

一种服务聚合中 QoS 全局最优服务动态选择算法^{*}

刘书雷⁺, 刘云翔, 张帆, 唐桂芬, 景宁

(国防科学技术大学 电子科学与工程学院, 湖南 长沙 410073)

A Dynamic Web Services Selection Algorithm with QoS Global Optimal in Web Services Composition

LIU Shu-Lei⁺, LIU Yun-Xiang, ZHANG Fan, TANG Gui-Fen, JING Ning

(School of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China)

+ Corresponding author: Phn: +86-731-4573480, E-mail: smartlliu@yahoo.com.cn

Liu SL, Liu YX, Zhang F, Tang GF, Jing N. A dynamic Web services selection algorithm with QoS global optimal in Web services composition. *Journal of Software*, 2007,18(3):646–656. <http://www.jos.org.cn/1000-9825/18/646.htm>

Abstract: As a new Web pattern, Web service has been rapidly developed in recent years. How to dynamically integrate the existent Web services to form a newly value-added and complex service to meet the requirement of different users is a popular research area. This paper presents an algorithm GODSS (global optimal of dynamic Web services selection) to resolve dynamic Web services selection with QoS global optimal in Web services composition. The essence of the algorithm is that the problem of dynamic Web Service selection with QoS global optimal is transformed into a multi-objective services composition optimization with QoS constraints. The theory of intelligent optimization of multi-objective genetic algorithm is utilized to produce a set of optimal Pareto services composition process with constraint principle by means of optimizing various objective functions simultaneously. Theoretical analysis and experimental results indicate the feasibility and efficiency of this algorithm.

Key words: Web services composition; service selection; QoS global optimal; GODSS (global optimal of dynamic Web services selection); multi-objective genetic algorithm

摘要: Web 服务作为一种新型的 Web 应用模式,近年来得到了迅速的发展.如何动态地把现存的各种 Web 服务整合起来以形成新的、满足不同用户需求的、增值的复杂服务已成为新的应用需求和研究热点.针对现有服务聚合中服务选择技术的不足,提出了一种解决服务聚合中服务动态选择 QoS 全局最优化问题的实现算法 GODSS(global optimal of dynamic Web services selection).算法的主要思想是把服务动态选择全局最优化问题转化为一个带 QoS 约束的多目标服务组合优化问题,利用多目标遗传算法的智能优化原理,通过同时优化多个目标函数,最终产生一组满足约束条件的 Pareto 优化服务聚合流程集.理论分析和实验结果说明了算法的可行性和有效性.

关键词: 服务聚合;服务选择;QoS 全局最优化;GODSS(global optimal of dynamic Web services selection);多目标遗

^{*} Supported by the National High-Tech Research and Development Plan of China under Grant Nos.2002AA134010, 2002AA134020, 2003AA135110 (国家高技术研究发展计划(863))

Received 2005-05-11; Accepted 2006-02-23

传算法

中图法分类号: TP393

文献标识码: A

Web 服务作为一种新型的分布式计算模型,近年来得到了工业界和学术界的极大关注.在实际应用中,单个 Web 服务一般提供一些比较单一的功能,通常无法满足复杂应用的需求.因此,如何集成单一服务所提供的各种功能以形成新的、功能更强大的服务来满足不同用户的复杂应用需求已成为一个新的研究热点.

服务聚合是指对现有的一组服务按照一定的业务逻辑进行集成,构建服务执行流程,从而更好地满足用户的需求.服务聚合广义上可以分为静态聚合、半自动聚合和自动聚合^[1].在 Web 环境中,服务是经常变化的,大部分情况下,静态聚合的方式并不能满足实际的应用需求,同时,由于完全智能化的自动聚合方式是一个非常复杂的过程,因此,很多关于服务聚合的应用和研究工作都侧重于半自动方式^[2,3].半自动服务聚合方式的实现,首先要由业务人员根据特定的行业背景,建立适合具体应用需求的通用服务聚合流程模型;服务聚合流程模型由多个服务结点组成,各服务结点包含功能需求描述,但不指定具体的服务调用实例;在 Web 环境中,满足相同功能需求而具有不同 QoS 参数(如执行时间、费用、可靠性等)的 Web 服务存在多个,如何从中选择满足各服务结点功能需求的具体服务,形成一个可执行的服务链来完成用户的需求就成为服务聚合中的一个关键问题,本文称其为服务动态选择问题.

目前,关于服务动态选择的研究绝大部分都是基于 QoS 局部最优的原则^[2,4,5],即对于满足聚合流程模型单个服务结点功能需求的一组服务,根据服务的各个 QoS 参数信息进行加权和排序,并以此为依据分别为聚合流程模型的各个服务结点选择加权和最大的服务来执行流程服务结点的功能.在这些方法中,各个服务结点对应具体服务实例的选择是相互独立的,并不能解决服务聚合的 QoS 全局优化问题,如要求聚合流程在满足可靠性和信誉等级均满足一定条件的情况下执行费用极低、时间极短.对于服务选择的 QoS 全局最优问题,目前的研究工作不多,文献[3,6]通过把服务聚合流程的各个 QoS 约束参数*线性加权转化为一个单目标函数,利用线性规划的基本原理来解决服务选择的 QoS 全局最优化问题;但文献[3,6]提出的方法存在 4 个缺陷:1) 把聚合流程中各个 QoS 参数线性加权转化为一个目标函数,因此产生的最优路径是满足约束条件的单目标最优解,不能解决多目标的优化问题,如同时优化聚合流程的执行时间和费用;2) 由于加权和的解其结果不仅对权重向量非常敏感,而且需要用户对问题有一定的认识,如参数的优先级、一个参数对其他参数的影响程度等;3) 产生的优化结果为单个解,用户没有选择的余地.通常情况下,用户更关心能否获得一组可接受的最优非劣解,根据实际需要选择最满意的聚合流程;4) 线性规划要求目标函数和约束均是线性的,这在一定程度上限制了算法的实用性.

