基于OPN的指控能力包服务动态组合方法研究

皇甫先鹏 陈洪辉 罗雪山
(国防科学技术大学C4ISR技术国防科技重点实验室 长沙 410073)

摘 要 服务组合方法对于分布式网络环境下基于SOA的军事应用系统的综合性能影响很大。提出了一种基于OPN的服务动态组合方法的解决方案。首先定义了基于对象Petri网的指控能力包服务描述和组合模型,对服务组合的数学算子进行了分析和证明,而后给出了基于OPN的服务动态组合流程,最后使用国防科大对象Petri网建模仿真环境工具对提出的服务组合模型进行建模仿真,并对实验数据进行了分析。

关键词 对象Petri网,指控能力包,服务动态组合
中图法分类号 TP393 **文献标识码** A

Research on C2 Capability Package Service Dynamic Composition Method Based on OPN

HUANGFU Xian-peng CHEN Hong-hui LUO Xue-shan
(C4ISR Technology National Defense Science and Technology Key Lab, National University of Defense Technology, Changsha 410073, China)

Abstract Service composition method is of great importance for military information system based on SOA in the distributed environment, the paper proposed a service dynamic composition method. Firstly the paper defined command and control capability package service description and composition model based on object Petri-net, analysed and proved service composition's arithmetic operators, then brought forward service dynamic composition flow chart, finally made use of object Petri-net modeling and simulation environment of national university of defense technology to simulate service composition model the paper has brought forward, and in the end analysed the experimental datum.

Keywords Object petri net, C2 capability package, Service dynamic composition

随着信息化发展和软件架构思想的进步,面向服务架构SOA的思想越来越受到关注,民用技术中,Web服务作为SOA的一种具体实现技术得到了长足发展。在我军C4ISR信息化发展道路上,基于SOA架构思想的指控能力包构建也在紧锣密鼓的研制当中。单一的服务只能解决单一的应用,服务组合(composition)才能实现复杂的应用,满足军事系统的需求。根据文献[1]的定义,服务组合是将已存在的服务,按照一定的规则动态地发现,并组装成为一个增值的、更大粒度的服务或系统,以满足用户的复杂需求,并提高软件生产率的过程。

当前,服务组合方法主要有3种:基于工作流的动态组合、基于语义网络、领域本体的方法及基于PI-演算或Petri网等工具的形式化方法[2]。文献[3]使用语义代理Agent的工作流方法实现了服务的自动组合;文献[4,5]基于语义网络建模,实现了服务的自动化和半自动化匹配,文献[6]基于领域本体的建模方法实现了服务的动态组合;基于Petri网对服务进行描述并实现服务组合建模也是常用的方法,文献[7]给出了4种基本服务组合模型和4种高级服务组合模型,并对数学性质进行了分析,文献[2,8,9]分别基于Petri网或有色Petri网建模,分析了服务组合的方法和性质。

面对复杂、对抗激烈的军事应用,基于服务的指控能力包在战场环境下面临着调用服务规模大,对服务响应实时性要求高等不同于民用Web服务的问题。基本Petri网存在着一个主要弱点,即规模难以控制。如果系统比较复杂,其Petri网模型的规模将变得难以控制。特别是当系统的可达状态增加时,模型的复杂性将呈指数增长,使Petri网模型的分析和理解变得十分困难,此外有色Petri网对服务建模,是在有色Petri网的基础上扩展输入输出描述实现的,对服务的描述力度不强。国防科大C4ISR技术国防科技重点实验室采用面向对象技术,提出了基于对象的Petri网——OPN,并设计实现了对象Petri网建模仿真环境OPMSE[10],本文基于OPN和OPMSE实现服务的描述、组合、数学算子进行分析和建模仿真。

1 对象Petri网及指控能力包

1.1 对象Petri网OPN

Petri网是一种适合于描述和分析离散事件动态系统尤其是并行系统的模型工具。下面给出Petri网和对象Petri网的定义。

定义1 若四元组$N=(P,T,F,M_0)$满足下述条件,则

到稿日期:2009-10-30 返修日期:2010-01-02 本文受国家自然科学基金(No. 70601036),武器装备预研项目(No. 51306040101)资助。
皇甫先鹏(1982—),男,博士生,主要研究方向为指控能力包构建、服务匹配与组合,E-mail:xphuangfu@163.com;陈洪辉(1969—),男,博士,教授,博士生导师,主要研究方向为军事需求工程、信息栅格技术、服务匹配与组合;罗雪山(1965—),男,博士,教授,博士生导师,主要研究方向为军事信息系统、体系结构技术、网络技术。

