

杨雪薇, 江凌云, 李研. 基于 NSGA-II 的自适应权值物联网服务组合方法[J]. 智能计算机与应用, 2024, 14(6): 1-10. DOI: 10.20169/j.issn.2095-2163.240601

# 基于 NSGA-II 的自适应权值物联网服务组合方法

杨雪薇<sup>1</sup>, 江凌云<sup>1,2</sup>, 李研<sup>3</sup>

(1 南京邮电大学 通信与信息工程学院, 南京 210003; 2 南京邮电大学 物联网研究院, 南京 210003;  
3 联通物联网有限责任公司, 南京 210003)

**摘要:** 由于网络服务种类繁多, 用户需求复杂多变, 因此在处理面向用户需求的多目标服务组合任务时, 根据用户需求偏好对多目标  $QoS$  属性进行加权处理得到的初始化目标函数, 在子目标的权值处理上会产生一定的偏差, 这可能使多目标优化算法在求解服务组合问题时得到的解集不准确, 出现不满足用户实际需求的情况。所以本文提出了一种面向用户需求的多目标自适应权值服务组合方法。该方法根据迭代过程中子目标均值与目标期待值之间的相对偏差值, 动态修正加权系数。通过权值的修正使构建的目标函数更符合用户需求偏好, 同时避免子目标陷入局部最优的困境。仿真结果表明, 改进后的算法在帕累托前沿上有明显的优势, 避免了局部最优, 且时间复杂度远低于其他算法。随着迭代过程中权值的修正, 目标函数均值也逐步贴近期待值, 有效地满足了用户的需求。

**关键词:** 多目标优化; 自适应权值; 服务组合;  $QoS$

中图分类号: TP393.0

文献标志码: A

文章编号: 2095-2163(2024)06-0001-10

## An adaptive weighting service composition method for Internet of Things based on NSGA-II

YANG Xuewei<sup>1</sup>, JIANG Lingyun<sup>1,2</sup>, LI Yan<sup>3</sup>

(1 School of Communications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; 2 Internet of Things Research Institute, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; 3 Unicom Internet of Things Co., Ltd., Nanjing 210003, China)

**Abstract:** Due to the wide variety of network services and the complex and changeable user demands, when dealing with the multi-objective service combination task oriented to user demands, the initial objective function obtained by weighting the multi-objective  $QoS$  attributes according to user demand preferences will produce certain deviations in the weight processing of sub-object. This may cause the solution set obtained by the multi-objective optimization algorithm to be inaccurate when solving the service composition problem, which may not meet the actual needs of users. Therefore, this paper proposes a multi-objective adaptive weight service composition method for user requirements. According to the relative deviation between the sub-target mean value and the target expected value in the iterative process, the weighted coefficient is dynamically modified. By modifying the weight value, the constructed objective function is more in line with the user's demand preference, and at the same time, the sub-objective is avoided to fall into the dilemma of local optimization. Simulation results show that the improved algorithm has obvious advantages on the Pareto frontier, avoids local optimization, and the time complexity is much lower than other algorithms. With the weight correction in the iterative process, the mean value of the objective function is also gradually close to the expected value, which effectively meets the needs of users.

**Key words:** multi-objective optimization; adaptive weight; service composition;  $QoS$

## 0 引言

随着物联网、云计算、边缘计算等技术的高速发

展, 单一的服务已经不能满足人们的需求, 面向用户的多服务组合方案应运而生。与此同时, 越来越多的应用以服务的形式发布与使用, 而这些服务大多

**基金项目:** 江苏省重点研发项目 (BE2020084-4)。

**作者简介:** 杨雪薇 (1999-), 女, 硕士研究生, 主要研究方向: 下一代网络技术, 物联网技术; 李研 (1971-), 男, 教授级高工, 主要研究方向: 5G, 物联网, 云计算。

**通讯作者:** 江凌云 (1971-), 女, 副教授, 硕士生导师, 主要研究方向: 下一代网络。Email: jiangly@njupt.edu.cn

收稿日期: 2023-04-12

具有不同的服务质量。例如,2个提供相同测温功能的传感器,在响应时间和准确性上可能存在差距。加之用户需求有着多样化、动态化和复杂化等特点,因此如何从大量的服务中选出满足用户需求的服务进行组合,成为服务供应中的重要问题。

从海量服务中选取满足用户需求的服务进行组合是一种服务优化任务,而用户需求的复杂性和多样性也使这种服务组合问题成为了一种多目标优化问题。例如,随着现代技术的发展,无人驾驶技术逐渐成熟,辅助停车系统作为其中一环也给人们生活出行带来了便利。当人们使用辅助停车系统时,系统根据驾驶者发出的指令,通过分析处理各传感器提供的服务和数据,如车身周围人流、车流情况、车与车位的具体位置距离、车辆自身状态等,选择符合指令的服务模块来控制车辆移动,完成辅助停车任务。在此次停车任务中,系统选择的服务在组合后需要同时满足安全性和准确性等多项性能目标,而待选择的服务功能相似,但性能上可能有较大差异。此时,服务组合问题就可以抽象为一个多目标优化问题,如何在功能相近的服务中,选择满足用户 QoS 的服务完成组合是优化问题的关键。