为此,本文提出了一种基于多目标遗传算法的 QoS 全局最优服务动态选择(global optimal of dynamic Web services selection,简称 GODSS)实现算法,通过把服务聚合中的服务动态选择 QoS 全局最优问题转化为一个带约束条件的多目标服务组合优化问题,利用多目标遗传算法的智能优化原理,通过同时优化多个目标参数,即在不同的目标之间取均衡,最终产生一组满足约束条件的最优非劣服务聚合流程集;用户可以根据特定需要从中选择最满意聚合流程,同时,未被选中的非劣路径可以作为备选方案,以便所选聚合路径发生意外时启用.本文假设各个具体服务的 QoS 参数信息在算法执行时是静态不变的.

本文第 1 节描述服务选择全局最优模型.第 2 节描述服务聚合流程基本模型及其 QoS 评价方法.第 3、4 节对 GODSS 进行了描述、分析.第 5 节描述实验结果.第 6 节进行总结.

1 问题描述

假设 $G(N, E, W)$ 表示有向图, N 为顶点集, E 为链路集, W 为链路权重;假设链路权重度量指标数目为 l , 则每条链路具有一个 l 维的链路权重向量,其中包括 l 个非负参数($w_i(u, v), i=1, \dots, l, (u, v) \in E$).

定义 1(多约束条件下多目标最优路径(multi-constrained, multi-objective optimal path,简称 MCOOP)).

* 服务流程可以看作一个新的复杂服务,并通过服务的 QoS 参数进行度量.

对于网络 $G(N,E,W)$,存在 $k(k \geq 2)$ 个约束 $C_i(i=1,\dots,k)$,同时,每一条从源点 S 到目的点 T 的路径 P 具有 $m(m \geq 2)$ 个非负的性能度量准则 f_1,\dots,f_m,P 为多约束条件下多目标最优非劣路径当且仅当 $\forall P^*(P^* \neq P)$,在 P 和 P^* 均满足约束 C_i 的条件下,对于所有的度量准则均使得 $f_i(P) \succ = f_i(P^*) (i=1,\dots,m)$,且至少存在一个目标 i 满足 $f_i(P) \succ f_i(P^*)$. 其中, $\succ =$ 和 \succ 分别表示度量准则之间的不劣于和优于关系.

动态服务聚合中,通用服务聚合流程模型由多个服务结点组成,每一个服务结点对应一个服务群.

定义 2(服务结点(service node,简称 SN)). 服务结点是构成服务聚合流程模型的基本逻辑单元,服务结点仅包含功能描述和接口信息,并不指向具体的 Web 服务;动态服务聚合流程建模中,建模人员利用服务结点按照特定的业务逻辑建立通用服务聚合流程模型.

定义 3(服务群(service group,简称 SG)). 服务群是指由不同服务提供者提供的、具有相同调用接口、能够实现相同功能的一组服务.同一服务群中服务具有相同的功能,所不同的是各个服务的 QoS 属性.

假设一串行聚合流程模型包括 m 个服务结点(如图 1 所示),服务结点 SN_i 对应的服务群 SG_i 包含 n_i 个服务,即 $SG_i = (WS_{1,i}, \dots, WS_{n_i,i})$,将每个服务结点所对应的服务群中的所有服务看作图的顶点;同时,为便于分析问题,增加两个虚拟服务 S 及 T ,分别对应服务聚合流程的起始顶点和终止顶点,则服务聚合流程图定义如下:

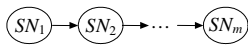


Fig.1 Sequence Web services composition process

图 1 串行服务聚合流程

定义 4(服务聚合流程图(services composition graph,简称 SCG)). 设服务聚合流程模型包括 m 个服务结点,服务结点对应的服务群为 $SG_i(1 \leq i \leq m)$,则服务聚合流程图 $SCG=(N,E,W)$,其中: $N=SG_1 \cup \dots \cup SG_m \cup \{S,T\}$; E 由下列 3 种链路所组成:

- (1) 链路 $(S,u), \forall u \in SG_1$;
- (2) 链路 $(v,T), \forall v \in SG_m$;
- (3) 链路 $(u,v), \forall u \in SG_i, \forall v \in SG_j, (SN_u, SN_v)$ 为聚合模型的链路.

链路 $(u,v) \in E$ 的权重向量为顶点 v 对应 Web 服务的 QoS 参数向量.设虚拟服务 S 及 T 的 QoS 参数均为 0,由定义 4,图 1 对应的服务聚合流程如图 2 所示.

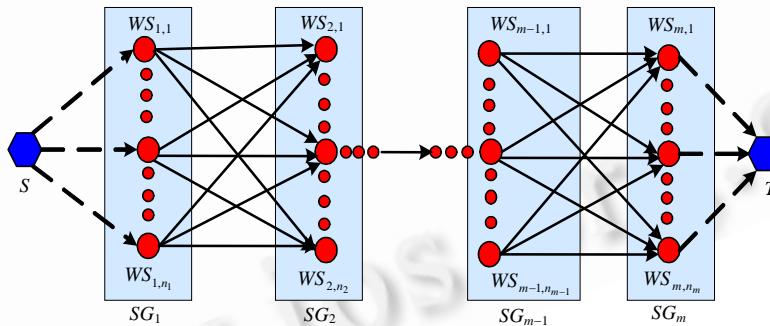


Fig.2 SCG of sequence services composition process

图 2 串行服务聚合流程图

服务动态选择 QoS 全局最优就是在聚合流程模型执行过程中,从各个服务结点对应的服务群中选择具体的服务组成一个可执行的服务链,使得服务链在满足特定 QoS 约束的前提下,多个目标(QoS 参数)达到最优化.由定义 1,可以把服务聚合的服务动态选择 QoS 全局最优化问题转化为一个求解 SCG 中从 S 到 T 带 QoS 约束条件的多目标最优路径问题,也即 SCG 中的 MCOOP 求解问题.本文研究并提出了求解该问题的算法 GODSS.

