

一种全信息描述框架下的 Web 服务发现算法

皇甫先鹏 魏巍 陈洪辉

(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 Web 服务技术作为分布式、松耦合的网络系统实现技术,得到了长足的发展,但是服务失效和 UDDI 注册中心不提供服务选择和优化,查询的服务不能有效满足用户需求等问题却仍然十分突出。提出了基于全信息描述框架的 Web 服务发现算法的解决方案,有效地解决了以上问题。通过对 Web 服务的形式化描述、服务选择算法和服务优化排序算法等方面进行研究,并结合实验分析,验证了优化算法能有效提高 Web 服务发现的查全率和查准率,为提高大规模网络环境下 Web 服务发现查找的效率打下了坚实的基础。

关键词 全信息描述框架, Web 服务, 发现算法, 改进层次分析法

中图分类号 TP393 **文献标识码** A

Research on Web Service Discovering Algorithm Based on Information Description Framework

HUANGFU Xian-peng WEI Wei CHEN Hong-hui

(C4ISR Technology National Defense Science and Technology Key Lab, National University of Defense Technology, Changsha 410073, China)

Abstract Web service has been greatly developed as a distributed, loose-coupled networked system technology. But the problems of service invalidation, UDDI not supporting service selection and optimization, and services found cannot efficiently satisfying the demand of user requirement still haven't been solved. The paper brings forward Web service discovering algorithm based on information description framework. Through the research of Web service formalized description, service selection algorithm and service optimization sort algorithm, and with the experiment and analysis, the research validates optimized algorithm efficiently improves recall and precision rate of service discovering, and strikes a stable base for the improvement of Web service discovering efficiency.

Keywords Information description framework, Web service, Discovering algorithm, AHP

1 引言

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,Web 服务是 SOA 架构下的一种具体实现方式,作为分布式、松耦合的网络系统实现技术,已经得到了长足的发展。

Kim 等人在文献[1]中研究了 2003~2004 年公共 Web 服务的使用情况,其数量并没有明显地增加,只有大约 34% 的服务可用,而且每周大约有 16% 的已注册可用的 Web 服务失效;此外,UDDI 注册中心不提供对服务的筛选和优化排序,也使得用户对服务的查全查准率不高。针对以上问题,本文基于全信息描述框架,给出了一种高效的 Web 服务发现算法,并结合实验分析,对比随机算法验证了该算法能有效提高服务的查全和查准率。

2 Web 服务全信息形式化描述框架

2.1 Web 服务的全信息描述框架

钟义信于 1984 年提出的“全信息理论”认为:要认识一个事物,要描述一个系统,唯一的办法就是通过各种可能的途径来获得关于该事物、该系统的信息,即获得关于该事物的内部

结构的状态及其变化方式和外部联系的状态及其变化方式,或即获得关于该事物运动的状态及其变化方式。

用户使用 Web 服务时,会考虑 4 个基本问题: what it does, how well it does, how it works, how to work with it. 这 4 个问题涵盖了用户使用服务全过程所需要的信息,这些信息可以分为 3 个层次——Web 服务的语用信息、服务的语义信息和服务的语法信息,如图 1 所示。



图 1 服务全信息的 3 个层次

参考文献[2],遴选出对服务发现模型有影响的参数信息,给出如下定义。

定义 1 Web 服务的语用信息 (Pragmatic Information, Pr I)

Pr I是服务对服务使用者的有用性信息,即效用信息。服务的有用性取决于用户对服务的需求状况,也就是由用户的需求状态和关注点与所提供的服务之间的相关性所决定。Pr I是以下参数的集合。

$$Pr I = \{respTime, satiDegree, secLv, authLv, confLv, Score, \dots\}$$

其中,respTime 为服务响应时间,satiDegree 为使用满意度,secLv 为服务安全等级,authLv 为服务(信息/控制力)权威等级,confLv 为服务密级,Score 为服务自发布以来被查询的量值。

定义 2 Web 服务的语义信息(Semantic Information, Sel)