称其为 Petri 网：

1）$P \cup T \neq \phi, P \cap T = \phi$；

2）$F \subseteq P \times T \cup T \times P$；

3）$dom(F)cod(F) = p \cup T$；

4）$M_0 : p \rightarrow (0,1,2,\cdots)$是初始标记。

其中 P 是位置(Place)的集合，T 是转移(Transition)的集合，F 称为流关系。

Petri 网中的位置、转移和 Token 所携带的系统信息量不够丰富，缺乏语义描述功能，且 Petri 网不涉及时间因素，仅能反映出事件间的因果关系。Petri 网方法不能进行数据处理，没有层次化的设计思想，无法描述系统内的时序关系，以及 Petri 网模型的状态空间爆炸问题大大限制了其对大型、复杂系统的建模和分析能力。为了适应 C4ISR 系统的特点和建模仿真的需要，国防科大 C4ISR 技术国防科技重点实验室从网的结构、建模元素、执行规则等几个方面对基本 Petri 网进行了扩展，提出了对象 Petri 网的概念，并在此基础上设计了模型描述语言 OPDL。具体的扩展是：

把 Petri 网和面向对象的思想方法结合起来，引进对象的概念和机制。对象作为建模的基本模块，也是模型重用的基本单位，事实上就是封装起来的一张子网。对象间通过接口实现交互，而它们的内部结构用扩展的 Petri 网进行描述。

定义 2 对象 Petri 网是一个八元组 $N = (P, T, SubOBJ, IP, OP, F, M_0, C)$，其中：

1）P 是位置的集合，T 是转移的集合，$SubOBJ$ 是子对象的集合。

2）IP 是对象输入端口的集合，OP 是对象输出端口的集合。输入、输出端口各分为两类，分别称为内部输入端口 IIP、外部输入端口 OIP、内部输出端口 IOP、外部输出端口 OOP。对象通过 IIP 和 IOP 与其子对象交互，通过 OIP 和 OOP 与外部调用者或仿真环境交互。

记 N 的所有子对象的所有外部输入端口的集合和外部输出端口的集合分别为 $SIP(N)$和 $SOP(N)$，即

$SIP(N) = \bigcup_{obj \in SubOBJ(N)} OIP(obj)$，$SOP(N) = \bigcup_{obj \in SubOBJ(N)} OOP(obj)$。

3）F 是一个流关系，$F \subseteq P \times T \cup T \times P \cup IIP \times T \cup T \times IOP \cup OIP \times T \cup T \times OOP$。

4）$C \subseteq IOP \times SIP \cup SOP \times IIP \cup SOP \times SIP$ 是连接关系。

5）$M_0 : (P \cup IIP \cup IOP \cup OIP \cup OOP) \rightarrow (0,1,2,\cdots)$是初始标记。

在图 1 中，位置和端口中的数字表示初始化时的标志即令牌数；对象用中间有双线的方框表示，方框上的是类名，下面是对象名，左边的小的实心长方形是输入端口，右边的小的实心长方形是输出端口；输入端口用同心圆环表示，输出端口用内嵌三角形的圆表示，其中 $S0$ 表示一个开关，收到信道送来的令牌时点火，但并不给后继的每个位置都放置一个令牌，而是执行其动作函数，完成令牌的分发。这样，可以根据信源和信道中设置的令牌类型将令牌分发到合适的位置上去。国防科大 OPMSE 对象 Petri 网建模仿真环境中，共定义了 5 类函数，分别适用于不同的元素，其对应关系如表 1 所列，函数使用 VBScript 语言编写，扩充了元素属性并加强了对建模仿真动作的支持。在此，主要使用的是开关的动作函数。

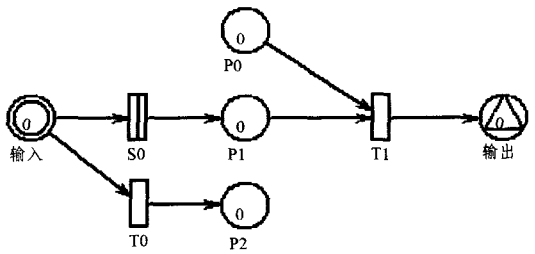

图 1 对象 Petri 网示例

表 1 OPMSE 中函数和元素的关系

函数类型	对象	位置	转移	开关	弧	端口
实例化函数	Y					
事件处理函数		Y	Y	Y		Y
动作函数			Y	Y		
延时函数			Y	Y		
谓词函数			Y	Y		

1.2 指控能力包的层次结构

指控能力包是在一体化信息基础设施基础服务和各军事领域应用服务的基础上，根据指挥控制任务需求和指挥控制活动的特点而构建的、基于 SOA 体系架构并应用多种网络服务建模方法的、满足特定能力的应用服务。

指控能力包支持各种联合作战指挥控制，根据作战任务的特点和需求，调用所需指控能力包动态构建指挥控制系统，完成作战使命。处于网格上的任何一个作战单元可以随时随地获取信息，实现指控能力包的“即插即用”，一旦完成任务，作战单元释放获得的作战资源。指控能力包的这种动态重新装配和再分配作战资源的能力，很大程度上提高了联合作战指挥控制的灵活性。

