

面向语义 Web 服务组合的模糊 Petri 网推理算法

葛敬军 黄华 胡建明

(南昌航空大学计算机学院 南昌 330063)

摘要 就语义 Web 服务自动组合技术而言,在服务组合时需要建立形式化的描述模型,以及有效利用这些已有的数据依赖关系实现服务合成的请求。而现有的组合方法与实施框架可能出现全局状态的指数性增长。针对这一问题,提出一种适合于 Web 环境的模糊 Petri 网逆向推理算法。该算法结合模糊 Petri 网的逆向推理和代数运算的优点,能有效减小算法空间复杂度,把一个大的、复杂的系统转化为一个只与问题相关的、小的系统来处理。最后,通过一个例子说明了算法的可用性。

关键词 Web 服务组合,模糊 Petri 网,逆向推理,关联矩阵

中图分类号 TP393 文献标识码 A

Backward Reasoning Algorithm of Fuzzy Petri Nets for Semantic Web Services Composition

GE Jing-jun HUANG Hua HU Jian-ming

(School of Computer, Nanchang Hangkong University, Nanchang 330063, China)

Abstract As for the technology of semantic Web services automatic composition, the key which solves this problem is to construct a model of the formal description of Web services and realize the Web service composition request by using this dependency relationship of available data. However, current approaches for service composition may return back lots of global state with exponential growth. To address this issue, backward reasoning algorithm of fuzzy petri nets for semantic Web services automatic composition with correctness guarantees was proposed. The algorithm fully takes advantage of mathematics foundation of petri nets, a complex system can be transformed into a simpler system closely related to the current problems. Thus, the space complexity of the algorithm can be reduced. Finally, an example was used to illustrate the applicability of this approach.

Keywords Web services composition, Fuzzy petri nets, Backward reasoning, Relational matrix

因特网创始人 Tim Berners-Lee 等发表在《Scientific American》杂志上的著名文章 "The Semantic Web" (语义 Web, 或语义万维网) 中指出^[1,2], 语义 Web 是当前万维网的扩展和延伸。语义 Web 的信息具有充分的、完备的语义定义, 它的主要任务是使数据能被计算机自动地处理和理解, 其最终目标是让计算机可以在这些海量信息中找到所需要的任何信息, 从而将万维网中现存的信息发展成一个巨大的全球信息库。要实现语义 Web, 使计算机能够处理信息, 就要依赖于将内容与形式化的意思表示对应起来。这样, 摆在语义 Web 面前的挑战就是提供一种语言能够同时描述数据以及根据数据进行推理的规则, 并且允许任何现存知识表示系统中的规则都能输出到 Web 上。语义服务通过引进语义来帮助消除服务发现和按需组合等过程中的二义性和模糊性, 从而为服务自动发现、组合、执行等提供良好的基础^[3-5]。

本文采用模糊 Petri 网形式化模型对语义 Web 服务组合建模, 将可用的服务用一组推理规则表示。将用户提供的输入表示为推理中的一组事实, 用户期待的输出表示为推理中的一个目标, 从而将寻找一个满足用户输入输出要求的组合

服务问题转化为一个模糊 Petri 网逆向推理问题。使用 T-不变量技术来确定是否存在一个满足用户输入输出要求的合成服务。最后, 通过实例说明代数模糊 Petri 网逆向推理算法可以获得一个满足用户诸项要求和约束输入输出要求、行为约束兼容要求的合成服务的有用性。