Sel 是有关服务意义的信息,其核心是服务的功能信息。服务的意义体现在其提供的信息或者操作能够满足用户的某种需要。服务功能的信息就包含在输入输出信息的变换和执行的操作之中。基于语义的算法描述研究最为广泛,这里就不再展开。

定义 3 Web 服务的语法信息(Syntax Information, SyI)

SyI 是用户如何调用服务的信息,是服务接口、消息格式、传输协议、访问位置等信息的集合。服务的语法信息与具体的实现平台和技术环境紧密相关,例如 WSDL 可作为 SyI 的一个子集。

$SPSet = \{S_1, \dots, S_k\}, k \geq 0$, 为服务遵循或者采用的行业标准或者技术规范集合,其中元素 S_k 是一项标准或者规范。

服务的全信息描述框架是一个全新的概念,相比于其它服务建模方法有如下特点:

(1)文献[3,4]从服务语义层次研究服务的发现算法。语义层次只是服务的一个内在方面,不能全面展示服务的内容,在算法研究中也不能满足用户全信息的需要,所以在对服务的刻画上力度不够。

(2)文献[5]建立了服务 QoS 三维模型:宿主节点维、服务维和方法维,在此基础上研究了服务的发现算法,而服务全信息描述框架则是建立在另一个坐标系中,从另一个角度立体剖析了服务的层次模型,相比于 QoS 模型来说,其涵盖范围更广泛,划分方法也更加科学。

2.2 Web 服务的层次结构

网络环境下,Web 服务是一种分布式的资源提供方式,根据用户需求的特点,通过服务发现和组合,以满足用户的任务需求。处于网格上的任何一个用户可以随时随地获取信息,实现 Web 服务调用的“即插即用”,一旦完成任务,用户释放获得的网络和服务资源。Web 服务的这种动态重新装配和再分配服务资源的能力,很大程度上提高了网络应用的灵活性。

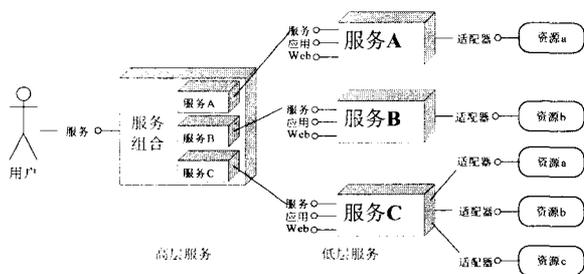


图 2 Web 服务的层次结构图

由图 2 的层次结构可以看出,服务发现算法的优劣决定了服务质量,服务质量的高低又决定了服务组合的综合效能。如何从已有服务中遴选出满足用户需求的服务,又如何从遴选出的服务中优化排序,选择综合效能指标更优的服务,是决定 Web 服务发现查找性能的关键因素。

3 基于全信息描述框架的 Web 服务发现算法

由于 UDDI 注册中心没有集成服务的筛选和过滤机制,导致服务的查全率查准率都不高,更不能实现在满足基本需求情况下服务的优化。本文在全信息描述框架下,对 Web 服务进行了形式化描述,并通过服务选择算法实现满足用户基本需求的服务的发现,而后通过基于 ExAHP 的服务优化排序算法实现 Web 服务按综合效能排序。

3.1 服务形式化描述

全信息描述框架从许多方面描述了 Web 服务信息,在服务发现算法中需要从中遴选出对其有影响的参数,总结起来,这些约束参数可以分为 3 类:选择匹配约束、定量指标参数和定性指标参数。一般的服务选择算法只是对服务的定量指标进行了约束,而没有考虑其他选择匹配和定性指标参数的影响。为了不失偏颇,对全信息描述框架来说,选择算法要充分考虑这 3 类参数的影响,首先应对后两种参数进行预处理:

(1)对于选择匹配约束,算法中优先对此类约束进行选择,从候选条目中确定其具体的参数值(或其集合)。

(2)对于定性指标参数,对其进行量化处理,也即是确定归一化处理,将指标参数用量的形式表示出来,定性指标量化一般采用序标度和区间标度两种成熟的做法,本文采用序标度方法,不再赘述。

