 T 所发布的消息中也包括了 t_1。协议要求 T 在收到 A 的提交信息后，首先判断当前时间是否小于 t_1。若小于，则发布密钥信息，否则不予发布。另外，T，B 都不必永久保存相关信息。T 可以在其本地时间的 t 之后删除公布信息。B 如果未获得密钥 K，最迟也可在其本地时间($t+t_u$)后删除密文 C。

NIZG 与 ZG2 最大的不同在于，在 ZG2 中，t_1 的约束对象是 A；而在 NIZG 中，t_1 的直接约束对象是 T，而 A 只是间接的约束对象。

新的改进能否消除协议对时钟同步的依赖性，是否具备不可否认性与公平性，需要做出进一步的分析。

3 NIZG 协议分析

首先需要分析协议的时钟同步依赖性，以便在之后的分析中不再考虑同步因素。

3.1 时钟同步依赖性

分析 NIZG 协议时钟同步依赖性采用 2.1 节中对 ZG2 的分析方法。其中前 6 种情况下，二者有类似的结论，此处不再重复。以下着重分析第 7 种情况：

(7)' $t_{1A}>t_1$。这种情况下，由于 A 的估计受到了时钟不同步的影响，可能使得 A 提交的消息到达 T 处的时间已经超过了 t_1，T 因此将不会公布 K 和 con_K。A，B 均无法获得其所需要的消息，这一点对 A，B 来说都是公平的。

造成(7)与(7)'不同的主要原因在于 NIZG 与 ZG2 对公布密钥的处理不同。在 NIZG 中，如果由于 A 的原因使得 T 无法在 t_1 之前公布密钥，T 将不再公布。而在 ZG2 中，t_1 仅仅是 A 与 B 之间的一种约定，T 对 t_1 一无所知。因此，即使因缺乏时钟同步使得 A 的提交消息到达 T 的时间超过 t_1，T 也依然会公布密钥。

可见，NIZG 消除了对时钟同步的依赖性。

3.2 不可否认性

在验证协议的不可否认性时，通常使用 Kailar 逻辑[7]或 SVO 逻辑[3]。篇幅所限，我们不打算给出详细的形式化验证过程。但在以下分析中，保持了使用 SVO 逻辑验证不可否认协议[2]的思路。

不可否认性分两种情况：A 否认曾经发送过消息 M 给 B；B 否认曾经收到过 A 发来的消息 M。即发送不可否认和接收不可否认。假设存在一个仲裁者，且仲裁者拥有各主体的公钥，可对 A，B，T 的签名消息作出验证。同时，如果仲裁者看到 K 及 con_K，并可验证 con_K 有 T 的签名，则他相信 A 向 T 提交过 K，相信 T 公布了 K 及 con_K，并进而相信 A，B 均可获得 T 所公布的消息。以下分别对发送不可否认及接收不可否认进行分析。

发送不可否认：如果 A 声明没有发送过 M 给 B，则 B 可

将 M,C,K,L,t,t_1,t_0,NRO 以及 con_K 提交给仲裁者。仲裁者可根据以下过程证明 A 确实发送了 M：

(1)检查 con_K 是可信第三方 T 对($f_{CON},A,B,L,t,t_1,t_0,K$)的签名结果；

(2)检查 NRO 是 A 对(f_{NRO},B,L,t,C)的签名结果；

(3)检查 $M=dK(C)$。

其中(1)表明 A 提交过密钥 K；(2)表示 A 发送过密文 C；(3)表明 C 是用 K 对 M 进行加密的结果。(1),(2),(3)构成一个证据链，证明 A 发送了消息 M。

接收不可否认：如果 B 声明没有收到过 A 发的 M，则 A 可将 $M,C,K,L,t,t_0,t_1,NRR,con\_K$ 提交给仲裁者。仲裁者可根据以下过程证明 B 确实收到了 M：

(1) 检查 con_K 是可信第三方 T 对($f_{CON},A,B,L,t,t_1,t_0,K$)的签名结果；

(2) 检查 NRR 是 B 对(f_{NRR},A,L,t,t_1,C)的签名结果；

(3) 检查 $M=dK(C)$。

其中(1)表明 T 公布了密钥 K，进而表明 B 可获得密钥 K；(2)表明 B 曾收到过密文 C；(3)表明 C 是用 K 对 M 进行加密的结果。(1),(2),(3)构成一个证据链，证明 B 收到了消息 M。