目前对服务组合优化的研究大多集中在从总体上改进算法性能<sup>[1]</sup>或通过聚类<sup>[2]</sup>等方法减少搜索空间,加快收敛等,而对于适应度函数的构建往往被忽略。但在初始化服务组合问题时,通过抽象用户需求构建的多目标适应度函数模型与实际的用户需求存在一定的偏差<sup>[3]</sup>,这会使优化算法得到的解集准确性降低,不满足用户的实际需求。此外,很多研究在处理 QoS 参数时,不考虑 QoS 参数之间的关联性<sup>[4]</sup>,仅通过线性加权的方法将多目标问题直接转变为单目标问题<sup>[5]</sup>。虽然降低了多目标维度,但单目标优化只能在总体服务质量上说明服务组合结果较好,并不能保证每一个 QoS 参数都满足用户需求,容易陷入 QoS 指标局部最优的情况。

综上所述,本文从用户需求的角度出发,提出一种基于 NSGA-II 的多目标动态加权算法。在优化过程中自适应地调整权值来修正抽象产生的偏差,提高优化算法解集的准确性;并考虑 QoS 参数之间的关联性,对 QoS 参数进行分组处理,降低目标维度,避免 QoS 目标陷入局部最优的困境。

## 1 相关工作

目前国内外学者对服务组合问题的研究主要集中在以下几个方面<sup>[4]</sup>:第一,采用服务聚类技术提

高算法性能;第二,提出混合式启发算法最大化算法优点,最小化缺点;第三,结合适应度函数和一些其他技术提高算法性能。

在服务聚类技术方向,文献[6]提出了一种上下文感知相似性(CAS)方法。文献[7]提出了一种基于词向量和 Biterm 主题模型的 Web 服务聚类方法。文献[8]提出了一种根据功能相似性筛选 Web 服务的方法来引导 Web 服务聚类。上述文献都通过服务聚类的技术来减小搜索空间,以提高算法的性能。

在混合式启发算法方向,文献[9]通过引进神经网络中的激活函数和一个能综合描述种群多样性以及进化代数的变量来对传统的遗传算法进行改进,能有效避免早熟收敛问题,产生适应度更高的个体。文献[10]引入了一种混合算法,将蚁群优化(ACO)和遗传算法(GA)相结合,在云上高效地完成组合服务。文献[11]提出了一种新的混合多目标进化算法 ADE-NSGA-II。自适应微分进化(ADE)算法使用自适应变异和交叉算子代替非支配排序遗传算法(NSGA-II),在解空间中更有效地搜索最优解,保证多样性。上述文献通过改进一种启发式算法,将其与另一种算法相结合,以最小化其缺点,最大化其优点,但这类方法复杂度高,导致时间复杂度也较高。

在结合算法适应度函数和一些其他技术提高算法性能方向,文献[12]提出了一种基于 QoS 属性和权重的模糊多目标遗传算法(FMOGA)来实现云服务组合,解决了多属性决策方法无法对众多候选服务进行全局优化的问题。文献[13]考虑到 QoS 参数之间的关联,基于自适应基因型进化的遗传算法(AGEGA),提出了一种新的最优适应感知算法。就是在平衡 QoS 参数的同时,也考虑到服务组合的连通性约束。上述文献考虑到 QoS 参数之间的关联性,通过对权重的处理或自适应的感知方法对多目标服务组合问题进行全局优化。

本文的研究从第2个方向出发,对多目标优化算法 NSGA-II 进行改进,采用混合式启发算法。同时考虑第3个方向中适应度函数的构建问题,修正抽象用户需求时产生的偏差。

非支配遗传算法(NSGA-II)作为一种经典的多目标全局优化方法,利用快速的非劣性排序方法和精英策略选择,并通过定义拥挤距离保持种群的多样性,以得到非劣解集。本文在使用 NSGA-II 算法解决组合优化问题时,将用户需求偏好抽象为多

目标函数的权重,同时考虑多个  $QoS$  参数之间的关联性,结合传统的加权法,利用加权系数将多个  $QoS$  参数合并为不同的子目标函数,来降低目标维度。文献[14]-[16]通过不同的方法来确定多目标优化问题的权重,本文对权重采用自适应的调整方法,可以使多目标优化结果更贴近用户需求,以此来修正抽象产生的偏差。

## 2 问题模型与定义

### 2.1 服务组合描述

服务组合就是将供应商提供的服务按照用户需求,从候选服务中选出具有相应功能的服务组合起来。服务的  $QoS$  指标可以向用户反映服务质量水平,也可以反映用户对服务的满意程度和服务组合结果与用户需求之间的差距。

由于用户需求的复杂性,单一的服务将无法满足其需求。因此需要分析用户需求,从不同的服务集中选取服务完成组合,服务组合流程图如图 1 所示。

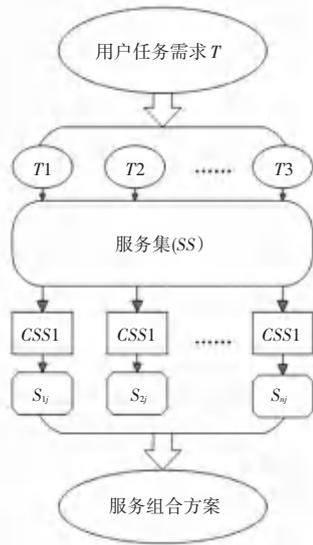


图 1 服务组合流程图

Fig. 1 Flow chart of service composition

其中,  $SS$  为供应商提供的服务集,  $CSS_i = \{S_{i1}, S_{i2}, \dots, S_{ij}, \dots, S_{in}\}$  为根据用户需求对应的第  $i$  个候选服务集,  $S_{ij}$  为从第  $i$  个候选服务集中选取的第  $j$  个服务。研究中,每一个服务  $S_{ij}$  对应  $QoS$  属性  $Q = \{Q_1, Q_2, \dots, Q_a, \dots, Q_m\}$ ,  $QoS$  模型如图 2 所示。