定义 4 基于全信息描述框架的 Web 服务描述向量为:

$$cpS = \{(PrI_1, \dots, PrI_l), (SeI_1, \dots, SeI_m), (SyI_1, \dots, SyI_n)\} \quad (1)$$

其 3 个子向量分别是语用信息向量、语义信息向量和语法信息向量,该向量支持 Web 服务实际参数的度量,也支持用户对目标服务指标的需求。

定义 5 约束关系 $r(q, u)$ 是任意 $q \in Q$ 对应度量空间的二元关系。其中 q 为服务度量指标的实际值, u 表示用户对度量指标的期望值,假设有 n 个用户,其期望值的计算公式为:

$$u = \sqrt{\sum_{i=1}^n u_i^2 / n}, 1 \leq i \leq n \quad (2)$$

对于选择匹配约束,其基本约束关系的集合可表示为 $R = \{=, \neq\}$,对于定量指标和量化后的定性指标,其基本约束关系的集合可表示为 $R_c = \{<, =, >, \leq, \geq\}$ 。

定义 6 任意 $q \in Q$,用户对目标服务的需求约束可表示为 $r(q, u), r \in R$ 。用户对 Web 服务需求可表示为:

$$reqU = \{(r_{11}(PrI_1, u_{11}), \dots, (PrI_l, u_{1l})), (r_{21}(SeI_1, u_{21}), \dots, (SeI_m, u_{2m})), (r_{31}(SyI_1, u_{31}), \dots, (PrI_n, u_{3n}))\} \quad (3)$$

$r_{ij} \in R_c \cup R_c, 1 \leq i \leq 3, 1 \leq j \leq \max\{l, m, n\}$;如果用户对某项属性不做要求,则对应单元为空;如果用户对某个信息向量不做要求,则对应的需求子向量为空;如果用户对所有全信息描述向量不做要求,则对应的整个需求向量为空^[7]。

3.2 服务选择算法

Web 服务发现算法主要应用于服务发现阶段,而通用的服务发现技术,各有优缺点,通过 Precision, Recall, Hardness 3 个指标对 Keyword-Based, Frame-Based, Deductive retrieval 3 种方法进行比较,如表 1 所列。

表 1 现有的服务发现技术比较

Service discovery technologies	Precision	Recall	Hardness
Keyword-Based	Low	High	Average
Frame-Based	High	Low	Average
Deductive retrieval	High	High	Hard

由 Kim 等人的研究可知,服务的生命周期不尽相同,而在 UDDI 注册中心并没有对服务的全生命周期进行有效管理,故盲目的通过关键字、ID 号或者定量指标的匹配并不能确定发现的服务是否可用,能否满足用户基本需求。服务选择算法就是在对 Web 服务形式化描述的基础上,确定满足用户基本需求的服务的集合,如下述算法^[8]。

算法 1 SelectbyPara(reqU, Dim, cpS)

```

1. cpSset = φ
2. for (j=0; j<reqU[Dim].length; j++) do
3.     if reqU[Dim][j] is false for corresponding value of cpS[Dim][j] then
4.         break;
5.     else
6.         add cpS[Dim][j] to cpSset;
7. return cpSset

```

对于 $R_i = \{<, =, >, \leq, \geq\}$ 的约束,算法 1 实现在一个维度上遍历,从中选出满足用户需求的候选服务,供算法 2 和算法 3 调用。

算法 2 SelectbySyntax(reqU, cpS)

```

Require: cpS ≠ φ, reqU[SyI] ≠ null
1. Dim = SyI
2. for(j=0; j<reqU[Dim].length; j++)
3.     {
4.     switch
5.     case (rij ∈ Rs): select all paras (rij == '='); break;
6.     case (rij ∈ Rk): select all paras (rij reach the specific value); break;
7.     case (rij ∈ Rk): cpS = SelectbyPara (reqU, Dim, cpS); break;
8.     }
9. return cpSset

```

算法 2 实现语法层次的服务选择,首先对 $r_{ij} \in R_s$ 的选择匹配约束,确定其选择匹配类型,对 $r_{ij} \in R_k$, 达到某一阈值,即满足要求约束,达到某一具体数值,即满足用户需求,对 $r_{ij} \in R_k$ 类型的数值约束,调用算法 1 选择满足约束要求的服务。最后返回满足语法约束条件的候选服务集合 cpS_{set} 。