与 ZG2 不同，在 NIZG 中仲裁者甚至不需要检查 t_1 是否小于 t_0，因为如果这一关系不满足，T 是不会公布 con_K 的。

3.3 公平性

一般而言，如果协议的一方主体在任何一步终止协议都不会使另一方处于不利地位，则称该协议是公平的[4]。该定义未考虑以下情况：主体虽不打算终止协议，但故意拖延执行协议时间，使得对方处于不利地位。鉴于此，我们将公平性的定义进行扩充：如果协议能够保证不使任何一方主体处于不利地位，则称该协议是公平的。同时，将以上所提到的两种情况分别称为终止公平性和时限公平性。

首先，分析 NIZG 协议的终止公平性。

(1) 在协议执行第 1 步后，B 单方面终止协议。由于该情况下 B 只收到了密文 C，尚未获得密钥 K，从而无法获取其中的消息 M，因此不会使 A 处于不利地位。

(2) 在协议执行第 2 步后，A 单方面终止协议。由于 A 只收到了 NRR，而未收到 con_K，因此不能证明 B 收到了 M，不会对 B 造成不利。

(3) 协议第 3 步后，需要 T 公布密钥信息。由于 T 是可信的，它不会无故终止协议。但即使 T 由于不可抗拒的原因终止了协议，导致无法公布密钥相关信息，则 B 将无法获得 K 以得到 M，同时也无法获得 con_K 以证明 A 发送过 M，从而不会使 A 处于不利地位。另外，A 将无法获得 con_K 以证明 B 收到过 M，也不会使 B 处于不利地位。

(4) 协议第 4,5 步后，A,B 都无法控制对方对 T 所公布的信息的获取。

可见，NIZG 协议满足终止公平性。

其次，分析 NIZG 协议的时限公平性。

(1) 在协议执行第 1 步后，B 拖延执行协议。这种情况只会使 B 自身无法及早获取密钥，而不会对 A 造成不利。

(2) 在协议执行第 2 步后，A 拖延执行协议。如果 A 的拖延使得 T 无法在 t_1 之前公布 Con_K 和 K，则 A,B 都无法获得所需要的信息，B 可在($t+t_u$)后放心地删除密文 C，而不用担心 A 拿出对自己不利的证据。

(3) 在协议执行第 3 步后，T 不会恶意拖延公布密钥的时间。即使由于不可抗拒的原因，使得时间超过了 t_1，T 将选择不公布密钥，从而不会使 A,B 任何一方处于不利地位。在 t 之后，T 也可删除密钥信息，而无须永久保存。

可见，协议是满足时限公平性的。

4 进一步讨论

在分析了协议的主要性质后，对其它一些性质以及协议的效率做进一步的讨论。

(1) 关于时限消息的完整性。读者可能会发现，在 NIZG 协议中，时限 t_1 是由 B 给出的，但却是由 A 提交给 T 的。如果 A 恶意将提交给 T 的 t_1 篡改到 t 的前一刻，则会产生类似于 ZG1 中对 B 不利的情形。即 A 在获取 con_K 的同时干扰 B，使 B 无法在 t 之前获得 K。事后，A 可诬陷 B 收到了消息 M。然而，这一诉求是不会得到仲裁者的支持的，因为为了证明 B 收到了消息 M，A 必须向仲裁者提交 t_1，NRR 及 con_K 等消息。而在 NRR 中包含了 B 对 t_1 的签名，A 对 t_1 的篡改将使得 NRR 无法通过仲裁者对签名进行验证。

(2) 关于协议的保密性。与 ZG 协议类似，NIZG 未涉及协议的保密性，即在 t_0 之后，任何人都可以从 T 的公开目录中获取密钥 K。但这一点并不影响对协议其它属性的讨论。Kim 在文献[4]中已对 ZG 的保密性提出解决方案，因此如果需要，可很容易地将其方案应用于 NIZG 协议。