由图 2 的层次结构可以看出，指控能力包是分层结构，由服务组合而成，所以研究指控能力包服务的组合问题，对于提高指控能力包的综合效能很有意义。

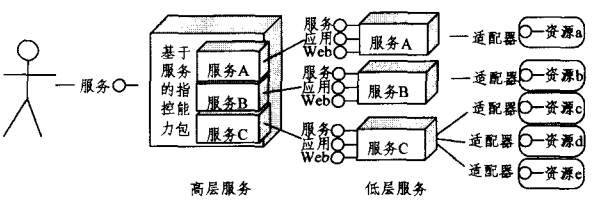

图 2 指控能力包层次结构图

2 基于 OPN 的指控能力包服务组合模型

基于对象 Petri 网建立服务向 OPN 的映射关系，对服务进行形式化描述，在此基础上研究基本的服务组合模型，对服务组合方法的数学算子进行分析并加以证明，最后给出了基于 OPN 服务组合模型和数学算子的动态服务组合流程[11]。

2.1 服务组合模型

从组成框架以及实现的角度讲，服务是一种网络操作，它能够利用标准的网络协议以及接口进行应用间的交互，所以，服务是一些有序操作的集合。OPN 模型由于是已经扩展了输入输出库所的 Petri 网模型，可以直接将服务映射成 OPN 模型，其中服务的状态用位置表示，操作用转移表示，输入输出库所描述服务接口信息，变迁描述其内部逻辑。

定义 3 指控能力包服务的 OPN 形式化描述为一个六元组 $cpS = (P, T, SubS, IP, OP, M_0)$，其中：

1)P 是服务状态的集合，T 是转移的集合，$SubS$ 是子服务对象的集合。

2)IP 是服务输入端口的集合，OP 是服务输出端口的集合。输入、输出端口各分为两类，分别称为内部输入端口 IIP、外部输入端口 OIP、内部输出端口 IOP、外部输出端口 OOP。对象通过 IIP 和 IOP 与其子服务对象交互，通过 OIP 和 OOP 与外部服务调用者或仿真环境交互。

记 cpS 的所有子对象的所有外部输入端口的集合和外部输出端口的集合分别为 $SIP(cpS)$ 和 $SOP(cpS)$，即

$SIP(cpS)=\bigcup_{cps\in SubS(cpS)} OIP(s)$，$SOP(cpS)=\bigcup_{cps\in SubS(cpS)} OOP(s)$。从定义 3，有以下推论。

推论 1 如果 $SubS=\phi$，则 cpS 为不可分原子服务，否则 cpS 为组合服务。

如果 $P\&T=\phi$，则 cpS 为空服务。

如果 $SubS=(SubP, SubT, SubSubS, SubIP, SubOP, SubM_0)$，且 $SubSubS\neq\phi$，则 cpS 为多级嵌套的迭代组合服务。

针对指控能力包服务组合的特点，确定其基本的运算法则有以下 5 条。

定义 4(顺序运算)

服务 $cpS=cpS_1\cdot cpS_2=(P,T,SubS,IP,OP,M_0)$，$\cdot$ 为服务组合的顺序运算符。其中 $cpS_1=(P_1,T_1,SubS_1,IP_1,OP_1,M_{10})$，$cpS_2=(P_2,T_2,SubS_2,IP_2,OP_2,M_{20})$，以下类同，如图 3(a)所示。

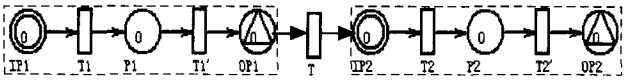

图 3(a) 指控能力包服务的顺序运算

- $P=P_1\cup P_2$；$T=T_1\cup T_2\cup\{t\}$；
- $SubS=SubS_1\cup SubS_2\cup\{cpS_1,cpS_2\}$；
- $IP=IP_1$；$OP=OP_2$；
- $M_0=M_{10}\cup M_{20}$。

定义 5(并发运算)

$cpS=cpS_1||cpS_2=(P,T,SubS,IP,OP,M_0)$，$||$ 为服务组合的并发运算符，如图 3(b)所示。

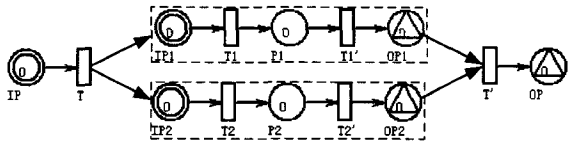

图 3(b) 指控能力包服务的并发运算

- $P=P_1\cup P_2$；$T=T_1\cup T_2\cup\{t,t'\}$；
- $SubS=SubS_1\cup SubS_2\cup\{cpS_1,cpS_2\}$；
- $IP=IP_1\cup IP_2$；$OP=OP_1\cup OP_2$；
- $M_0=M_{10}\cup M_{20}\cup IP\cup OP$。

定义 6(选择运算)