1 基本概念

本文将基于代数的模糊 Petri 网逆向推理来解决 Web 服务组合问题。因此, 首先介绍一些模糊 Petri 网的相关概念。

根据文献[6,7]的定义, 一个 FPN 被定义为八元组:

$$FPN = (P, T, D, I, O, f, \alpha, \beta)$$

其中, $P = \{p_1, p_2, \dots, p_n\}$ 是一个有限库所集合; $T = \{t_1, t_2, \dots, t_m\}$ 是一个有限变迁集合; $D = \{d_1, d_2, \dots, d_n\}$ 是一个有限命题集合; $|P| = |D|$; $I: T \rightarrow P^\infty$ 是一个输入函数, 映射一个变迁到它的输入库所集合; $O: T \rightarrow P^\infty$ 是一个输出函数, 映射一个变迁到它的输出库所集合; $f: T \rightarrow [0, 1]$ 是一个函数, 映射变迁到一个 0~1 的数值, 用来表示变迁对应的推理规则置信度 (Confidence Factor) S ; $\alpha: P \rightarrow [0, 1]$ 是一个函数, 映射库

到稿日期: 2008-10-14 返修日期: 2008-12-30 本文受国家自然科学基金项目(60673055), 江西省自然科学基金项目(0611094)资助。

葛敬军(1976-), 男, 博士研究生, 讲师, 主要研究方向为语义 Web 服务、Petri 网理论与应用, E-mail: gejingjun2000@126.com; 黄华(1973-), 女, 副教授。

所到一个0~1的数值,用来表示该库所对应的命题成立的真实度; $\beta: P \rightarrow D$ 是一个函数,映射库所对应的命题。

当用 FPN 进行模糊推理时,一个库所表示一个命题,一个变迁表示一条模糊推理规则,即两个命题之间的因果关系。托肯值代表命题的真实度。每个变迁有一个置信度,表示推理规则的可信度。

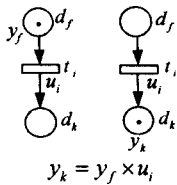
FPN 的基本推理规则的形式是:

$R_i: IF d_f THEN d_k (CF = u_i)$, 这里 d_f, d_k 是两个命题。推理过程可以建模为图 1, 其中命题 d_f, d_k 用库所 p_f 和 p_k 表示, 命题 d_f 的真实度为 y_f 。命题之间的因果关系用变迁 t_i 表示, 它的置信度为 u_i 。当变迁触发后, 命题 p_k 的真实度为 $y_f \times u_i$ 。一条推理规则若包含“AND”或者“OR”连接符, 就称为组合产生式规则。根据文献[6], 组合产生式规则被分为基本的 3 类:

类型 1: $IF d_{f1} AND d_{f2} AND \dots AND d_{fn} THEN d_k (CF = u_i)$, 这里 $d_k (1 \leq k \leq n) \in D$ 。推理过程可以建模为图 2, 推理后命题 d_k 的真实度是 $\min(d_{f1}, d_{f2}, \dots, d_{fn}) \times u_i$ 。

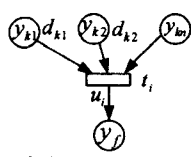
类型 2: $IF d_f THEN d_{k1} AND d_{k2} AND \dots AND d_{kn} (CF = u_i)$, 这里 $d_k (1 \leq k \leq n) \in D$ 。推理过程可以建模为图 3。

类型 3: $IF d_{f1} OR d_{f2} OR \dots OR d_{fn} THEN d_k (CF = u_i)$, 这里 $d_k (1 \leq k \leq n) \in D$ 。推理过程可以建模为图 4。



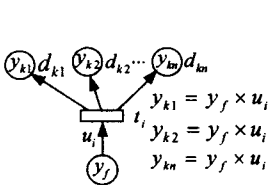
$$y_k = y_f \times u_i$$

图 1 基本 FPN 推理过程



$$y_k = \min(y_{f1}, y_{f2}, \dots, y_{fn}) \times u_i$$

图 2 类型 1 的 FPN 推理过程

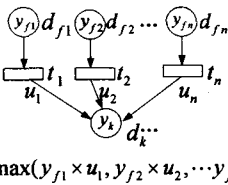


$$y_{k1} = y_f \times u_i$$

$$y_{k2} = y_f \times u_i$$

$$y_{kn} = y_f \times u_i$$

图 3 类型 2 的 FPN 推理过程



$$y_k = \max(y_{f1} \times u_i, y_{f2} \times u_i, \dots, y_{fn} \times u_i)$$

图 4 类型 3 的 FPN 推理过程

在以上各种类型的 FPN 推理规则中, 可用服务规则和参数组合性规则可以通过扫描服务注册中心以及本体推理得到, 可以一次建立并存储在规则库中, 供反复使用。用户请求规则是在用户有服务请求到达的时候建立, 可以临时存储在工作区中。