### 2.2 $QoS$ 参数的处理

服务的  $QoS$  属性包括很多方面,例如响应时间、可靠性、价格、可用性、声誉、安全性等等。文中将做阐释分述如下。

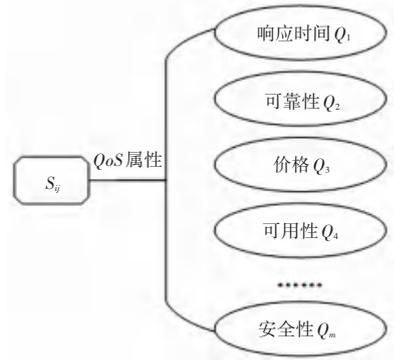


图 2  $QoS$  属性图

Fig. 2  $QoS$  attribute graph

- (1) 响应时间:指提交请求和服务响应之间的时间间隔。
- (2) 可用性:一个服务在单位时间内可以被访问的概率。
- (3) 可靠性:一个服务在单位时间内服务请求被正确执行的概率。
- (4) 价格:指服务请求者在访问某一服务时需要支付一定的服务使用费用。
- (5) 声誉:客户对服务的认可程度。
- (6) 安全性:所提供的服务是否可信、完整、真实。

像价格和响应时间这类数值越大对应服务质量越差的属性,称为成本型  $QoS$  指标。而可靠性和安全性这类数值越大对应服务质量越好的属性,则称为效益型  $QoS$  指标<sup>[5]</sup>。

不同供应商提供的服务的  $QoS$  度量方法可能存在差异,导致相互之间无法进行比较。因此需要对不同服务的  $QoS$  值进行归一化处理使其能够在同一度量环境下进行比较计算。利用式(1)将  $QoS$  属性按成本型和效益型进行分类计算,使数值映射在  $[0, 1]$  之间:

$$q = \begin{cases} \frac{Q_{\max} - Q_x}{Q_{\max} - Q_{\min}}, & \text{当 } Q_x \text{ 为效益型且 } Q_{\max} \neq Q_{\min} \\ \frac{Q_x - Q_{\min}}{Q_{\max} - Q_{\min}}, & \text{当 } Q_x \text{ 为成本型且 } Q_{\max} \neq Q_{\min} \\ 0, & Q_{\max} = Q_{\min} \end{cases} \quad (1)$$

其中,  $Q_x$  为服务  $S_{ij}$  中第  $x$  个要归一化处理的  $QoS$  属性值;  $Q_{\max}$  为该候选服务集  $CSS_i$  中对应属性值最大的  $QoS$  值;  $Q_{\min}$  为该候选服务集  $CSS_i$  中对应属性值最小的  $QoS$  值;  $q = \{q_1, q_2, q_3, \dots, q_m\}$  为其归

一化处理后的  $QoS$  属性值。

服务组合按照一定的逻辑将单个服务组合成具有某种功能的复合服务。这些服务之间的关系大多数情况下不是单纯的线性关系,不同的  $QoS$  属性在

不同的服务组合逻辑顺序下得出的值也不相同。常见的服务组合有4种基本结构:顺序结构、循环结构、并行结构和选择结构,见表1。

表1 服务组合基本结构  
Table 1 Service composition infrastructure

组合模型	响应时间 (RT)	可用性 (A)	可靠性 (RE)	价格 (P)	声誉 (RD)
顺序结构	$\sum_{i=1}^n Q_i(RT)$	$\prod_{i=1}^n Q(A)$	$\prod_{i=1}^n Q(RE)$	$\sum_{i=1}^n Q_i(P)$	$\sum_{i=1}^n \frac{Q_i(RD)}{n}$
循环结构	$k \times Q_i(P)$	$(Q_i(A))^k$	$(Q_i(RE))^k$	$k \times Q_i(P)$	$Q_i(RD)$
并行结构	$\max\{Q_1(RT), \dots, Q_i(RT)\}$	$\prod_{i=1}^n Q(A)$	$\prod_{i=1}^n Q(RE)$	$\sum_{i=1}^n Q_i(P)$	$\sum_{i=1}^n Q_i(RD)$
选择结构	$\sum_{i=1}^n PQ_i(RT)$	$\sum_{i=1}^n PQ_i(A)$	$\sum_{i=1}^n PQ_i(RE)$	$\sum_{i=1}^n PQ_i(P)$	$\sum_{i=1}^n PQ_i(RD)$

### 2.3 目标函数的构建

基于  $QoS$  的方案在解决服务组合问题时,通过对用户的  $QoS$  需求进行分析,并考虑每一个  $QoS$  属性的实际意义,将其  $QoS$  需求分为多组。且每一组组内  $QoS$  之间不存在冲突关系,分组如下:

$$\begin{aligned} q_1 &= \{q_{11}, \dots, q_{1h}\} \in q, \\ q_2 &= \{q_{21}, \dots, q_{2s}\} \in q, \\ &\dots \\ q_m &= \{q_{m1}, \dots, q_{mu}\} \in q \end{aligned}$$

其中,

$$\begin{aligned} q_1 \cup q_2 \cup \dots \cup q_m &= q, \\ q_1 \cap q_2 \cap \dots \cap q_m &= \emptyset \end{aligned}$$