语义匹配主要是判断服务基本语法元素形成的语义是否合理和正确,是否满足用户的语义需求,是否存在相互矛盾、冲突和不合理的内容。语义匹配是在语法验证正确的基础上进行的。语用匹配主要是分析候选服务是否满足各项需求指标,以及满足需求指标的程度如何。由于语义和语用信息都只包含后两类约束,即定量和量化后的定性指标参数,故此处将两个层次统一考虑。

算法 3 SelectbySemPri(reqU, cpS)

```

Require: cpS ≠ φ
1. for(i=0; i<2; i++)
2.     for(j=0; j<reqU[Dim].length; j++)
4. if reqU[SeI] ≠ null then
5.     Dim = SeI;
6.     cpSset = SelectbyPara(reqU, Dim, cpS);
7. else if reqU[PrI] ≠ null then
8. Dim = PrI;
9. cpSset = SelectbyPara(reqU, Dim, cpS);
10. return cpSset

```

算法 3 在语法匹配的基础上,调用算法 1 对语义和语法信息进行匹配,最后输出满足语法、语义和语用约束条件的服务候选集合 cpS_{set} 。

在全信息描述框架的支持下,算法 1, 算法 2 和算法 3 首先从 PrI, SeI, SyI 三个维度来确定选择匹配约束的参数或其集合,而后对定量指标和量化后的定性指标进行选择,若有一项不满足用户基本需求,则放弃此服务。

3.3 服务优化排序算法

经过服务选择算法的过滤后,服务集 cpS_{set} 均为满足用户基本需求的候选服务。服务组合总体能力的发挥则依赖于候选服务的质量,综合指标高的服务的组合能使其发挥更大的效能。在此,给出 Web 服务效能的概念,即是指在网络环境下,Web 服务执行其总体功能所能达到预期目标的程度。在全信息描述框架下,Web 服务效能即是指 Web 服务在 PrI, SeI, SyI 全信息描述下完成预期目标的程度。

由于层次分析法非常适合解决半结构化的复杂问题,它将问题中的各种因素通过划分相互联系的有序层次使之合理化,根据对一定客观现实的主观判断(两两比较)将每一层次的因素的相对重要性进行定量描述,利用数学方法确定反映每一层次全部因素的相对重要性次序的权值。通过所有层次之间的总排序来确定所有方案的排序。

对于选择匹配约束,优化排序算法中将不予考虑,对于定量指标和量化后的定性指标,其基本约束关系的集合可表示为 $R_i = \{<, =, >, \leq, \geq\}$, 由于量纲不统一,现需对此类指标进行预处理:

$$q' = \begin{cases} 1/q, & \text{if } R_i \in \{<, \leq\} \\ q, & \text{if } R_i \in \{=, >, \geq\} \end{cases} \quad (4)$$

其中, $q \in \{PrI, SeI, SyI\}$ 。

层次分析法是定性方法与定量方法的集合,但由于人为主观因素的参与,使得算法中容易出现较大的误差。本文 Web 服务优化算法则采用改进了的 ExAHP^[9-12], 该算法能有效降低主观因素的影响,提高算法的准确度。为方便起见, cpS_{set} 中 PrI, SeI, SyI 指标数量仍然用 l, m, n 表示。服务优化算法共分为 4 个步骤。

步骤 1 根据 Web 服务效能指标,建立如图 3 所示的层次结构。

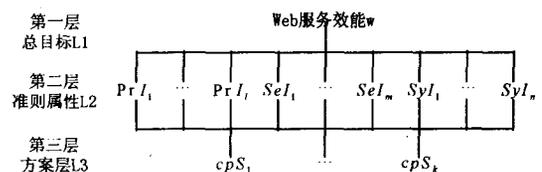


图 3 服务选择层次结构图

步骤 2 构造判断矩阵。

对于各层上的元素可以依次相对于与之有关的上一层元素,进行两两比较,从而建立一系列的判断矩阵,为了有效降低主观判断中个别权重任务的影响,假设 n 个用户,下文中所有比较权重的计算公式为:

$$w = \sum_{j=1}^n w_j \times \lambda_j + \gamma \quad (5)$$

$$1 \leq j \leq n, 0 \leq \lambda_j \leq 1 \text{ 且 } \sum_{j=1}^n \lambda_j = 1$$