2 服务聚合流程基本模型及 QoS 计算方法

服务聚合流程实际上是一个基于服务的工作流,与 WfMC 定义的工作流 4 种基本模型^[7]相对应,本文给出了服务聚合流程的 4 种基本模型和基于约简规则的模型 QoS 计算方法,大部分服务聚合流程都可以由这 4 种基本模型组合而成,聚合流程的 QoS 参数也可以通过基本模型的 QoS 评价方法来获取.基于 Petri 网的流程建模方法,服务聚合流程基本模型及其约简规则表示如图 3 所示,其中:变迁代表服务;变迁的前置库所和后置库所分

别表示服务的输入和输出.假设 Web 服务包括 4 种 QoS 参数,即执行时间 $T(\text{time})$ 、执行费用 $C(\text{cost})$ 、信誉等级 $Rep(\text{reputation})$ 和可靠性 $R(\text{reliability})$.设 cs 为多个服务形成的聚合服务, s_i 为组成聚合服务的单个服务, s_i 和 cs 的服务质量模型分别为 $Qs_i=(T_i, C_i, Rep_i, R_i), Q_{cs}=(T_{cs}, C_{cs}, Rep_{cs}, R_{cs})$,服务聚合流程基本模型的 QoS 参数计算方法如下:

(1) 串联模型(如图 3(a)所示):

$$T_{cs} = \sum_{i=1}^n T_i, C_{cs} = \sum_{i=1}^n C_i, R_{cs} = \prod_{i=1}^n R_i, Rep_{cs} = \sum_{i=1}^n Rep_i / n;$$

(2) 并联模型(如图 3(b)所示):为便于描述,引入了辅助变迁 t_a, t_b (辅助变迁对服务流程的 QoS 参数没有贡献,仅用于优化流程结构),

$$T_{cs} = \text{Max}(T_1, T_2, \dots, T_n), C_{cs} = \sum_{i=1}^n C_i, R_{cs} = \text{Min}(R_1, R_2, \dots, R_n), Rep_{cs} = \sum_{i=1}^n Rep_i / n;$$

(3) 选择模型(如图 3(c)所示):设第 i 个分支被选中的概率是

$$\alpha_i, \sum_{i=1}^n \alpha_i = 1, T_{cs} = \sum_{i=1}^n T_i \alpha_i, C_{cs} = \sum_{i=1}^n C_i \alpha_i, R_{cs} = \sum_{i=1}^n R_i \alpha_i, Rep_{cs} = \sum_{i=1}^n Rep_i \alpha_i;$$

(4) 循环模型(如图 3(d)所示):设循环次数为 k ,则

$$T_{cs} = k * \sum_{i=1}^n T_i, C_{cs} = k * \sum_{i=1}^n C_i, R_{cs} = \prod_{i=1}^n R_i, Rep_{cs} = \sum_{i=1}^n Rep_i / n.$$

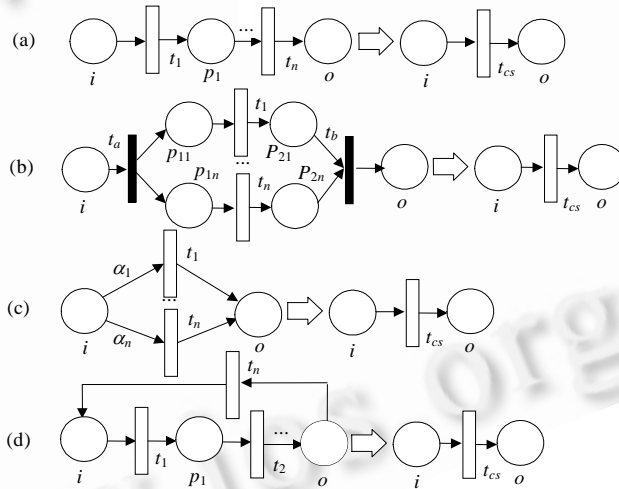


Fig.3 Basic services composition process models and its reduction rules

图 3 服务聚合流程基本模型及约简规则

3 GODSS 算法描述

3.1 GODSS思想和模型

为方便阐述,本文将执行费用和时间作为两个目标准则,希望聚合流程执行费用极少、时间极短;信誉等级和可靠性作为两个约束条件, Rep_0 和 R_0 分别表示所要求的服务聚合路径的最低信誉等级和最低可靠性,则一个带约束条件的多目标服务聚合优化模型可以形式化地描述如下:

$$\text{Min}F(P)=(T(P), C(P)) \tag{1}$$

式(1)中,路径 P 受限于两个约束条件,即

$$\begin{cases} Rep(P) \geq Rep_0 \\ R(P) \geq R_0 \end{cases} \quad (2)$$

其中: $T(P), C(P), Rep(P)$ 和 $R(P)$ 对应服务聚合流程的 QoS 参数计算式,可以通过第 2 节定义的服务聚合模型 QoS 评价方法来获取;式(1)表示取向量极小化,使得目标 $F(P)$ 中的目标函数值均同时极小化.实际上,服务 QoS 参数的种类很多,主要思想是将它们看作优化模型的目标函数和约束条件.所以,本文模型可以推广到任意多个目标函数和约束条件.