再对每一组  $QoS$  属性值采用加权方式进行计算用于构建目标函数,利用 NSGA-II 算法对多目标问题进行优化。

目标函数构建如下:

$$\begin{aligned} f_1 &= 1 - w_1 \times q_1^T, \quad i \text{ 为第一组 } QoS \text{ 属性} \\ f_2 &= 1 - w_2 \times q_2^T, \quad j \text{ 为第二组 } QoS \text{ 属性} \\ &\dots \\ f_m &= 1 - w_m \times q_m^T, \quad l \text{ 为第 } M \text{ 组 } QoS \text{ 属性} \end{aligned} \quad (2)$$

其中,  $w_1 = \{w_{11}, \dots, w_{1h}\}$ ,  $w_2 = \{w_{21}, \dots, w_{2s}\}, \dots$ ,  $w_m = \{w_{m1}, \dots, w_{mu}\}$ , 为每组  $QoS$  对应的加权系数,  $f_1, f_2, \dots, f_m$  为构建的目标函数。

## 3 算法设计

### 3.1 多目标遗传算法

遗传算法是模拟达尔文进化论的自然选择和遗

传学机理的生物进化过程的计算模型,是一种通过模拟自然进化过程搜索最优化的方法。多目标优化与单目标不同,多目标优化问题不存在一个确定的最优解,而是把所有可能的解称为非劣解,也称 Pareto 解。而多目标优化算法是使优化的方向沿着 Pareto 前沿移动,来找到更优的个体<sup>[17]</sup>。非支配遗传算法(NSGA-II)作为一种经典的多目标全局优化方法,使用快速的非劣性排序方法、精英策略选择,并通过定义拥挤距离保持种群的多样性,来求解多目标问题。NSGA-II 算法流程图如图3所示。

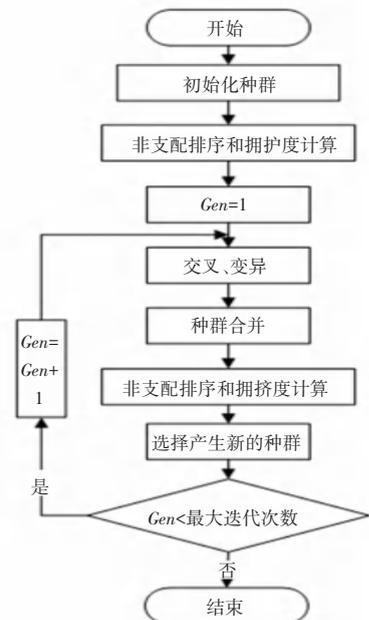


图3 NSGA-II 流程图

Fig. 3 NSGA-II flow chart

### 3.2 算法优化

对于固定的加权系数来说,由于在抽象用户需求构建多目标适应度函数时存在偏差,初始设置的加权值可能并不能使优化结果满足用户需求。并且在多目标问题中,可能出现某一子目标在优化时得到最优,但其他子目标优化结果较差的情况。具体到组合实际问题中,可能会出现最终优化结果在总目标上满足用户需求,但对于用户需求的某一 *QoS* 属性来说可能存在较大差距。

因此本文提出了一种动态调整权值的方法,根据构建的目标函数  $f_1, \dots, f_i, \dots, f_m$ , 计算每代种群中各个子目标的均值 *Fave*。第 *i* 个子目标在第 *k* 次迭代的均值为:

$$Fave_i^k = \frac{f_{i1}^k + f_{i2}^k + \dots + f_{in}^k}{n} \quad (3)$$

其中, *n* 为第 *k* 次迭代对应的种群个体数,  $f_{in}^k$  为第 *k* 次迭代中第 *n* 个个体的第 *i* 个子目标值。

计算均值和子目标期待值 *Fobj* 之间的相对偏差 *d* 时,求得的第 *i* 个子目标在第 *k* 次迭代的相对偏差为:

$$d_i = \frac{Fobj - Fave_i^k}{Fobj - Fave_i^0} \quad (4)$$

在算法优化过程中,为了使子目标同时趋近期待值,要求各个子目标的相对偏差同时趋近于 0 或者数值相同<sup>[18]</sup>。

目标的相对偏差值能够反映子目标和目标期待值之间的距离,根据目标偏差值动态地调整加权系数,使子目标更好地趋近期待值。利用以下公式,通过各个子目标的相对偏差值来控制其权值的变化:

$$w_x^k = w_x^{k-1} \quad (5)$$

$$w_i^k = w_i^{k-1} + (d_i - d_x) \Delta \quad (6)$$

其中,  $w_i^k$  为第 *i* 个子目标在第 *k* 次迭代时对应的权重值,子目标 *a* 为第 *k* 次迭代中相对偏差值  $d_i$  最小的子目标。

由于第 *x* 个子目标相对偏差值最小,说明这一子目标最贴近期待值。使相对偏差值最小的子目标对应权值保持不变,通过式(5)、式(6)修正其他子目标权值,在下一迭代中其他子目标权重将得到提高。

在多目标的服务组合问题中,由本文构建的目标函数可知,每个子目标对应多个 *QoS* 属性,每个 *QoS* 属性拥有对应的加权系数,即  $w_1 = \{w_{11}, \dots, w_{1h}\}$ ,  $w_2 = \{w_{21}, \dots, w_{2s}\}$ ,  $\dots$ ,  $w_m =$