$cpS=cpS_1\oplus cpS_2=(P,T,SubS,IP,OP,M_0)$，$\oplus$ 为服务组合的顺序运算符，如图 3(c)所示。

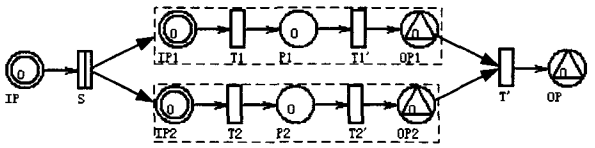

图 3(c) 指控能力包服务的选择运算

- $P=P_1\cup P_2$；$T=T_1\cup T_2\cup\{s,t'\}$；
- $SubS=SubS_1\cup SubS_2\cup\{cpS_1,cpS_2\}$；
- $IP=IP_1\cup IP_2$；$OP=OP_1\cup OP_2$；
- $M_0=M_{10}\cup M_{20}\cup IP\cup OP$。

S 是开关，当转移或开关点火时，除了执行点火函数外，转移和开关的区别在于，转移自动生成和传递发往后继的令牌，而开关没有自动传递机制，这使得使用开关时可以用脚本语言有选择地输出令牌到指定的位置上，实现多重选择功能。同时，OPN 描述能力的加强，也有效降低了服务组合模型的复杂度。

开关 S 脚本语言中动作函数的伪代码表示为：

Algorithm 1 SelSwitchActFun(IP)

```
Require: IP≠ϕ
1. if IP1≠ϕ
2.    do match IP1 with restriction of IP
3.    IP1←Token
4. else if IP2≠ϕ
5.    do match IP2 with restriction of IP
6.    IP2←Token
7. return Token
```

定义 7(随机顺序)

$cpS=cpS_1\Theta cpS_2=(P,T,SubS,IP,OP,M_0)$，$\Theta$ 为服务组合的随机顺序运算符。其它几种基本服务组合模型的现实意义都容易理解，随机顺序的现实意义是：军事应用中对于资源的利用效率要求更高，如果完成一项任务有两项服务缺一不可，但是又对时限要求不高，采用并行服务又太浪费资源，此时就可以采用随机顺序的服务模型，这种情况是很普遍的，所以也将之列为基本服务组合模型，如图 3(d)所示。

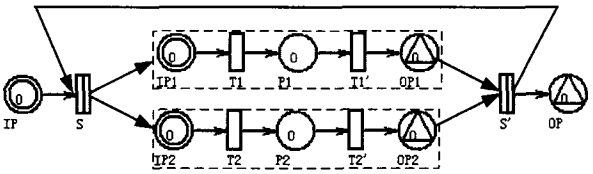

图 3(d) 指控能力包服务的随机顺序运算

- $P=P_1\cup P_2$；$T=T_1\cup T_2\cup\{s,s'\}$；
- $SubS=SubS_1\cup SubS_2\cup\{cpS_1,cpS_2\}$；
- $IP=IP_1\cup IP_2$；$OP=OP_1\cup OP_2$；
- $M_0=M_{10}\cup M_{20}\cup IP\cup OP$。

开关 S，S' 脚本语言中动作函数的伪代码表示为：

Algorithm 2 SelSwitchActFun(IP)

```
Require: IP, IP1, IP2≠ϕ
1. switch s receive Token form IP
2. if (do match IP1 with restriction of IP)
3.    IP1←Token
4. else(do match IP2 with restriction of IP)
5.    IP2←Token
6. switch s' receive Token form OP1 or OP2
7.    switch s←Token
8.    (IP1 or IP2 which has not got Token)←Token
9. switch s' receive Token form OP1 or OP2
10.   OP←Token
11. return Token
```

定义 8(迭代运算)

$cpS=\mu cpS_1=(P,T,SubS,IP,OP,M_0)$，$\mu cpS_1$ 代表重复 μ 次 cpS_1，如图 3(e)所示。

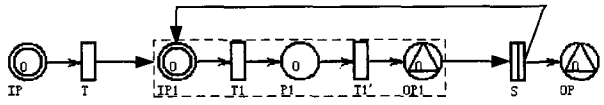

图 3(e) 指控能力包服务的迭代运算

- $P=\mu P_1$；$T=\mu T_1\cup\{t,\mu s\}$；
- $SubS=\mu SubS_1\cup\mu cpS_1$；
- $IP=IP_1$；$OP=OP_1$；
- $M_0=\mu M_{10}\cup IP\cup OP$。

开关 S 脚本语言中动作函数的伪代码表示为：

Algorithm 3 SelSwitchActFun(IP)

Require: IP≠ϕ

1. swith s receive Token form OP_1
2. for (i=0;i<μ,i++)
3. IP_1←Token
4. jump over from the above circulation and i=μ−1
5. OP←Token
6. return Token