多目标优化的特点在于各目标之间的相互冲突性(执行时间短的服务其执行费用往往比较高),所以,多目标服务聚合优化模型最后的结果不是单一解,而是在服务流程执行时间和费用之间取折衷,得到一个满足约束条件的 Pareto 优化解集.优化解集中各条路径之间的比较采用了相对优于的概念,由于带约束的多目标优化问题不同于不带约束的问题,它比绝对的 Pareto 优于关系要复杂,因此,本文采用文献[8]的方法来区分问题解空间中不同个体的相对优于关系.

文献[9]证明了 MCOOP 问题是一个 NP-Complete 问题.遗传算法作为一种智能优化方法,具有并行计算、群体寻优的特点;同时,不同于加权法等传统的多目标优化方法,遗传算法不需要与应用背景相关的启发式知识,只需要目标函数和相应适应值函数.因此,它已广泛应用于各种 NP-Complete 问题的求解.

基于上述思想和模型,本文基于多目标遗传算法设计了求解服务聚合中服务动态选择 QoS 全局最优化问题的实现算法 GODSS;通过同时优化聚合路径的多个 QoS 参数,最终产生一组满足约束条件的 Pareto 最优解,用户可以根据特定需要从中选择最满意解;未被选中的非劣路径可以作为备选方案,在所选聚合路径发生意外时启用.

3.2 过程描述

GODSS 算法从多目标优化的角度出发,搜索 SCG 中 S 和 T 之间满足所有约束条件的一组多目标 Pareto 非劣解.基本思想是:将每一个服务聚合流程编码为一个染色体,通过染色体之间的交叉、变异等重组操作,产生具有更高目标标准值的新染色体.这一过程不断进行,也就实现了在解空间的并行全局搜索;算法停止时,得到一个染色体集合,也就对应了多条收敛于 Pareto 优化或近似优化路径的聚合路径集.具体流程见算法 1.

算法 1. GODSS.

输入:种群 P_1 及辅助种群 P_2 的规模,进化世代数 T .

输出:Pareto 解集 P^* .

```

Begin
(1)  $t \leftarrow 0$ ;
(2)  $P_1^t, P_2^t \leftarrow \emptyset$ 
(3) Initialize
(4) While ( $t \leq T$ )
(5) FitnessSet  $P_1^t$ 
(6)  $P_2^{t+1} \leftarrow OptimalSelect(P_1^t, P_2^t)$ 
(7)  $P_m^t \leftarrow MateSelect(P_1^t), P_2^{t+1}$ 
(8)  $P_1^{t+1} \leftarrow Evolve(P_m^t)$ 
(9)  $t \leftarrow t+1$ 
(10) Goto (4)
(11)  $P^* \leftarrow P_2^t$ 
(12) Output  $P^*$ 
End

```

其中:步骤(3)进行算法初始种群的生成和编码空间的选择,见第 3.2.1 节;步骤(5)进行个体适应值的分配,见第 3.2.2 节;步骤(6)利用种群 P_1 及辅助种群 P_2 进行优良解保持,见第 3.2.3 节;步骤(7)采用轮盘赌选择方式从种群 P_1 中选择优势个体,与辅助种群 P_2 中的个体混合后赋给交配池;步骤(8)执行交叉和变异操作,产生新的染色体作为新种群.对于给定的聚合流程,遗传空间中个体的编码长度相同,交叉和变异产生的路径都是合法的,本文采用两点交叉的方法来加快算法收敛的速度;变异位置的选择应该是除了 S 和 T 以外的其他基因位,基因的变异要在基因对应的服务群范围内进行.由于初始种群采用随机方法产生,初始种群的多样性受到限制,进化过程中的空间搜索能力完全取决于交叉和变异操作,因此,交叉概率和变异概率取值不可过低,交叉概率可取 1,变异概率可取 0.15.步骤(11)、步骤(12)将辅助种群内的非劣解作为服务聚合的可行路径进行输出.非劣解的每一个

基因位对应服务群中的一个具体服务,所有的服务形成一条可执行的服务链.

3.2.1 初始化操作

3.2.1.1 初始聚合路径集的生成

初始聚合路径集的个体对应 SCG 中的一条路径.本文利用随机方法产生一组满足约束条件的从 S 到 T 的初始聚合路径集.具体过程见算法 2:假设种群规模为 N ,目标函数集 $F=\{f_i|1\leq i\leq m\}$,其中: $Constr(P)$ 表示从集合 P 中选取满足约束的路径; $SR(S,T)$ 表示利用随机方法选择 S 到 T 的一条路径.

算法 2. InitPathSelect.

输入:目标函数集 F ,种群规模 N .

输出: P .

Begin

(1) $P\leftarrow\emptyset$

(2) while ($|P|<N$)

(3) $P\leftarrow P\cup Constr(SR(S,T))$

(4) Goto (2)

(5) Output P

End

3.2.1.2 编码策略

首先对聚合流程中的各个服务结点进行顺序排列;其次,增加虚拟服务结点 S 和 T 作为聚合流程的起始点和终止点.每一个给定的聚合流程所包含的服务结点数相同,因此采用整数定长编码的方式,将聚合路径映射为遗传空间中的个体.染色体中第 1 个(最后)基因总是聚合流程图中的顶点 $S(T)$,染色体中间的每一个基因对应一个具体服务在服务群中的编号.