$\{w_{m1}, \dots, w_{mu}\}$ , 为每组 *QoS* 对应的加权系数。每个子目标权值修正后,对应子目标中每个 *QoS* 属性的权值也需要修正,公式如下:

$$w_{ij}^k = \frac{w_{ij}^{k-1}}{\sum_{j=1}^m w_{ij}^{k-1}} \cdot w_i^k \quad (7)$$

$$w_i^k = \sum_{j=1}^l w_{ij}^k \quad (8)$$

其中, *i* 为第 *i* 个子目标; *j* 为第 *i* 个子目标对应的第 *j* 个 *QoS* 属性;  $w_{ij}^k$  为第 *k* 次迭代时第 *i* 个子目标中第 *j* 个 *QoS* 属性对应的权值;  $w_i^k$  为第 *k* 次迭代时第 *i* 个子目标中所有 *QoS* 属性对应权值的和。

通过各个子目标的相对偏差控制其权值的修正,再将修正后的子目标的权值按比例分配给每个 *QoS* 属性,最终完成多目标服务组合的动态加权修正,使优化结果在每个子目标上都更贴近用户需求。

但由于 NSGA-II 算法作为启发式算法在优化过程中需要一定的迭代次数才能得到较新的种群,若每次迭代都进行相对偏差值的判断会增加算法复杂度。因此在本算法中,每迭代 50 次进行一次相对偏差的计算和权值的调整,这样一来就使得种群个体得到进化,保证每次计算的相对偏差值得到趋近,也不增加算法复杂度。对应算法伪代码如下。

#### 算法 1 权值自适应函数 *SetWeight(F1, W)*

输入 最新个体集群 *F1*, 多目标适应函数的权值 *W*

输出 修正后的权值 *W'*

1. /\* 根据最新个体集群 *F1* 计算个体子目标均值 \*/

For *i* = 1 to *m*

/\* 由式(3)计算第 *i* 个子目标在本次迭代的均值 \*/

For *a* = 1 to *n*

$Fave_i^k \leftarrow F1$

End

/\* 由式(4)计算第 *i* 个子目标在本次迭代的相对偏差 \*/

$d_i \leftarrow Fave_i^k$

End

2. /\* 由式(5)对子目标权值进行自适应修正 \*/

If ( $d_x == \text{Max}(\text{abs}(d_i))$ )

$w_i^k \leftarrow d_i$

End

3. /\* 由式(6)对子目标中每个  $QoS$  属性的权值进行修正 \*/

For  $j = 1$  to  $t$

$$w_{ij}^k \leftarrow w_i^k$$

End

4. Return  $W = w_{ij}^k$

### 算法2 本文算法

输入 设置初始化参数。最大迭代次数  $Maxit$ , 种群大小  $nPop$ , 交叉概率  $pCrossover$ , 变异概率  $pMutation$

输出 Pareto 前沿解集

1.  $pop = []$  /\* 初始化  $pop$  存放支配关系的排序结果 \*/

$F1 = []$  /\* 存放本算法 Pareto 前沿解集 \*/

/\*

$W = []$  /\* 根据用户需求设置初始化权值 \*/

\*/

2. /\* 按式(2)构建多目标函数作为本算法的适应度函数,计算个体适应度值,用于非支配排序、拥挤度计算、种群排序、交叉变异等 \*/

$Pop.cost = function\ obj(W)$

$[pop, F] = NonDominatedSorting(pop);$

/\* 非支配排序 \*/

$Pop = CalcCrowdingDistance(pop, F);$

/\* 计算拥挤度 \*/

$[pop, F] = SortPopulation(pop);$

/\* 种群排序 \*/

/\* 通过以上处理得到初始化种群  $pop$  \*/

3. For  $it = 1$  to  $Maxit$

$popc$  /\* 交叉得到新的个体 \*/

$popm$  /\* 变异得到新的个体 \*/

$pop = [pop, popc, popm]$  /\* 种群合并 \*/

$[pop, F] = NonDominatedSorting(pop);$

/\* 对新的种群进行非支配排序 \*/

$Pop = CalcCrowdingDistance(pop, F);$

/\* 对新的种群计算拥挤度 \*/

$[pop, F] = SortPopulation(pop);$

/\* 选择产生新的种群 \*/

$F1 = pop(F\{1\})$  /\*  $F1$  为当前算法得到的非支配解 \*/

/\* 算法每迭代 50 代进行一次权值的修正 \*/

If ( $it\%50 == 0$ )