其中, λ_j 为第 j 个用户的权重, γ 为客观因素的影响。表 2 给出 L2 层判断矩阵。

表 2 L2 层判断矩阵

W	Pr I ₁	...	Pr I _l	Se I ₁	...	Se I _m	Sy I ₁	...	Sy I _n
Pr I ₁	1	...	w _{l,1}	w _{l+1,1}	...	w _{l+m,1}	w _{l+m+1,1}	...	w _{l+m+n,1}
...	...	1
Pr I _l	w _{1,l}	...	1	w _{l-1,l}	...	w _{l+m,l}	w _{l+m+1,l}	...	w _{l+m+n,l}
Se I ₁	w _{1,l+1}	...	w _{l,l+1}	1	...	w _{l+m,l+1}	w _{l+m+1,l+1}	...	w _{l+m+n,l+1}
...	1
Se I _m	w _{1,l+m}	...	w _{l,l+m}	w _{l+1,l+m}	...	1	w _{l+m+1,l+m}	...	w _{l+m+n,l+m}
Sy I ₁	w _{1,l+m+1}	...	w _{l,l+m+1}	w _{l+1,l+m+1}	...	w _{l+m,l+m+1}	1	...	w _{l+m+n,l+m+1}
...	1	...
Sy I _n	w _{1,l+m+n}	...	w _{l,l+m+n}	w _{l+1,l+m+n}	...	w _{l+m,l+m+n}	w _{l+m+1,l+m+n}	...	1

对于第 3 层上各 Web 服务依次相对于上一层的目标属性,表 3 给出如下 L2—L3 层判断矩阵,空格处依表 2 确定其

相对权重值。

表 3 L2—L3 层判断矩阵

Pr I ₁	cpS ₁	...	cpS _k	...	cpS ₁	...	cpS _k	Pr I ₁	cpS ₁	...	cpS _k
cpS ₁	1	cpS ₁	1	cpS ₁	1
...	...	1	1	1	...
cpS _k	1	cpS _k	1	cpS _k	1
Se I ₁	cpS ₁	...	cpS _k	...	cpS ₁	...	cpS _k	Se I _m	cpS ₁	...	cpS _k
cpS ₁	1	cpS ₁	1	cpS ₁	1
...	...	1	1	1	...
cpS _k	1	cpS _k	1	cpS _k	1
Sy I ₁	cpS ₁	...	cpS _k	...	cpS ₁	...	cpS _k	Sy I _n	cpS ₁	...	cpS _k
cpS ₁	1	cpS ₁	1	cpS ₁	1
...	...	1	1	1	...
cpS _k	1	cpS _k	1	cpS _k	1

步骤 3 各个判断矩阵对应的元素排序(权重计算)及其一致性检验。

对于所建立的各个判断矩阵,分别计算其中相应元素的权重,并检验判断矩阵的一致性。所谓层次单排序是指:根据判断矩阵计算对于上一层某元素而言本层次与之有联系的元素重要性次序的权值。计算权值就是求解判断矩阵的最大特征值和相应的特征向量,即求解:

$$PW = \lambda_{\max} W$$

其中, P 为各层次判断矩阵, λ_{\max} 为判断矩阵 P 的最大特征值, W 是对应于 λ_{\max} 的规范化特征向量,则 W 的分量便是相应元素单排序的权值。

判断矩阵中的数值一般是根据经验和数据资料得到的,不可能做到完全一致性。特别是对网络环境下的分布式 Web 服务发现问题更是如此。为考察 ExAHP 得到的结果是否基本合理,需对判断矩阵作一致性检验。