2 逆向模糊推理

为了表达推理过程, 参考文献[8, 9]中的方法, 结合逆向推理算法的需要, 定义下面 4 个运算符:

$\oplus: X \oplus Y = Z$, X, Y 和 Z 都是 $m \times n$ 维矩阵, $z_{ij} = \max(x_{ij}, y_{ij})$;

$\otimes: X \otimes Y = Z$, X, Y 和 Z 分别是 $m \times s$ 维、 $s \times n$ 维和 $m \times n$ 维矩阵, $z_{ij} = \max(x_{ik} \times y_{kj}), 1 \leq k \leq s$;

$\odot: X \odot Y = Z$, X, Y 和 Z 都是 $m \times n$ 维矩阵, $z_{ij} = \min(x_{ij}, y_{ij})$;

$\odot_n: X \odot_n = Z$, X, Z 是 $m \times n$ 维矩阵, n 是一个数字, $z_{ij} = 1$ If $x_{ij} \times n; z_{ij} = 0$ If $x_{ij} \times n \neq 1$ 。

逆向推理算法由 3 阶段完成: 首先将 FPN 结构用代数形式表示。第二阶段完成逆向推理, 其功能有二: 一是通过逆向推理去掉所有与目标库所无关的库所和变迁, 化简 FPN 网结构; 二是完成第三阶段的推理计算, 这一步确定了变迁触发的顺序, 可以保证所有变迁的输入库所在变迁触发前都是有值的。第三阶段是计算目标命题真实度。经过第二步的逆向推理, 计算目标命题真实度的过程仅仅是按变迁触发顺序依次计算其输出库所的真实度的过程。最终, 得出目标命题的真实度。

Input: FPN 结构, 目标库所的名字。

Output: 目标库所的真实度。

第一步 将 FPN 结构用矩阵形式表示,

$\Delta = \{\delta_{ij}\}$ 为输入矩阵: $\delta_{ij} = \begin{cases} 1, & \text{如果 } p_i \in I(t_j) \\ 0, & \text{如果 } p_i \notin I(t_j) \end{cases}$

$\Gamma = \{r_{ij}\}$ 为输出矩阵: $r_{ij} = \begin{cases} 1, & \text{如果 } p_i \in O(t_j) \\ 0, & \text{如果 } p_i \notin O(t_j) \end{cases}$

向量 S_0 为命题的初始真实度,

$$S_0 = [a(p_1) \ a(p_2) \ \dots \ a(p_n)]^T$$

向量 G 表示需要计算的目标库所:

$$G = \{x_i\} x_i = \begin{cases} 1, & \text{如果 } p_i \text{ 是结果命题, } p_i \in P \\ 0, & \text{否则} \end{cases}$$

这里只能指定一个命题为结果命题, 也就是说向量 G 中只能有一个 1。

向量 U 表示变迁的置信度向量:

$$U = [f(t_1) \ f(t_2) \ \dots \ f(t_n)]^T$$

第二步 逆向推理, 简化输入输出矩阵。

$T' = \Gamma^T \otimes G$; $/ * 获得目标库所的输入变迁 * /$

$z = 0; T_x = T'; P_x = G / * 初始化 * /$

$T = (O_1, O_2, O_3, \dots, O_n)^T$;

While($T! = T'$) $\{z = z + 1; T = T'; P' = \Delta \otimes T; P_x = P' \times z + P_x; T' = T' \otimes P'; T_x = T_x + T';\} z = z - 1$;