3.2.2 个体适应值分配和多样性保持

按照进化理论,总是希望选择其中具有较高质量的染色体参与后续遗传操作.适应值分配就是给每一个染色体赋予一个标量适应值,从而对当代群体内个体之间优于关系进行排序.本文采用两个步骤进行个体适应值的分配:首先基于多个目标函数对整个群体中属最优非劣解的个体标记为顺序 1;然后,从竞争中移去这些个体,在余下的个体中寻找最优非劣个体并将其标记为顺序 2.该过程持续到种群中所有个体都分配到次序后结束.对于出现多个个体具有相同排序值的情况,通过个体的多样性保持策略进行个体适应值修正,使所获得的近似解分布性好,从而实现种群多样性的保持.

常用的多样性保持策略是小生境技术^[10],其主要思想是利用共享函数来限制相似个体的选择概率.但是,小生境技术对小生境半径 σ_{share} 非常敏感, σ_{share} 的略微偏移会导致共享函数取值的极大偏差,因此,本文采用基于 Shannon 的信息熵理论来保持群体的多样性^[11],它不存在参数的估算和敏感问题.本文认为,相似的个体路径具有较多数目的相同结点.适应值分配过程见算法 3:步骤(4)首先计算个体路径的信息熵 $H_{i,j}$,其中, P 为个体 i 与个体 j 包含的相同结点的比率;步骤(5)计算个体 i 和个体 j 的相似程度 $S_{i,j}$,其中, $S_{i,j}\in(0,1)$;步骤(8)计算个体 i 在群体中的密度 D_i ,其中 N 为群体规模, pop 为种群;步骤(9)进行个体适应值的分配,其中 f_i 表示个体排序时分配的序号值;适应值服从极大化原则,即大适应值的个体具有较高的繁殖概率.

算法 3. FitnessSet.

输入:排序值 f .

输出:适应值 $fitness$.

Begin

(1) for each $i\in pop$ do

(2) $m\leftarrow 0$

(3) for each $j\in pop\wedge i\neq j$ do

(4) $H_{i,j}\leftarrow -P\times\ln P$

(5) $S_{i,j}\leftarrow \frac{1}{1+H_{i,j}}$

(6) if $S_{i,j}>\omega$ ($0.9\leq\omega\leq 1$) then

(7) $m\leftarrow m+1$

(8) $D_i\leftarrow m/N$

(9) $fitness_i\leftarrow \frac{1}{f_i}\times D_i$

(10) Output $fitness_i$

End

3.2.3 优良解保持

由于算法的随机性往往导致在优化过程中产生的优良解丢失,因此,为了保证算法的优化性和收敛性,引入辅助种群对当前产生的优良解实施保留.在进化过程的每一代,通过加入新产生的满足约束条件的非支配解,同时删除所有被支配解对辅助种群进行更新.由于算法涉及多个准则,这样的优良个体可能有很多,为便于分析,希望优化解的数量在一定范围内,所以设辅助种群个体数上界为 N^* .优良解保持具体过程见算法 4.首先从种群 P_1^t 中选出非支配个体 r ,若 r 满足约束条件,则把辅助种群 P_2^t 中受 r 支配的个体表示为集合 ε (步骤(4));若集合 ε 非空,在 P_2^t 中将这受支配个体删除,然后加入个体 r (步骤(6)),若集合 ε 为空,表示 r 与辅助种群中个体相互非支配,为保持辅助种群的规模,不将 r 直接放入辅助种群,而将 r 放入临时集合 $Temp$ (步骤(8));然后根据情况从 $Temp$ 中选取适量个体加入辅助种群,保证辅助种群的个体数目不超过 N^* .其中: $OC(\cdot)$ 表示求解非支配解; $Constr(\cdot)$ 判断个体是否满足约束条件; $Select(n,Temp)$ 表示从临时集合 $Temp$ 中随机选取 n 个个体.

算法 4. OptimalSelect.

输入:第 t 代种群 P_1^t 和辅助种群 P_2^t .

输出:第 $t+1$ 代辅助种群 P_2^{t+1} .

Begin

- | | |
|---|--|
| (1) $r \leftarrow \emptyset, \varepsilon \leftarrow \emptyset, Temp \leftarrow \emptyset$ | (7) else |
| (2) for each $r \in OC(P_1^t)$ do | (8) $Temp \leftarrow Temp \cup \{r\}$ |
| (3) if $Constr(r)$ then | (9) $n \leftarrow \min\{N^* - P_2^t , Temp \}$ |
| (4) $\varepsilon \leftarrow \{\alpha \mid \alpha \in P_2^t \wedge r \succ \alpha\}$ | (10) $P_2^{t+1} \leftarrow P_2^t \cup Select(n, Temp)$ |
| (5) if $\varepsilon \neq \emptyset$ then | (11) Output P_2^{t+1} |
| (6) $P_2^t \leftarrow (P_2^t \setminus \varepsilon) \cup \{r\}$ | End |

4 GODSS 算法分析

4.1 收敛性分析

定理 1. 如果存在满足约束条件的可行服务聚合路径,则 GODSS 在足够大的遗传种群与进化代数情况下能够搜索出可行路径.

证明:GODSS 采用以下 3 种实现策略:(1) 以概率 1 进行交叉操作;(2) 变异概率 $p_m=0.15, p_m \in (0,1)$;(3) 每代群体在选择操作前进行最优解的保持.文献[12]提出的定理 2.7 证明了满足此 3 个条件的遗传算法能够收敛到最优解.所以,在存在可行服务聚合路径的条件下,GODSS 在足够大的遗传种群与进化代数情况下,能够搜索出可行路径.