定义 6 一致性指标 $CI = \frac{\lambda_{\max} - n}{n - 1}$, 随机一致性指标:

$$CR = \frac{CI}{RI} \quad (6)$$

其中, RI 为平均随机一致性指标。

当判断矩阵具有完全一致性时, $CI = 0$ 。 CI 越大,判断矩阵的一致性越差;当 $CR < 0.1$ 时,认为判断矩阵具有满意的一致性,否则就需要调整。

步骤 4 总层次排序及一致性检验。

层次总排序即是利用同一层次中所有元素的单排序结果,计算针对上一层所有元素而言,本层次所有元素的权值。

对于第 2 层 L2,其总排序即为该层次单排序。对于第 3 层 L3,设与 L2 对应的本层次元素为 cpS_1, \dots, cpS_k ,单排序结果为 $w_{cpS_1}^j, \dots, w_{cpS_k}^j$,则 L3 层元素的总排序按表 4 进行。

表 4 L3 层元素的总排序

L2—L3 层	Pr I ₁	...	Pr I _l	Se I ₁	...	Se I _m	Sy I ₁	...	Sy I _n
	w _{pr1}}	...	w _{prl}}	w _{sei1}}	...	w _{seim}}	w _{syi1}}	...	w _{syin}}
cpS ₁	w _{cpS1}^1}	...	w _{cpS1}^l}	w _{cpS1}^{l+1}}	...	w _{cpS1}^{l+m}}	w _{cpS1}^{l+m+1}}	...	w _{cpS1}^{l+m+n}}
cpS ₂	w _{cpS2}^1}	...	w _{cpS2}^l}	w _{cpS2}^{l+1}}	...	w _{cpS2}^{l+m}}	w _{cpS2}^{l+m+1}}	...	w _{cpS2}^{l+m+n}}
cpS ₃	w _{cpS3}^1}	...	w _{cpS3}^l}	w _{cpS3}^{l+1}}	...	w _{cpS3}^{l+m}}	w _{cpS3}^{l+m+1}}	...	w _{cpS3}^{l+m+n}}

L3 层 cpS_k 的总排序为:

$$w_{cpS_k} = \sum_{i=1}^l w_{pr_i} w_{cpS_k}^i + \sum_{i=l+1}^{l+m} w_{se_i} w_{cpS_k}^i + \sum_{i=l+m+1}^{l+m+n} w_{sy_i} w_{cpS_k}^i \quad (7)$$

定义 7 层次总排序的一致性指标 $CI = \sum_{i=1}^m w_{l_{ps}} CI_i$, 随机一致性指标

$$RI = \sum_{i=1}^m w_{c_{ps}} RI_i \quad (8)$$

层次总排序的随机一致性比例 $CR = \frac{CI}{RI}$, 其中 CI_i 是对应于 $w_{c_{ps}}$ 的 P 层判断矩阵的一致性指标, RI_i 是对应于 $w_{c_{ps}}$ 的 P 层判断矩阵的随机一致性指标。当 $CR < 0.1$ 时, 认为判断矩阵具有满意的一致性, 否则就需要调整。

算法 4 CpsSortOrder($w_{c_{ps_k}}$, cpS_{set})

```
Require:  $cpS_{set} \neq \emptyset$ 
1. for( $i=0; i < k; i++$ )
2.   for( $j=i+1; j < k; j++$ )
3.     if( $w_{c_{ps_j}} < w_{c_{ps_i}}$ ) do
4.        $cpS_{max} = cpS_i$ 
5.        $cpS_j = cpS_i$ 
6.        $cpS_i = cpS_{max}$ 
7. return  $cpS_{set}$ 
```

在得到总排序 $w_{c_{ps_k}}$ 和候选服务集 cpS_{set} 的基础上, 通过算法 4 就可以实现按 Web 服务效能的排序, 排在 cpS_{set} 上面的为综合效能高的服务, 下面的为效能较低的服务, 为用户从中选择最优的服务提供了依据, 也为后续服务组合性能优化打下了坚实的基础。