去掉 Δ 和 Γ 中的 T_x, P_x 对应的行列中为零的项, 得到两个新的矩阵 Δ', T' ; 去掉 S_0 对应 P_x 中为零的项, 得到向量 Q_0 ; 去掉 P_x 中为零的项, 得到向量 P_x' ; 去掉 U 中对应 T_x 为零的项, 得到新向量 U' ;

第三步 计算目标库所的真实度。

M 为一个大小为 $|U'|$ 的单位矩阵。

for ($k = 0; k < z; k++$) $\{Q_{k+1} = Q_k \oplus [(\Gamma' \times M \times U') \otimes (\Delta_x \otimes \frac{1}{(z-k)} \odot Q_k)]\}$

最终, 向量 Q_{k+1} 中对应的目标库所的值是目标库所的真实度。可以看出, 该算法先将 FPN 结构转换成代数形式, 得到矩阵输入、输出矩阵 Δ, Γ 、初始真实度向量 S_0 、目标库所向量 G 、变迁的置信度向量 U 。经过第二步逆向推理, 得到与目标命题相关的矩阵 Δ', Γ' 、初始真实度向量 Q_0 、代表变迁触发顺序的向量 P_x' 、变迁的置信度向量 U' 以及需要变迁触发的次数 z 。其中 Δ_x 中的数字记录着变迁发生的顺序。第三步, 经过循环 z 次, 根据 Δ_x 中的记录的触发顺序, 可以计算出目标命题的真实度。

3 应用举例

下面给出一个简单的例子来具体说明语义 Web 服务自

动组合的逆向模糊推理算法。假定在下面的情景中有一个可用的服务和一个用户请求。每个服务接受输入,产生一定的输出。有一些行为约束值在该例中没有给出。可以看出,没有任何单一可用的服务可以满足该请求,要想满足用户的请求,只有进行服务组合。

为方便计算,这里假设所有的服务提供者和服务请求者使用同一个共享的个体。但在实际应用中,由于存在词汇的差异,即使同一个领域,服务提供者和请求者也可能使用不同的个体。在这种情况下,可以创建一个映射个体来捕获不同的个体所使用的概念之间的关系。通过上面的建模方法,建立一个包括7条模糊产生式推理规则的知识系统,它们是:

- R1: IF s1 THEN s3 ($\mu=0.9$)
- R2: IF s2 THEN s6 ($\mu=0.95$)
- R3: IF s3 THEN s7 ($\mu=0.95$)
- R4: IF s4 THEN s8 ($\mu=0.9$)
- R5: IF s5 THEN s9 ($\mu=0.9$)
- R6: IF s6 AND s7 THEN s9 ($\mu=0.85$)
- R7: IF s7 AND s8 THEN s10 ($\mu=0.95$)

命题 s1, s2, s3, s4 的真实度分别为 0.9, 0.8, 0.85, 0.6。计算命题 s9 的真实度,可以得到如图 5 所示的 FPN。

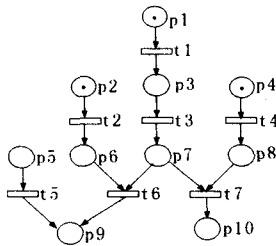


图 5 知识系统对应的 FPN

第一步 转换 Petri 网矩阵形式

$$\Delta = \begin{matrix} & t1 & t2 & t3 & t4 & t5 & t6 & t7 \\ \begin{matrix} p1 \\ p2 \\ p3 \\ p4 \\ p5 \\ p6 \\ p7 \\ p8 \\ p9 \\ p10 \end{matrix} & \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} & \begin{matrix} & t1 & t2 & t3 & t4 & t5 & t6 & t7 \\ \begin{matrix} p1 \\ p2 \\ p3 \\ p4 \\ p5 \\ p6 \\ p7 \\ p8 \\ p9 \\ p10 \end{matrix} & \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \end{matrix}$$