4.2 算法复杂度

本文对 GODSS 的时间复杂度进行分析,算法 GODSS 的时间复杂度主要由利用遗传算法进行迭代的时间复杂度组成.假设目标函数的数目为 m ,约束条件的数目为 n ,群体规模为 N ,染色体长度为 L ,迭代次数为 T .

遗传算法迭代的时间复杂度主要包括两个部分,即个体适应值分配以及个体之间相对优于关系的比较.个体适应值分配的时间复杂度为 $O(N^2(m+n+L))$.优劣比较在种群与辅助种群的个体之间进行,假设辅助种群的群体规模为 N^* ,比较个体之间的相对优于关系时要进行 $O((m+n)NN^*)$ 比较,则算法 GODSS 总的时间复杂度为 $O(((m+n+L)N^2+(m+n)NN^*)T)$.不难看出,整个算法的时间复杂度与服务聚合流程图的规模、初始种群和辅助种群的规模以及算法迭代次数有关.

5 实验结果及讨论

5.1 实验设计

结合我们正在承担的 SIG(spatial information grid)项目,以城市规划应用为背景来验证算法 GODSS 的可行性和有效性.城市规划应用是针对规划部门的一种自动化处理流程,通过集成地理上分布的各种空间数据资源和空间信息处理资源,为规划部门的决策提供支持,具体的执行过程如图 4 所示,其中, t_1 :应用请求解析服务; t_2 :地理编码服务; t_3 :行政区划数据服务; t_4 :影像数据服务; t_5 :DEM 数据服务; t_6 :交通数据服务; t_7 :地下管道数据服务; t_8 :商业数据服务; t_9 :数据集成服务; t_{10} :缓冲区分析服务.首先,接受规划部门请求,启动规划应用流程;把规划区域中心点的地名转化为地理坐标;通过地理坐标从各个部门获取规划区域的行政区域、影像、DEM、道路交通、地下管道和商业数据,对获取的各种数据进行叠加、集成并进行缓冲区分析,直观显示一定范围内的各种信息,使得规划部门能够直观地了解规划区域的周边环境等信息,为科学决策提供支持.

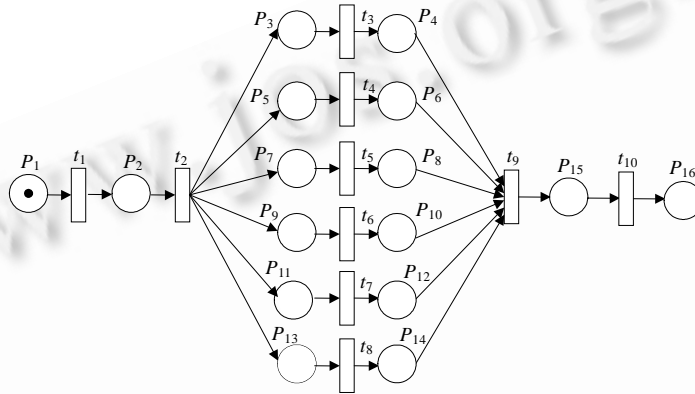


Fig.4 Process of SIG city layout

图 4 SIG 城市规划应用流程

流程中的每一个服务结点对应一个服务群.服务群的建立和维护由服务聚合框架来完成,本文不作描述.实验环境为 100M 局域网,算法运行微机配置为 PentiumIV 2GHz 处理器,512M 内存,操作系统为 Windows2000 Server,算法用 C++实现.各个服务群中服务的 QoS 参数采用随机方法在一定范围内生成.服务及其 QoS 参数信息在集中式 UDDI 注册中心进行注册,参数取值范围设定为 $0 < T \leq 10$ 秒, $0 < C \leq 100$ 元, $0 < Rep \leq 5$, $0 < R \leq 1$,最小信誉等级为 2,最小可靠性为 0.1.把服务聚合流程的执行时间和费用作为两个目标准则,信誉等级以及可靠性作为两个约束条件.希望利用 GODSS 从各个服务结点对应的服务群中选择具体服务形成可执行的服务链,使得服务链在满足两个约束条件的情况下,执行时间极短、费用极少.

5.2 算法参数设置

5.2.1 编码空间的确定

根据第 3.2 节算法的描述,如图 4 所示的服务聚合流程模型的一个染色体编码实例为

0	$\alpha(1)$	$\alpha(2)$	$\alpha(3)$	$\alpha(4)$	$\alpha(5)$	$\alpha(6)$	$\alpha(7)$	$\alpha(8)$	$\alpha(9)$	$\alpha(10)$	0
---	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	-------------	--------------	---

其中, $\alpha(i)$ 表示服务聚合流程在服务结点 t_i 对应的服务群 $SG_i(i \in \{j | 1 \leq j \leq 10\})$ 中所选具体服务的编号.