(3) 与相关工作的比较。与 ZG,ZG1,ZG2 相比，NIZG 克服了它们所存在的缺陷，但并未降低原协议的效率：协议的总交互次数均为 5 次，可信第三方的参与次数也均为 3 次，且通信量相当。与其它同类研究相比，在文献[8]对 ZG 协议的改进中，可信第三方的参与次数增加为 4 次，且可信第三方还承担了额外的证据转发、证据检索、证据匹配等任务，增加了可信第三方的负担。在文献[9]对 ZG 协议的改进中，虽然显式出现的总交互次数为 5 次，可信第三参与 3 次，但主体 A,B 各需要从 T 的公开目录获取 T 的当前时间 1 次，使得真正的交互次数为 7 次。其中可信第三方参与 5 次，降低了协议的执行效率。同时，由于网络时延的不确定性，A,B 所获得的时间值无法保证其准确度。可见，NIZG 的改进在克服原协议缺陷的同时，并未以降低协议效率为代价。

结束语 本文在深入研究 ZG 协议及其几种改进所存在缺陷的基础上，提出一种新的改进协议 NIZG。新的改进消除了协议对时间同步的依赖性，降低了协议对使用环境的要求，同时保持了协议的不可否认性与公平性。在协议效率方面，NIZG 未增加协议的交互次数及可信第三方的负担，保持了协议的高效性。

参 考 文 献

[1] Zhou J, Gollman D. A fair non-repudiation protocol[A]// Proc. of the 1996 IEEE Symp. on Security and Privacy[C]. Los Alamitox: IEEE Computer Society Press, 1996: 55-61

[2] Zhou J, Gollmann D. Towards verification of non-repudiation protocols[A]// Proceedings of the 1998 International Refinement Workshop and Formal Methods Pacific[C]. Canberra, Australia, 1998: 370-380

(下转第 148 页)

式。要应用点规则，就必须：说明用来替换的项是正确类型的对象；对谓词中这些变量的出现进行替换。

由(1)，可以得到一个新的等价式中的6个部分：

$\emptyset \in FComponent\_instances$

$\emptyset \in FConnector\_instances$

$\emptyset \in FConnector\_instances$

$\emptyset \in FConnector\_instances$

$\emptyset \in Component\_instance \vdash\!\!\!\rightarrow Connector\_instance$

$\emptyset \in Connector\_instance \vdash\!\!\!\rightarrow Connector\_instance$

由(2)，可以得到进一步的限制：$dom\emptyset \subseteq \emptyset \wedge ran\emptyset \subseteq \emptyset \wedge \emptyset = \emptyset$。

所以由定理1所描述的初始化定理现在就成了：

$\vdash \emptyset \in FComponent\_instances \wedge \emptyset \in FConnector\_instances \wedge \emptyset \in FConnector\_instances$

$\wedge \emptyset \in FConnector\_instances \wedge \emptyset \in Component\_instance \vdash\!\!\!\rightarrow Connector\_instance$

$\wedge \emptyset \in Connector\_instance \vdash\!\!\!\rightarrow Connector\_instance \wedge dom\emptyset \subseteq \emptyset \wedge ran\emptyset \subseteq \emptyset \wedge \emptyset = \emptyset$

第一个到第四个子目标可由定律 $\emptyset \in FS(L24)$ 立即得到证明，第五个和第六个子目标可由定律 $\emptyset \in S \vdash\!\!\!\rightarrow T(L13)$ 得到证明，第七个子目标可由定律 $\mathrm{dom}\emptyset = \emptyset(L6)$ 得到证明，第八个子目标可由定律 $ran\emptyset = \emptyset(L7)$ 得到证明，最后 $\emptyset = \emptyset$ 显然成立。所以定理成立，即对于该规格说明，初始状态式存在的，也就是说我们验证了该规格说明的正确性。

5 相关工作

对于在软件设计阶段软件体系结构的重用，国内外研究机构提出了很多不同的方法，下面是目前具有代表意义的研究成果的介绍以及它们和本文提出的基于反射机制的软件体系结构重用方法对比研究：