$SetWeight(F1, W)$

$Pop.cost = function\ obj(W)$  /\* 根据修正

的适应度函数重新计算种群个体的适应度值 \*/

End

End

4. Return  $F1$

根据以上步骤绘制流程如图4所示。

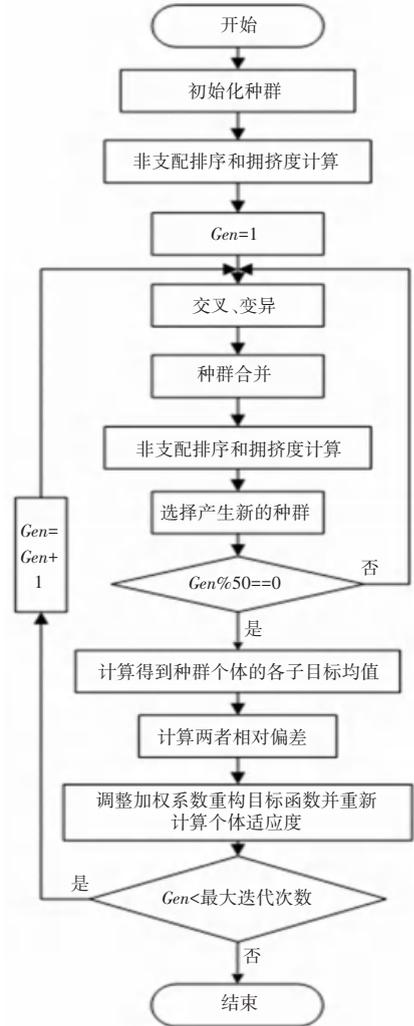


图4 本文算法流程图

Fig. 4 Flow chart of the algorithm in this paper

## 4 实验仿真

遗传算法作为经典的启发式算法通过 Matlab 仿真十分便捷,因此本文使用 Matlab 实现算法的仿真对比。实验平台的配置为:软件环境是 Microsoft Windows 11 家庭中文版(64位),版本号 Matlab R2021a;硬件环境是 Intel(R) Core(TM) i5-12500H。为了验证本文算法的性能,设计实验如下。

### 4.1 实验参数

本文生成 1 000 个服务,将其分为 5 个服务类,每个服务类中含有 200 个服务,每一个服务考虑 5

个  $QoS$  属性, 分别为价格、安全性、荣誉、可靠性和响应时间。对应  $QoS$  属性范围见表 2。

表 2  $QoS$  属性值生成范围

Table 2 Range of  $QoS$  attributes value generation

价格	安全性	荣誉	可靠性	响应时间
[100,500]	[0.1,0.9]	[0.2,0.8]	[0.3,0.7]	[0.1,4.0]

本次实验用户对  $QoS$  的主要需求是响应时间较小, 但安全性较高。而候选服务中安全性高的服务, 响应时间也较大, 这就使得用户的 2 个需求出现了冲突。

为了验证动态加权的多目标遗传算法, 对具有冲突属性的各子目标的优化情况, 从 5 类服务中各选取一个服务进行组合, 将其看作一个个体。将每个服务的 5 个  $QoS$  属性根据用户需求分为 2 组, 第一组中包含的  $QoS$  属性为安全性、价格和可靠性, 第二组为响应时间和荣誉。并对每个服务的各个  $QoS$  值采用式(1)进行归一化处理, 再利用式(2)构建目标函数将多目标变为双目标, 将构建的子目标函数作为个体的适应度函数用于组合优化。

由于权值的偏差是在抽象用户需求的过程中产生的, 与算法的优化过程无关。因此在计算目标期待值时, 通过构建目标函数, 设置固定的权值不对其进行更新, 利用 NSGA-II 算法根据随机生成的  $QoS$  数据, 多次求解多目标服务组合问题, 将优化结果的均值作为目标期待值。在验证本文算法时, 使初始权值与计算目标期待值时设置的固定权值不同, 将两者的差值作为抽象用户需求的过程中产生的偏差, 通过算法予以修正。其他算法的权值也设置为相同的初始权值来对比分析。

#### 4.2 评价指标

多目标优化算法与单目标不同, 多目标算法在优化结果上得到的是 Pareto 最优解, 是一系列解集。在对比实验结果时无法直观地进行比较, 因此本文采用 3 个指标来评价算法的性能。文中给出综合论述如下。

(1) 数量指标<sup>[19]</sup> (Number of Pareto Solutions,  $NPS$ )。该指标表示算法求得的 Pareto 解集中解的数量。该指标的值越大, 表明用户可以选择的方案越多, 算法的种群多样性更好。

(2) 欧式距离<sup>[20]</sup> (Euclidean Distance,  $ED$ )。该指标表示  $n$  维空间里点与点之间的距离。可以用于计算解集中某一个体对应的粒子与目标值对应的粒

子间的距离大小, 值越小, 表明两者越贴近, 个体越靠近目标值。

$n$  维欧氏空间是一个点集, 空间中的每个点可以表示为  $(x(1), x(2), \dots, x(n))$ , 其中  $x(i) (i = 1, 2, \dots, n)$  是实数称为  $x$  的第  $i$  个坐标, 2 个点  $x$  和  $y$  之间的距离  $ED(x, y)$  定义公式如下:

$$ED = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (9)$$

将目标期待值表示为  $x = (f_1', f_2')$ , 算法的多目标优化结果表示为  $y = (f_1, f_2)$ , 由以上公式计算欧氏距离  $ED$ 。

(3) 运行时间 ( $Time$ )。该指标表示算法每次运行所需的实际时间 (单位: s)。可以用于衡量算法的复杂性。运行时间越小, 说明算法优化速度越快, 算法自身复杂性越低。

#### 4.3 实验结果分析

##### 4.3.1 本文算法结果分析

利用随机生成的  $QoS$  数据, 根据改进后的流程图算法进行组合优化, 设置交叉概率为 0.7, 突变概率为 0.4, 最大迭代次数为 1 000 代, 种群大小为 50, 设置每个  $QoS$  属性初始固定加权值都为 0.2。根据用户需求设子目标  $f_1$  的用户期待值为 0.65, 子目标  $f_2$  的用户期待值为 0.15。其中, 横坐标为  $f_1$  的目标函数值, 纵坐标为  $f_2$  的目标函数值。当算法每计算 50 代, 根据式(3)~(6)修正权值, 更新适应度函数继续优化, 重复此过程, 直到迭代结束。仿真结果详见如下。