5.2.2 目标函数及约束函数的确定

由第 3.1 节的描述可知,算法目标函数和约束函数分别对应聚合流程的 QoS 参数计算式:根据第 2 节描述的服务聚合流程基本模型及 QoS 评价方法,对如图 4 所示的模型采用自底向上逐步综合替代的层次分析方法,可以得到流程的 QoS 参数计算公式,其中,图 4 中变迁 $t_3, t_4, t_5, t_6, t_7, t_8$ 构成一个并联模型;基于第 2 节所描述的模型简化规则可以把 $t_3, t_4, t_5, t_6, t_7, t_8$ 简化为一个等价变迁 t_{cs} ,由并联模型 QoS 计算方法可以得到 t_{cs} 的 QoS 参数; t_{cs}

与 t_1, t_2, t_9, t_{10} 组成串联模型,则由串联模型 QoS 计算方法可以得到如图 4 所示的服务聚合流程的 QoS 参数计算式,从而得到算法的目标函数和约束函数.

$$T(P)=T_1+T_2+T_9+T_{10}+\text{Max}(T_3, T_4, T_5, T_6, T_7, T_8), C(P)=\sum_{i=1}^{10} C_i,$$

$$R(P)=R_1R_2R_9R_{10}\text{Min}(R_3, R_4, R_5, R_6, R_7, R_8), \text{Rep}(P)=\left(6*\sum_{i=1}^2 \text{Rep}_i + 6*\sum_{j=9}^{10} \text{Rep}_j + \sum_{m=3}^8 \text{Rep}_m\right) / 30.$$

5.3 实验分析

5.3.1 有效性实验

该实验通过算法执行的 CPU 开销来验证 GODSS 解决实际问题的有效性.实验分别考虑了服务群规模为 5,10 和 20,进化代数取 100,200,300 和 400 的情况下,利用 GODSS 求解满足约束的 Pareto 优化服务链的 CPU 时间开销.对于每一种情况,算法分别运行 10 次取平均值.由图 5 可以看出:随着服务群数目的增加,在不同的进化代数下,CPU 运行时间并没有大量增加.本文考虑的城市规划处理流程包含 10 个服务结点,在服务群规模为 20 的情况下,算法迭代 400 次得到优化解的执行时间在 16 秒左右.这一求解规模一般可以满足大部分服务聚合求解的需要,因此,算法 GODSS 的有效性还是明显的.

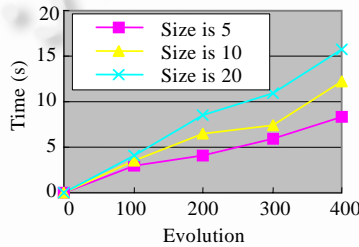


Fig.5 GODSS average time
图 5 GODSS 平均执行时间

5.3.2 可行性实验

该实验的目的是用于验证算法 GODSS 找到 QoS 全局最优服务链的可行性.作比较的另一种方法是穷举法,即逐个地穷举计算出满足约束条件的可执行服务链的各个目标函数值的最优解,图 6 为在服务群规模分别为 5,10,20 的情况下,GODSS 迭代 400 代所获得的关于各优化指标的全局最优结果的百分率.这里的百分率是在相同服务结点和规模条件下算法运行 20 次所得非劣解集中存在某优化指标的穷举最优值的统计概率.从图 6 可以看出:在有限进化代数(400 代)条件下找到各优化指标最优值的概率都在 90% 以上,因此,GODSS 解决带约束多目标优化(QoS 全局优化)问题是可行的.

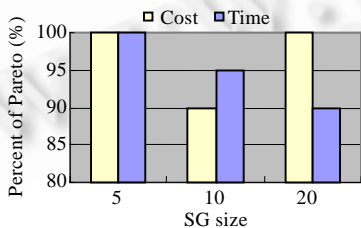


Fig.6 Rate of optimal result
图 6 最优解的比例

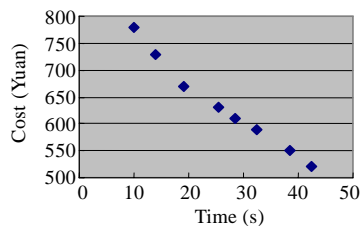


Fig.7 Optimal result
图 7 优化解集

服务群规模为 20,算法迭代 400 代产生的一组非劣解如图 7 所示.从单个目标准则来看,这些解有可能不是最优的;但是,同时从服务链执行时间和费用两个角度来看,这些解是非劣的,解的分布也是均匀的.用户可以根据实际的偏好信息选择最合适的 Pareto 解,因此,算法 GODSS 是有效的.其他没有被选用的 Pareto 解可以作为

备选路径,在聚合过程发生意外时启用.现有的全局最优算法^[3,6]是基于单目标的,最终只能产生一个最优解,因此不具备上述能力.

5.3.3 采用策略分析

GODSS 采用信息熵理论进行种群多样性的保持,采用辅助种群进行优良解保持.假设不采用多样性保持和优良解保持的方法为策略 A,通过对策略 A 与 GODSS 进行比较,来说明本文采用种群多样性保持和优良解保持策略的有效性.图 8 为在服务群规模分别为 5,10,20 的情况下,策略 A 与 GODSS 分别迭代 400 代的运行时间比较,这里的时间为策略 A 与 GODSS 分别运行 10 次取平均.从图 8 可以看出:因为种群多样性策略和优良解保持策略的加入,GODSS 的时间开销要大于策略 A.图 9 为在服务群规模为 20、进化代数 of 400 的条件下,策略 A 与 GODSS 所得的最优解的分布情况.可以看出,GODSS 所求得的最优解性能和分布要优于策略 A.因此,GODSS 虽然计算复杂性高,但其得到的解质量好,适合于 QoS 性能要求严格的服务选择;策略 A 计算复杂性低,但其最终得到的解质量差.对于不同的聚合应用需求,可以根据实际情况灵活地采用最适合的策略.