本节建立了服务向 OPN 的映射关系，给出了 5 种基本服务组合模型，并研究了其约束关系，通过开关动作函数的伪代码表示，简化了服务组合模型的复杂度。

2.2 数学算子分析

在 2.1 节基本服务组合模型研究的基础上，可以通过将数学算子应用于基本组合模型，产生更复杂的服务组合方式；或者通过数学算子分析对复杂组合服务进行化简，从而产生易于理解和分析的基本组合服务模型。基本服务组合模型和数学算子的结合易于分析服务组合过程的归纳和回溯，便于研究服务组合的机理。下面在定理 1 的基础上给出服务组合模型的等价数学算子。

定理 1 对于组合服务 cpS,cpS'，如果 $IP=IP'$，并且 $OP=OP'$，那么 cpS,cpS' 等价，记为 $cpS=cpS'$。

证明(反证法)：如果 $IP=IP'$，并且 $OP=OP'$，但 cpS,cpS' 不等价，那么 cpS,cpS' 间功能上必然会有一个量差 θ，组合服务 cpS,cpS' 的对外输出端口 OP,OP' 也相应有一个量差 ξ，这与命题中 $OP=OP'$ 相矛盾，故 $cpS=cpS'$。证毕。

对于服务 cpS_1,cpS_2,cpS_3，服务组合模型具有交换性、结合性和自反性等数学性质，如表 2 所列。

表 2 指控能力包服务组合模型的数学算子

$(cpS_1\cdot cpS_2)\cdot cpS_3=cpS_1\cdot(cpS_2\cdot cpS_3)$	(1)
$(cpS_1\|\|cpS_2)\|\|cpS_3=cpS_1\|\|(cpS_2\|\|cpS_3)$	(2)
$(cpS_1\oplus cpS_2)\oplus cpS_3=cpS_1\oplus(cpS_2\oplus cpS_3)$	(3)
$cpS_1\|\|cpS_2=cpS_2\|\|cpS_1$	(4)
$cpS_1\oplus cpS_1=cpS_1$	(5)
$cpS_1\oplus cpS_2=cpS_2\oplus cpS_1$	(6)
$cpS_1\|\|cpS_1=cpS_1$	(7)
$(cpS_1\|\|cpS_2)\cdot cpS_3=(cpS_1\|\|cpS_3)\cdot(cpS_2\|\|cpS_3)$	(8)
$(cpS_1\oplus cpS_2)\cdot cpS_3=(cpS_1\cdot cpS_3)\oplus(cpS_2\cdot cpS_3)$	(9)
$(cpS_1\oplus cpS_2)\|\|cpS_3=(cpS_1\|\|cpS_3)\oplus(cpS_2\|\|cpS_3)$	(10)
$cpS_1\Theta cpS_2=cpS_2\Theta cpS_1$	(11)
$cpS_1\Theta cpS_1=cpS_1\cdot cpS_1$	(12)
$cpS_1\Theta cpS_2=(cpS_1\cdot cpS_3)\oplus(cpS_2\cdot cpS_3)$	(13)

算子(1)证明：对于三个服务 cpS_1,cpS_2,cpS_3，$cpS=(cpS_1\cdot cpS_2)\cdot cpS_3$，$cpS'=cpS_1\cdot(cpS_2\cdot cpS_3)$，左边的输入均为 IP_1，输出均为顺序组合后的输出 OP_3，由定理 1 可知，$cpS=cpS'$。同理，可证明其余算子。

图 4 为算子(5)中两个等价的服务组合，通过对组合模型和数学算子的扩展，可以得到更强的服务组合工具，也为复杂服务的分解与研究提供了方法。在组合模型和数学算子的支持下，下节将研究服务动态组合的流程。

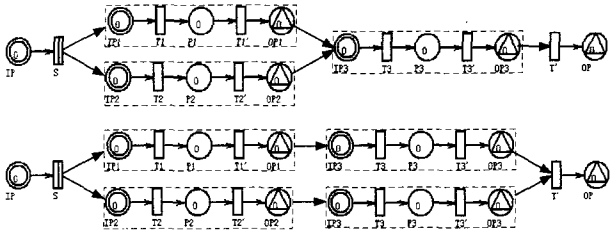

图 4 算子(5)中两个等价的服务组合

2.3 服务动态组合流程

服务组合可以是静态的——服务间通过预先定义好的方式交互，也可以是动态的——动态地发现服务并进行交互[12]。静态的服务组合方式只能满足日常指控系统运行的基本要求，BPEL4WS 和 WSCI 支持静态服务组合的开发，对于战场环境复杂多变的军事应用而言，实时动态的服务组合的研究则更有意义。动态的服务组合要尽量减少人员的参与，提高自动化组合的程度，常用的动态服务组合方法有 eFlow，ICARIS，SWORD，SHOP2 等[13]。

文献[14,15]构建了动态服务组合的框架，并把动态服务组合的过程定义为：定义服务通用接口，根据用户需求，选择所需服务，产生服务目录，以便定位服务的位置，最后依照组合服务框架动态构建组合服务。