图 5 为本文算法优化得到的 Pareto 最优解。由帕累托前沿图可看出, 本文算法的解集分布均匀且解集数量较多。

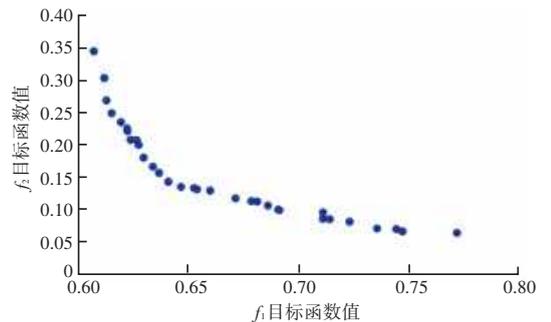


图 5 本文算法优化结果

Fig. 5 Optimization results of the algorithm in this paper

改进后的算法每迭代 50 代根据式(3)~(8)对权值进行修正。图 6 为每 100 代即修正 2 次的权值折线图。对应图 7 中种群适应度均值与目标期待值

的欧氏距离,可以看出随着权值的修正,两者间的欧式距离逐渐减小,种群个体的均值越来越贴近目标期待值,说明权值的修正使重新构建的目标函数更符合用户需求偏好。

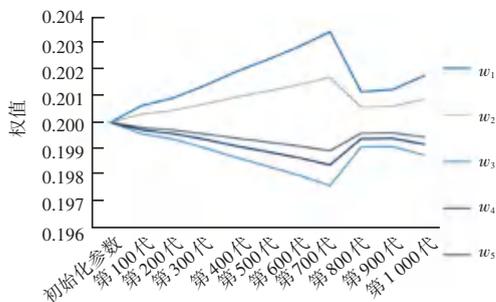


图6 每100代QoS属性对应权值

Fig. 6 Weight of QoS attributes in every 100 generations

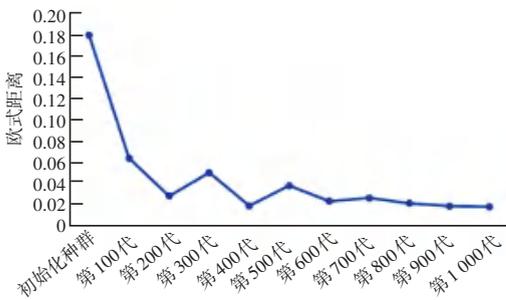


图7 种群适应度均值与目标期待值的欧氏距离

Fig. 7 Euclidean distance between population fitness mean and target expectation value

#### 4.3.2 最新算法对比分析

为了评价本文算法的性能,用本文算法和NSGA-II、MOHHO、MOGWO算法对相同的问题实例进行求解,分别将迭代次数设为400,800进行优化,得到帕累托前沿解。对NPS、ED和(时间复杂度)三个评价指标取多次优化的均值进行对比分析,对应参数见表3。

表3 4种算法参数设置

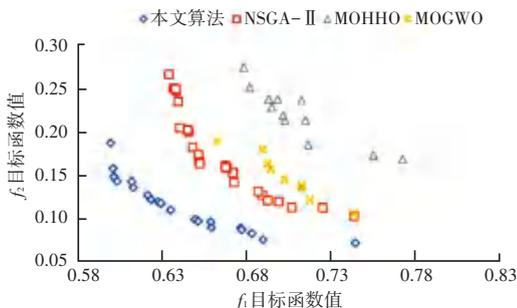
Table 3 Four algorithms parameters settings

算法名称	迭代次数	种群大小
本文算法	400,800	50
NSGA-II	400,800	50
MOHHO	400,800	50
MOGWO	400,800	50

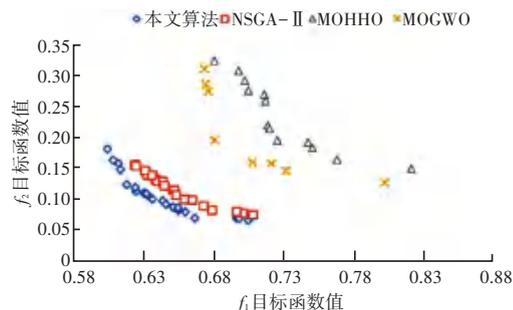
表3中,NSGA-II为本文改进前的算法,文献[21]对哈里斯鹰算法优化,在大规模场景下求解服务组合问题。文献[5]采用灰狼算法求解服务组合问题。因此本文选取这2种启发式算法对应的多目

标优化算法MOHHO,MOGWO来进行对比分析。

4种算法在2种迭代次数下的帕累托前沿解集如图8所示。从帕累托前沿图分析,由图8可以看出本文算法的帕累托前沿解集在分布上明显优于其他3种算法,且收敛性优于其他3种算法,没有陷入局部最优的困境。



(a) 400代



(b) 800代

图8 4种算法在2种迭代次数下的帕累托前沿解集

Fig. 8 The Pareto frontier solution set of four algorithms in two iterations

4种算法在2种迭代次数下的NPS值如图9所示。根据图9从NPS指标上分析,可以看到在最大迭代次数为400时,本文算法的NPS明显高于其他3种算法,说明此时用户可以选择的方案越多,本文算法的种群多样性更好。但最大迭代次数为800时,本文算法和NSGA-II算法的NPS都较高,说明这2种算法都有较好的种群多样性,但结合图9中可知本文算法有更好的帕累托前沿解集。