4 实验和分析

实验分析借鉴信息检索中性能评价标准, 用查全率和查准率两个指标来衡量基于全信息描述框架的 Web 服务发现算法的优劣。实验基于以下的假设。

假设 1 根据文献[1], 假设 34% 的 Web 服务可用, 而且每周大约有 16% 的已注册可用的 Web 服务失效, 采样周期间隔 T 为 1 周, $ExAHP$ 算法中比较权重取 0.8, Web 服务指标集数为 $Num_{l+m+n} = \{9, 12, 15, 18, 24\}$ 。

图 4 为一个采样周期内, 候选服务数为 300 时, 随着服务指标集数量的增加, 随机算法和优化算法查全查准率分布情况, 从图中可以看出, 在候选服务数一定时, 随机算法的查全查准率都很低, 并且会随着采样周期的延长, 比率会进一步降低。对于优化算法来说, 由于其采用了两步走的算法实施方案, 随着指标集的增加, 对服务的刻画也越来越完善, 越能符合用户对服务的期望, 其查全率能维持在一个较高水平, 查准率则会越来越高。

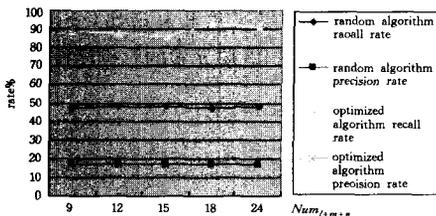


图 4 服务指标集增加时随机和优化算法查全查准率分布

结束语 基于全信息描述框架, 在 PrI , SeI , SyI 3 个维

度对 Web 服务做了形式化描述, 对选择匹配约束、定量指标和定性指标确立了处理原则并进行了量纲的统一, 建立了 Web 服务的选择算法, 以满足用户对服务的基本需求。在此之上, 通过 $ExAHP$ 算法, 建立了 Web 服务效能的优化排序算法, 该算法能有效提高服务的查全率和查准率。最后通过实验和分析, 证明了优化算法对查全查准率的贡献要远远大于随机算法。下一步将继续深入研究 Web 服务匹配和组合问题。

参考文献

- [1] Kim SM, Rosu MC. A survey of public web services[A]//Proceedings of 13th international conf. on World Wide Web (WWW2004)[C]. USA: ACM, 2004: 312-313
- [2] 魏巍, 罗爱民, 陈洪辉, 等. 指控能力包中军事领域应用服务的全信息描述框架研究[J]. 中国电子科学研究院学报, 2008, 3(3): 262-265
- [3] 崔晓红, 刘兴伟, 姚书怀. 语义 Web 服务匹配算法的研究[J]. 西华大学学报, 2006, 25(4): 18-23
- [4] 许卓明, 石磊. 基于语义的 Web 服务发现算法研究[J]. 计算机应用与软件, 2006, 23(5): 21-23
- [5] 郭得科, 任彦, 陈洪辉, 等. 一种 QoS 有保障的 Web 服务分布式发现模型[J]. 软件学报, 2006, 17(11): 2324-2334
- [6] Ferbrache D. Network Enabled Capability: concepts and delivery [J]. Ministry of Defense, Journal of Defense science, 2003, 8(3): 104-107
- [7] Sirin E, Hendler J, Parsia B. Semi-automatic composition of Web services using semantic description[C]//Accepted to Web Services: Modeling, Architecture and Infrastructure workshop in conjunction with ICEIS2003
- [8] Yu Tao, Lin K-J. Service selection algorithms for web services with end-to-end QoS constraints[C]//Proceeding of IEEE International Conference on E-Commerce Technology (CEC'04). 2004: 129-136
- [9] Sycara K, Paolucci M, Ankolekar A, et al. Automated discovery, interaction and composition of semantic Web service[J]. Journal of Web Semantics, 2003, 1(1)
- [10] Paolucci M, Hanamura T, Payne T R, et al. Semantic Matching of web Services Capabilities[M]. Berlin/Heidelberg: Springer, 2002
- [11] Cormen T H, Leiserson C E, Rivest R L, et al. Introduction to Algorithms[M]. MIT OpenCourseWare, 2001
- [12] Schlosser M, Sintek M, Decker S, et al. A scalable and ontology-based P2P infrastructure for semantic Web services[C]//Peer-to-Peer Computing. Second international conference. 2002