基于 OPN 的指控能力包服务 cpS 由精确的六元组描述，清晰地定义了服务的输入输出接口 IP,OP，完全满足文献[16]中对松散接口的定义，实现了对服务的统一刻画，对于服务的匹配优化选择已经有很多研究工作，本文就不再扩展。在确定了所需的服务集 cpS_{set} 之后，通过语义分析，确定服务组合模型的方式和所需的数学算子，就能输出具有通用接口 IP,OP 的组合服务 cpS，从而只做很少的工作，就能实现对动态组合服务的支持，基于 OPN 的动态组合流程如图 5 所示。

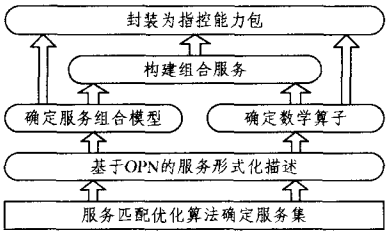

图 5 基于 OPN 的服务动态组合流程图

3 实验仿真与分析

3.1 OPMSE 中态势信息组合服务建模

实验仿真使用国防科大 OPMSE 对象 Petri 网建模仿真环境，以态势能力包构建中的态势信息订制、分发和显示 3 个服务为例，其中涉及到两个对象：信息中心和作战节点。作战节点发出信息订制请求服务 cpS_1，信息中心接收请求后提供信息分发服务 cpS_2，cpS_1 和 cpS_2 两个服务是顺序关系，作战

节点接收到态势信息后响应信息显示服务 cpS_3，为了保证态势信息的实时性，此时的态势信息显示是与订制和分发并行的。所以态势信息组合服务 $cpS=(cpS_1 \cdot cpS_2)||cpS_3$，如图 6 所示。其中各控制变迁功能分别是：

$T1$ 为订制态势信息成功时传送执行结果。

$T2$ 为态势信息分发成功时传送执行结果。

$T3$ 为态势信息显示成功时传送执行结果。

$T1$ 的动作函数 VBScript 代码如下，通过它可以控制令牌的点火时间和传递属性信息：

```
Set token=Object. GetTokenOnARc("A13",0)
  dz=token. GetTokenProperty("订制请求"). GetValue
  Object. SetProperty "请求输出",dz
```

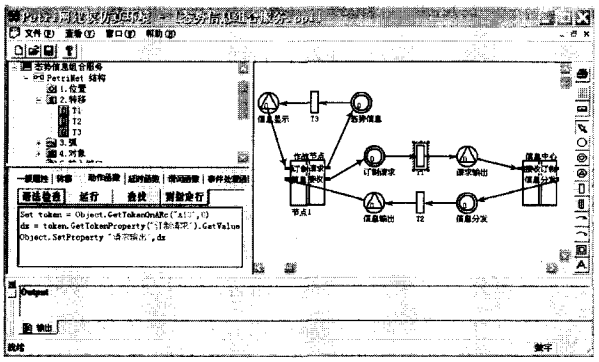

图 6　OPMSE 中态势信息组合服务建模

在 OPMSE 下，可以模拟组合服务的调用执行过程，可清晰地表达组合服务的组成逻辑，并可通过 OPN 的层次化设计方法实现服务的多级嵌套。通过使对象、位置、转移等元素携带属性信息或设置动作函数，从而可以实现丰富的语义功能。在对象 Petri 网仿真运行器 OPDL-Runner 的支持下，可以实现仿真模型的连续或单步运行，并输出事件控制脚本函数、仿真报告结果等，为服务组合的分析提供支持，从而可以在仿真阶段就发现制约效能的瓶颈，并有针对性地加以解决，以减少服务组合实际部署时才发现问题所引起的巨大成本消耗。

3.2　基于 OPN 的服务动态组合性能分析

服务组合性能分析涉及广泛，由于篇幅限制，在此只研究在服务数量增加的情况下，基于 OPN 的服务随机顺序组合方式的平均执行时间和平均信息量。实验计算机配置为主频 C2.4 G，内存 768M，仿真基于以下假设。

假设 1　假设所有服务均为不可分原子服务，服务组合采用随机顺序组合模型，不考虑服务匹配算法，也即是所有服务均为候选原子服务。单个原子服务执行时间为 5s，信息量为 10kByte[17]。

如表 3 和图 7 所示，从仿真数据可以看出，在随机顺序组合模式下，随着服务数量的增加，平均执行时间和平均信息量逐渐增加，但上升速率则远低于服务数量增加的速率，基于 OPN 的服务描述和动态组合能实现资源的较小开销和较高的性能，同时，通过仿真实验，也可以发现制约大型服务组合性能的瓶颈，从而可以针对性地，如可以通过提高硬件设施水平和增加信道容量等方式，提高军事背景中大规模服务组合的综合性能。

表 3　随着服务数量的增加随机顺序运算仿真数据

编号	服务数量	平均执行时间时间/S	平均信息量/(kByte/s)
1	5	5.062	2.027
2	10	5.527	2.803
3	20	5.986	3.677
4	35	7.519	4.569
5	50	8.186	6.380