图 6 输入矩阵

图 7 输出矩阵

$$S_0 = [0.9 \ 0.8 \ 0 \ 0.85 \ 0.6 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$$

$$G = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0]^T$$

$$U = [0.9 \ 0.95 \ 0.95 \ 0.9 \ 0.9 \ 0.85 \ 0.95]^T$$

第二步 逆向推理,得到

$$P_x = [3 \ 2 \ 2 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0]^T$$

$$T_x = [1 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0]^T$$

记录下 T_x, P_x 中不为 0 的项所代表的变迁和库所:

$$T_r = [t1 \ t2 \ t3 \ t5 \ t6]$$

$$P_l = [p1 \ p2 \ p3 \ p5 \ p6 \ p7 \ p9]$$

根据上式,得到新的关联矩阵 Δ', Γ'

$$\Delta' = \begin{matrix} & t1 & t2 & t3 & t5 & t6 \\ \begin{matrix} p1 \\ p2 \\ p3 \\ p5 \\ p6 \\ p7 \\ p9 \end{matrix} & \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} & \begin{matrix} & t1 & t2 & t3 & t5 & t6 \\ \begin{matrix} p1 \\ p2 \\ p3 \\ p5 \\ p6 \\ p7 \\ p9 \end{matrix} & \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \end{matrix}$$

选择 S_0, P_x 中相应的行,得到 Q_0, P_x', P_x' :

$$Q_0 = [0.9 \ 0.8 \ 0 \ 0.6 \ 0 \ 0 \ 0]^T$$

$$P_x' = [3 \ 2 \ 2 \ 1 \ 1 \ 1 \ 1]^T$$

$$U' = [0.9 \ 0.95 \ 0.95 \ 0.9 \ 0.85]^T$$

$$\Delta_x = \Delta'^T \times (M \times P_x') = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

$z = 3;$

第三步 计算目标库所的真实度,经过 3 次循环得出

$$Q_1 = [0.9 \ 0.8 \ 0.81 \ 0.6 \ 0 \ 0 \ 0]^T$$

$$Q_2 = [0.9 \ 0.8 \ 0.81 \ 0.6 \ 0.76 \ 0.77 \ 0]^T$$

$$Q_3 = [0.9 \ 0.8 \ 0.81 \ 0.6 \ 0.7 \ 0.77 \ 0.76]^T$$

最终得出结论,目标命题 S9 的真实度是 0.76,说明存在一个潜在的满足用户输入输出请求的最优合成服务。对于本例中得到的 Petri 网的关联矩阵,就其物理含义而言,矩阵的列表示语义概念。如果行表示(经过分解后的)Web 服务,则矩阵元素表示该语义概念是否在该服务中出现(1 表示此语义概念在该服务的输出中出现,0 则表示此语义概念未在该服务中出现)。该关联矩阵是一种 0-1 稀疏矩阵,矩阵的每一行中大部分元素都是 0。针对具有这种特征的关联矩阵的 T-不变量求解, Borujij^[10] 等人给出了一种分治求解的方法。首先将一个大的关联矩阵分解成若干个小的子系统,然后针对每个子系统单独求解 T-不变量,最后汇聚这些局部不变量得到整个系统的 T-不变量。该方法适宜并行,对于求解服务组合这样大规模的问题非常合适。

结束语 随着 Web 应用越来越复杂多样,人们常常需要组合多个 Web 服务来完成一个较复杂的任务。而在语义 Web 环境下,计算机(程序或 Agent)可根据必要的信息自动、动态地组合服务。本文在分析多种模糊 Petri 网的推理算法的优缺点的基础上,提出一种新的基于模糊 Petri 网的逆向推理算法。该算法不仅可以使得数据结构简单,有利于计算机处理,而且采用逆向推理策略能有效减小算法空间复杂度,把一个复杂的系统转换为一个只与问题相关的小的系统,从而提高计算效率,在语义 Web 服务自动组合和知识推理的应用领域中具有很好的应用前景。

参考文献

[1] Berners-Lee T, Hendler J, Lassila O. The Semantic Web [J]. Scientific American, 2001, 284(5): 34
 [2] Zhao J F, Xie B, Zhang L, et al. A Web services composition method supporting domain feature [J]. Chinese Journal of Computers, 2005, 28(4): 731-738.