面向领域的体系结构重用：如产品线体系结构、特定领域的体系结构等[5]。特定领域的体系结构方法需要针对特定领域。而本文的软件体系结构重用方法具有通用性。

体系结构设计知识的重用：如体系结构风格[6,7]、参考体系结构等。然而把体系结构设计知识以软件制品的形式进行重用时，仍然面临着许多重大的技术障碍。

软件框架可以将体系结构的设计方案连同其实现代码一起进行重用，是支持实现阶段而非设计阶段的重用。

本文提出的基于反射的体系结构重用方法是一种更通用、更便捷的体系结构制品本身的重用方法，它具有统一的体系结构建模方法的信息，利用反射机制来屏蔽基级体系结构描述语言来实现体系结构设计时制品的重用，所以本文以及后续的研究内容在一定程度上可以部分解决软件体系结构重用存在的主要问题。

结束语 本文详细给出了基于反射机制的反射式软件体系结构的基级元素模型和元级元素模型，用形式规格说明语言 Object-Z 对基级元素模型进行了完整的描述，选取基级元素模型的一个代表性模式，给出它的初始化定理及其证明过程。

我们的研究工作与已有研究不同的地方在于：(1)提供了一种软件设计阶段重用软件体系结构及其组成元素的方法。(2)对反射式体系结构的基级元素模型进行了形式化及其推理证明。

作为今后的工作，我们将对元信息模型不断丰富和完善，即完善 PMB 协议的定义，开发 RMRSA 的支撑工具，以及用形式化方法 Object-Z 语言描述 PMB 协议和元级体系结构模型，给出形式化反射式体系结构的完整初始化和相关性质的定理及其证明过程，以此说明反射式体系结构的正确性及其元级和基级一致性。

参 考 文 献

[1] Bass L, Clements P, Kazman R. Software Architecture in Practice, Second Edition[M]. Addison Wesley, April 2003

[2] Mili H, Mili A, Yacoub S. Reuse-based Software Engineering: Techniques, Organization, and Controls[M]. Jonh Wiley & Sons Ltd., 2001

[3] Keller R K, Schauer R. Design Components: Towards Software Composition at the Design Level[J]. ICSE, 1998: 302-311

[4] Medvidovic N, Rosenblum DS, Taylor RN. A language and environment for architecture-based software development and evolution[C]// Proc. of the 21st Int'l Conf. on Software Engineering. New York: ACM Press, 1999: 44-53

[5] Binns P, Engelhart M, Jackson M, et al. Domain-Specific Software Architectures for Guidance, Navigation, and Control[J]. Int'l J Software Eng and Knowledge Eng, 1996, 6(2)

[6] Monroe R T, Garlan D. Style-based reuse for software architectures[C]// Proceedings of the 4th International Conference on Software Reuse. Orlando, FL, USA, 1996

[7] Schmerl B, Garlan D. AcmeStudio:: Supporting Style-Centered Architecture Development[C]// Proceedings of International Conference on Software Engineering. Edinburgh, Scotland, May 2004

(上接第97页)

[3] Syverson P F, van Oorschot P C. On unifying some cryptographic protocol logics[A]// Proceedings of the 1994 IEEE Computer Society Symposium on Research in Security and Privacy[C]. Los Alamitos: IEEE Computer Society Press, 1994: 14-28

[4] 卿斯汉. 一种电子商务协议形式化分析方法[J]. 软件学报, 2005, 16(10): 1757-1765

[5] 董荣胜，陈大伟，郭云川，等. 公平非否认协议的有限状态分析[J]. 计算机科学, 2005, 32(8): 83-86

[6] Kim K, Park S, Baek J. Improving fairness and privacy of Zhou-Gollmann's fair non-repudiation protocol[A]// Proceedings of the 1999 ICPP Workshops on Security[C]. Aizu, Japan, 1999: 140-145

[7] Kailar R. Accountability in electronic commerce protocols[J]. IEEE Transactions on Software Engineering, 1996, 22(5): 313-328

[8] 黎波涛，罗军舟. 不可否认协议的一种新的改进[J]. 计算机学报, 2005, 28(1): 35-45

[9] 蔡永泉，朱勇. 非否认协议的分析与改进[J]. 计算机应用研究, 2007, 24(7): 242-245