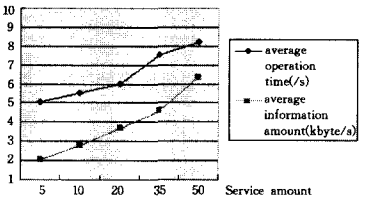

图 7　随机顺序服务组合仿真数据拟合图

结束语　基于 OPN 的服务组合模型清晰地定义了服务的输入输出接口，开关和脚本支持等 OPN 特有的描述方式大大加强了对服务的描述能力，同时也降低了服务组合模型的复杂性；在基本服务组合模型的基础上，通过对数学算子的扩展，可以构建更复杂的组合服务；同时，基于组合模型和数学算子的描述能力，也支持动态的服务组合。今后将继续研究指控能力包服务的匹配和组合问题，改进服务匹配的优化算法和提升服务组合的效率，进一步关注原型系统的设计和实现。

参 考 文 献

[1] 饶元，冯博琴，李尊朝. 基于 Web Services 的服务合成技术研究综述[J]. 系统工程与电子技术，2005，27(8)：1481-1489

[2] 郭玉彬，杜正越，奚建清. Web 服务组合的有色网模型及运算性质[J]. 计算机学报，2006，29(7)：1067-1075

[3] Korhonen J，Pajunen L，Puustjarvi J. Automatic composition of Web service workflows using a semantic agent[C]// Proceedings of the IEEE/WIC International Conference on Web Intelligence (W I 2003). Halifax，Canada，2003：566-569

[4] Sirin E，Hendler J，Parsia B. Semi-automatic composition of Web services using semantic description[C]// Web Services：Modeling，Architecture and Infrastructure workshop in conjunction with ICEIS2003. 2003

[5] Sycara K，Paolucci M，Ankolekar A，et al. Automated discovery，interaction and composition of semantic Web service[J]. Journal of Web Semantics，December 2003，1(1)

[6] 李曼，王大治，杜小勇，等. 基于领域本体的 Web 服务动态组合[J]. 计算机学报，2005，28(4)：644-650

[7] Hamadi R，Benatallah B. A Petri net based model for Web service composition[C]// Proceedings of the 14th Australian Database Conference on Database Technologies. Adelaide，South Australia，2003：191-200

[8] 钱柱中，陆桑璐，谢立. 基于 Petri 网的 Web 服务自动组合研究[J]. 计算机学报，2006，29(7)：1057-1066

[9] 李景霞，侯紫峰. 基于颜色 Petri 网的 Web 服务组合建模及应用[J]. 计算机应用研究，2006(9)：149-154

[10] 柏晓莉，余滨，黄光奇. 基于对象 Petri 网的计算机网络仿真建模[J]. 计算机仿真，2003，20(9)：45-47

[11] 陈彦萍，李增智，唐亚哲，等. 一种满足马尔可夫性质的不完全信息下的 Web 服务组合方法[J]. 计算机学报，2006，29(7)：1076-1083

[12] Srivastava B，Koehler J. Web Service Composition-Current Solutions and Open Problems[C]// ICAPS 2003 Workshop on Plannning for Web Services. 2003：28-35

[13] Fujii K，Suda T. Semantics-Based Dynamic Service Composition [J]. IEEE Journal on selected areas in Communications，2005，23 (12)

[14] Mennie D, Pagurek B. An Architecture to Support Dynamic Composition of Service Components[C]// the Fifth International Worshop on Component-Oriented Programming-WCOP 2000. held in conjunction with ECOOP 2000, Sophia Antipolis, France, June 2000

[15] Tosic V, Mennie D, Pagurek B. On Dynamic Service Composition and Its Applicability to E-Business Software Systems[C]// WOOBS'01(Workshop on Object-Oriented Business Solutions) workshop(at ECOOP 2001). Budapest, Hungary, June 2001

[16] Fujii K, Suda T. Loose Interface: An Extended Interface Definition for Dynamic Service Composition[C]// Proc. of the First Annual Symposium on Autonomous Intelligent Netwrks and Systems. Los Angeles, CA, May 2002

[17] Wang jiacun, Deng Yi, Xu Guang. Reachability analysis of real-time system using Petri nets[J]. IEEE Trans. on system, Man and Cybernetics, 2000, 30(5)

(上接第 35 页)