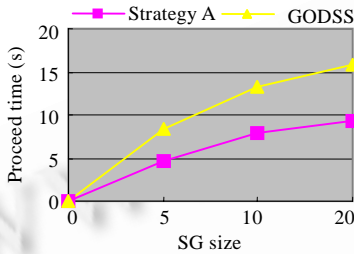


Fig.8 Average execution time of GODSS and A

图 8 GODSS 和策略 A 的平均执行时间

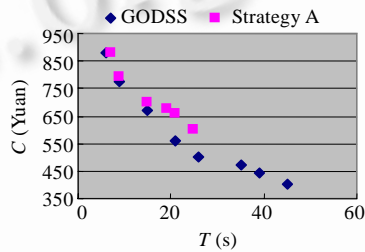


Fig.9 Optimal results of GODSS and A

图 9 GODSS 和策略 A 的优化解集

6 结 论

服务动态选择是服务聚合的一个重要问题.本文针对国内外现有服务选择技术的研究成果大多侧重于 QoS 局部最优、考虑全局优化的工作较少且具有片面性的研究现状提出了动态服务选择 QoS 全局最优模型,并基于多目标智能优化的思想,将服务动态选择 QoS 全局最优问题转化为一个带约束条件的多目标优化问题.提出了一种基于多目标遗传算法的动态服务选择全局 QoS 最优实现算法 GODSS,并对算法进行了分析.最后结合所承担的 SIG 项目中的城市规划应用对算法 GODSS 进行了实验.实验结果证明了算法的可行性和有效性.

由算法描述和实验结果可知,与现有服务选择算法^[2-6]相比,算法 GODSS 的优势在于:1) GODSS 从 QoS 全局最优的角度把服务动态选择问题转化为一个带 QoS 约束的多目标服务组合优化问题,有效地解决了局部最优方法的不足;2) GODSS 通过同时优化多个目标函数,得到一组满足约束条件的非劣解.与现有全局最优算法基于单目标优化仅得到一个优化解相比,更好地满足了用户的需求;3) GODSS 对目标函数和约束函数没有限制,同时在算法运行过程中不需要人为设置各个参数的权值,也不需要与应用背景相关的启发式知识,算法的适用性比较好;4) 在问题的输入规模变大时,现有的确定性算法的性能将急剧下降,GODSS 在求解最优解时,随着染色体长度的增加,染色体交叉、变异操作以及目标函数和约束函数的计算复杂度呈线性增加,从而保证了算法的鲁棒性.

References:

- [1] Shalil M, Walker DW, Gray WA. A framework for automated service composition in service-oriented architectures. In: Bussler C, ed. Proc. of the ESWS 2004. LNCS 3053, Heraklion, Berlin: Springer-Verlag, 2004. 269–283.
- [2] Benatallah B, Dumas M, Sheng QZ, Ngu A. Declarative composition and peer-to-peer provisioning of dynamic Web services. In: Proc. of the 18th Int'l Conf. on Data Engineering. San Jose: IEEE Computer Society, 2002. 297–308.

- [3] Zeng LZ, Benattallah B, Dumas M. Quality driven Web service composition. In: Proc. of the WWW 2003. Budapest: ACM, 2003. 411-421.
- [4] Casati F, Inicki S, Jin LJ, Krishnamoorthy V, Shan MC. eFlow: A platform for developing and managing composition e-services. Technical Report, HPL-2000-36, HP Laboratories Palo Alto, 2000.
- [5] Liu YT, Anne HH, Zeng LZ. QoS computation and policing in dynamic Web service selection. In: Proc. of the WWW 2004. New York: ACM, 2004. 66-73.
- [6] Zhao JF, Xie B, Zhang L, Yang FQ. A Web services composition method supporting domain feature. Chinese Journal of Computers, 2005,28(4):731-738 (in Chinese with English abstract).
- [7] WFMC. Workflow management coalition terminology & glossary. Document number WFMC-TC-1011, document status-issue 3.0. Technical Report, Brussels: Workflow Management Coalition, 1999.
- [8] Cui XX, Lin C. A constrained quality of service routing algorithm with multiple objectives. Journal of Computer Research and Development, 2004,41(8):1368-1375 (in Chinese with English abstract).
- [9] Garey MR, Johnson DS. Computers and Intractability: A Guide to the Theory of NP-Completeness. New York: W.H.Freeman and Company, 1979.
- [10] Wang XP, Cao LM. Genetic Algorithms-Theory, Application and Software Implement. Xi'an: Xi'an Jiaotong University Press, 2002. 74-76 (in Chinese).
- [11] Cui XX, Lin C. Multicast QoS routing optimization based multi-objective genetic algorithm. Journal of Computer Research and Development, 2004,41(7):1144-1150 (in Chinese with English abstract).
- [12] Chen GL, Wang XF. Genetic Algorithm and Its Applications. Beijing: People's Posts & Telecommunications Press, 1996 (in Chinese).

附中文参考文献:

- [6] 赵俊峰,谢兵,张路,杨芙清.一种支持领域特性的 Web 服务组装方法.计算机学报,2005,28(4):731-738.
- [8] 崔逊学,林闯.一种带约束的多目标服务质量路由算法.计算机研究与发展,2004,41(8):1368-1375.
- [10] 王小平,曹立明.遗传算法——理论、应用与软件实现.西安:西安交通大学出版社,2002.74-76.
- [11] 崔逊学,林闯.基于多目标遗传算法的多播服务质量路由优化.计算机研究与发展,2004,41(7):1144-1150.
- [12] 陈国良,王煦法.遗传算法及其应用.北京:人民邮电出版社,1996.



刘书雷(1979 -),男,河南邓州人,博士生,主要研究领域为服务聚合, workflow 技术.



唐桂芬(1977 -),女,博士生,主要研究领域为空间数据集成技术.



刘云翔(1978 -),男,博士生,CCF 高级会员,主要研究领域为空间信息网格技术,数据库技术.



景宁(1963 -),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为地理信息系统,数据库技术.



张帆(1977 -),男,博士生,主要研究领域为多目标决策技术.