性和可靠性的情况下,简便计算如下:

不妨设 WS_1 和 WS_2 的费用权重为 0.7 和 0.5, 这与表 1 中的变权满足归一化条件。

将以上特征值和权重代入式(12)可得

$$\begin{aligned} \mu_5 &= \frac{0.44}{0.44+0.28 \times 1} + \frac{0.28 \times 1}{0.44+0.28 \times 1} j \\ &= 0.61+0.29j \\ \mu_{10} &= \frac{0.22}{0.22+0.5 \times 0.5} + \frac{0.5 \times 0.5}{0.22+0.5 \times 0.5} j \\ &= 0.47+0.53j \\ shi_w^{(5)}(H) &= \frac{0.61}{0.29} = 2.1 \\ shi_w^{(10)}(H) &= \frac{0.47}{0.53} = 0.9 \end{aligned} \quad (18)$$

可见,无论从特征加权联系度,还是从特征加权集对势来评价,其结果都是 WS_1 优于 WS_2 。

(2)特征常权集对分析方法

对于第 2 节例 2 中提到的问题,可以简单地使用特征常权集对分析方法处理。

评价者甲重视响应时间,他对于 Web 服务特征的偏好排序为响应时间→费用→可靠性→安全性,则不妨设相应的权重为 $w_b^{(1)}=0.4, w_c^{(1)}=0.3, w_k^{(1)}=0.2, w_s^{(1)}=0.1$ 。

评价者乙重视安全性,他对于 Web 服务特征的偏好排序为安全→可靠性→费用→响应时间,则不妨设相应的权重为 $w_b^{(2)}=0.4, w_k^{(2)}=0.3, w_c^{(2)}=0.2, w_s^{(2)}=0.1$ 。

经过无量纲、一致性极性、等数量级处理, WS_1 的特征值为 $v_b^{(1)}=1, v_s^{(1)}=0.8, v_k^{(1)}=1, v_c^{(1)}=1$; WS_2 的特征值为 $v_b^{(2)}=0.5, v_s^{(2)}=1, v_k^{(2)}=0.6, v_c^{(2)}=0.5$ 。

由式(12)得

甲对 WS_1 和 WS_2 评价(特征加权联系度)为

$$\mu_1(1) = \frac{2}{49} + \frac{47}{49}i, \mu_2(1) = \frac{2}{57} + \frac{55}{57}i$$

乙对 WS_1 和 WS_2 评价(特征加权联系度)为

$$\mu_1(2) = \frac{8}{23} + \frac{15}{23}i, \mu_2(2) = \frac{40}{73} + \frac{33}{73}i$$

可见,对于甲 WS_1 较优,对于乙 WS_2 较优。这个结论符合实际。

4.2 与联系数加权的比较

由第 2 节例 1 和例 2 的结论,经典集对分析无法分清 WS_1 和 WS_2 的优劣。

文献[13]提到的联系数加权方法,联系度由式(5)确定,其中 α, β, γ 由文献[15]确定,即需要专家对联系数进行半定

性、半定量的分析、评价,用区间评价值叠加成区间权重,再采取平均准则处理区间权重,最后归一化得到联系度权重。这种方法受专家影响太大,不适合 Web 服务的 QoS 评价。