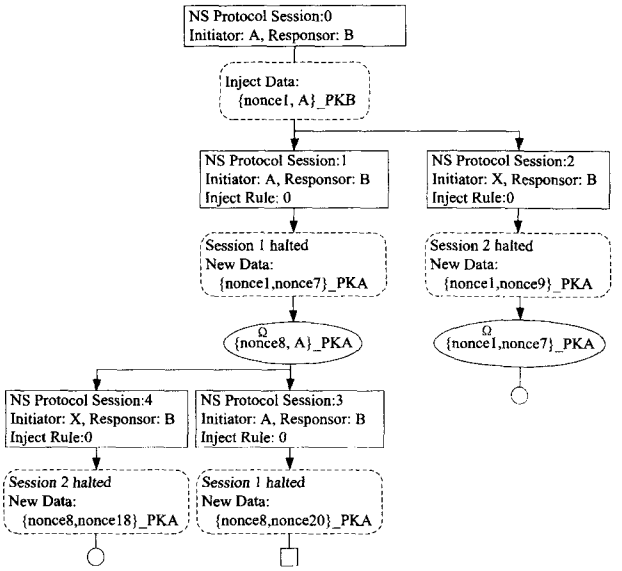

图 1 无攻击者 X 参与的会话所生成的数据形成的注入求解序列

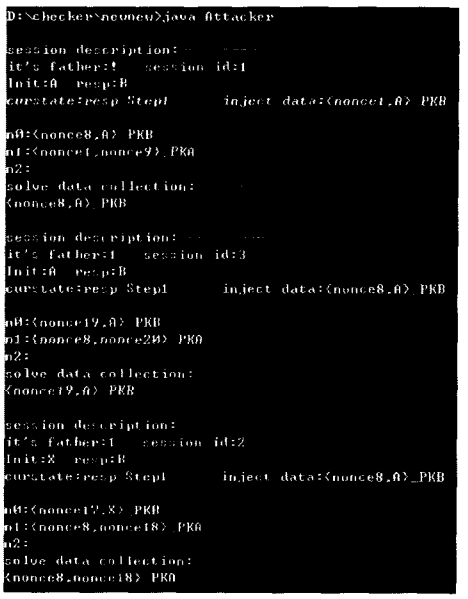

图 2 实验结果

结束语 本文分析了注入攻击是安全协议攻击者实现其攻击目标的必要手段，并基于注入攻击和注入攻击序列的性质，提出了一种基于攻击序列搜索的安全协议验证算法。本文分析了攻击序列的性质，并证明了对于规则安全协议的验证算法的可终止性。

本文的算法与其它同类方法，如模型检测、理论证明等方法相比有以下优点：

基于本方法实现的自动化检验工具对于规则安全协议一定会终止并给出确定结果。其它同类工具，如 OFMC[7] 工具虽然能在发现漏洞时终止，但是用户很难判断其运行状态，不知道是协议没有漏洞还是需要继续等待下去。而其它方法经常会出现无解的情况和不能终止的情况。因此本算法与之相比，其终止性更具实用性。

基于本方法实现的自动化检验工具能给出具体的攻击方法。与理论证明方法不同的是，它们虽然能证明可能存在的安全漏洞，却未必能给出具体攻击方法，如 SPI[3,4]，BAN[2] 等。

基于本方法便于实现并行检验，因为本方法是针对安全协议的每个可实施注入攻击的规则进行的，因此可将检验工作拆分成多个子任务进行验证。

本文算法的缺点在于目前方法只适用于规则安全协议。下一步的工作将对算法进行扩展，使其能实现更多类型的安全协议的验证。

参 考 文 献

[1] Syverson P, Meadows C, Cerversato I. Dolev-Yao is no better than Machiavelli[C]// Degano P, ed. Proceedings of the First Workshop on Issues in the Theory of Security. Geneva, Switzerland, 2000: 87-92

[2] Burrows M, Abadi M, Needham R. A logic of authentication[C]// Proceedings of the Twelfth ACM Symposium on Operating Systems Principles. 1989: 1-13

[3] Abadi M, Gordon A D. A calculus for cryptographic protocols: the spi calculus[C]// Graveman R. ed. Proceedings of the 4th ACM Conference on Computer and Communications Security. Zurich, 1997: 36-47

[4] Abadi M, Gordon A D. Reasoning about cryptographic protocols in the spi calculus[C]// Proceedings of the 8th International Conference on Concurrency Theory. 1997: 59-73

[5] Durgin N A, Lincoln P D, Mitchell J C, et al. Multiset rewriting and the complexity of bounded security protocols[J]. Journal of Computer Security, 2004

[6] Javier F, Herzog J C, Guttman J D. Strand Spaces: Proving Security Protocols Correct[J]. Journal of Computer Security, 1999, 7 (2/3): 191-230

[7] Abadi M, Blanchet B. Secrecy Types for Asymmetric Communication[C]// Honsell F, Miculan M, eds. Proceedings of the 4th International Conference on Foundations of Software Science and Computation Structures. London, UK: Springer-Verlag, 2001: 25-41

[8] Basin D, Modersheim S, Vigano L. An On-the-Fly Model-Checker for Security Protocol Analysis[C]// Proc. Eighth European Symp. on Research in Computer Security. 2003: 253-270

[9] Lowe G. An attack on the Needham-Schroeder public key authentication protocol[J]. Information Processing Etters, 1995 (3): 131-136