特征加权集对分析中的联系度和集对势对于 Web 服务的 QoS 评价服务事物的客观规律,能更贴切地进行基于 QoS 评价的 Web 服务的比较、选取。

结束语 在对集对分析相关理论和可加权和评价研究的基础上,笔者提出了特征加权集对分析方法。它不同于目前的联系数加权,是对特征的联系数加权。经应用比较分析,特征加权集对分析方法比联系数加权集对分析方法在描述相关事物之间的联系方面更准确、合理。

参考文献

- [1] Zadeh L. A. Fuzzy sets [J]. Information and Control, 1965, 8: 338-356
- [2] DENG Julong. Control problem of grey systems [J]. System & Control Letters, 1982(3)
- [3] Pawlak Z. Rough sets [J]. International Journal of Computer and Information Sciences, 1982, 11: 341-356
- [4] 赵克勤. 集对分析对不确定性的描述和处理 [J]. 信息与控制, 1995, 24(3): 162-162
- [5] 赵克勤. 集对分析及其应用初探 [M]. 浙江: 浙江科技出版社, 2000
- [6] 赵克勤. 四元联系数的事态排序及应用 [M]. 集对分析与界壳论的研究与应用. 北京: 北京气象出版社, 2002
- [7] 赵克勤. 基于集对分析的对立分类、度量及应用 [J]. 科学技术与辩证法, 1994, 11(2): 26-30
- [8] 赵克勤. 集对分析中的不确定性理论 [J]. 大自然探索, 1995, 14(4): 87-88
- [9] 赵克勤. 集对论——一种新的不确定性理论方法与应用 [J]. 系统工程, 1996, 14(1): 18-23
- [10] 黄树林, 张江, 李华, 等. 模糊化的联系数及其应用初探 [M]. 模糊集理论与应用. 河北: 河北大学出版社, 1998
- [11] 朱军. 基于物元的集对分析预测 [C] // 第四届全国集对分析论文. 1998
- [12] 李光军, 贺仲雄. 基于 SPA 的模式识别 [J]. 通讯和计算机, 2007, 4(6)
- [13] 周泽南, 贺仲雄. 基于集对分析的决策支持系统及其应用 [J]. 通讯和计算机, 2006, 3(4)
- [14] 单鑫, 董文洪. 基于变权 SPA 的空中目标威胁综合评估方法 [J]. 现代防御技术, 2007, 35(5)
- [15] 闫皓, 贺仲雄. 权重分析系统 [J]. 系统工程与电子技术, 1992, 4: 41-45

(上接第 207 页)

- [3] Farrell J, Lausen H. Semantic annotations for WSDL and XML schema [EB/OL]. <http://www.w3.org/TR/sawSDL>, 2007
- [4] Akkiraju R, Sapkota B. Semantic annotations for WSDL and XML schema—Usage guide [EB/OL]. <http://www.w3.org/TR/sawSDL-guide/>, 2007
- [5] Cabrea F, Copeland G, Cox B. <http://www.ibm.com/developerworks/library/ws-transpec/>
- [6] Chen Shyi - Ming. Fuzzy backward reasoning using fuzzy Petri nets [J]. IEEE Transaction on System, Man, and Cybernetics-part B, 2000, 30(6): 846-856
- [7] 乐晓波, 葛敬军. 自动制造系统设计和仿真中的死锁控制 [J]. 计

算机工程, 2004, 31(4): 192-194

- [8] GAO Mei-mei, WU Zhi-ming. A fuzzy reasoning Petri net model and its reasoning algorithm [J]. Journal of Shanghai Jiao tong University, 1999, E-4(2): 5-9
- [9] Men Peng, Duan Zhang - hua. Backward Reasoning Algorithm for Petri Nets Based on Algebra [J]. Journal of System Simulation, 2007, 19: 161-163
- [10] Borujij A, Boutayeb M, Cecchin T. A decentralized approach for computing invariants in large scale and interconnected Petri nets [C] // Proc. of the 1997 IEEE Int'l Conf. on System, Man and Cybernetics. Orlando: IEEE Computer Society, 1997: 1741-1746