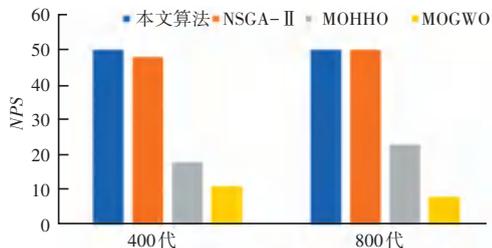


图9 4种算法在2种迭代次数下的NPS值

Fig. 9 NPS values of the four algorithms in two iterations

4 种算法均值与目标期待值的欧氏距离如图 10 所示。根据图 10 从算法优化过程中的适应度均值和目标期待值之间的欧氏距离分析可知,在 4 种算法中,本文算法随着权值的修正,每代适应度均值和目标期待值之间的欧氏距离逐渐减小,且欧式距离值最小。说明本文算法的优化结果更好地贴近目标期待值,有效地满足了用户的需求。

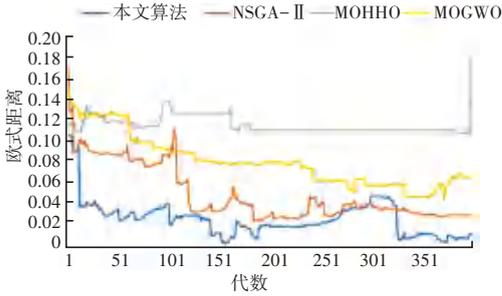


图 10 4 种算法均值与目标期待值的欧氏距离

Fig. 10 Euclidean distance between the mean value of the four algorithms and the expected value of the target

3 种算法不同最大迭代次数对应运行时间对比见表 4。由表 4 可以看出,与其它 2 种启发式算法相比,本文算法的运行时间在所有最大迭代次数下都是最短的,说明本文算法可以更快地找到帕累托前沿解,在时间复杂度上有更好的表现。

表 4 3 种算法不同最大迭代次数对应运行时间

Table 4 Corresponded running time of different maximum iterations of three algorithms

最大迭代次数/代	本文算法	MOHHO	MOGWO
200	<b>14.159</b>	16.493	19.235
400	<b>28.068</b>	33.189	38.578
500	<b>35.114</b>	41.220	47.757
600	<b>41.988</b>	49.423	57.205
800	<b>57.452</b>	66.232	75.645
1 000	<b>70.030</b>	81.163	93.226

综上所述,可以看出根据子目标均值与目标期待值之间的相对偏差值,动态调整加权系数,可以有效地修正初始化过程中抽象用户需求产生的偏差,使优化结果快速贴近目标期待值,满足用户需求,同时与其他算法相比本文算法在种群多样性、计算时间等方面都有较为优秀的表现。

### 5 结束语

在服务组合问题中,根据用户需求构建准确的目标函数,来保证优化结果的准确性是十分必要的,本文针对基于 *QoS* 的多目标服务组合问题,通过对

现有的启发式算法进行分析,发现采用传统的加权法和启发式算法相结合,并考虑 *QoS* 参数之间的关联性,将多个 *QoS* 参数合并为不同的子目标函数,可以在降低多目标维度的同时,更好地完成服务的组合优化。但在多目标优化时,采用传统的固定加权算法,并不能确定算法在初始化时设置的加权系数是最优的。基于此,本文设计自适应加权方法,根据子目标均值与期待值之间的相对偏差,动态调整各个目标的加权系数,修正多目标函数,使其更贴近用户需求,保证每个子目标都能获得较优解。

仿真实验通过计算种群适应度均值与目标期待值的欧氏距离,证明了随着权值的调整两者间的欧氏距离不断减小,优化结果更好地满足了用户需求,验证了算法的可行性和对目标函数的正确修正。实验结果还表明相比其他启发式算法,本文算法的在种群多样性上表现优秀,且可以更迅速地找到贴合用户需求的服务组合方法。但启发式算法具有不稳定性,在这一方面本文算法有待进一步改进。

### 参考文献

- [1] KHADIR K, GUERMOICHE N, GUITTOUM A, et al. A Genetic Algorithm-Based approach for fluctuating QoS aware selection of IoT services[J]. IEEE Access, 2022(10): 17946-17965.
- [2] PARIMALAM T, MEENAKSHI S K. Efficient clustering techniques for Web services clustering [C]//IEEE International Conference on Computational Intelligence and Computing Research (ICCCIC). Coimbatore, India :IEEE,2017: 1-4.
- [3] NAGHAVIPOUR H, SOON T K, IDRIS M Y I B, et al. Hybrid metaheuristics for QoS-aware service composition: a systematic mapping study[J]. IEEE Access, 2022(10): 12678-12701.
- [4] JATOTH C, GANGADHARAN G R, BUYYA R. Optimal fitness aware cloud service composition using an adaptive genotypes evolution based Genetic Algorithm [J]. Future Generation Computer Systems,2019(94):185-198.
- [5] 徐雪敏,张秀国,肖媛元,等. 基于优化的灰狼算法的大规模 Web 服务组合[J]. 计算机应用, 2022, 42(10):3162-3169.
- [6] KUMARA K, BANAGE T G S, PAIK I, et al. Context-Aware Web service clustering and visualization[J]. International Journal of Web Services Research (IJWSR),2020,17(4): 32-54.
- [7] YANG Dongju, HE Dan. Web service clustering method based on word vector and biterm topic model [C]// 2021 IEEE 6<sup>th</sup> International Conference on Cloud Computing and Big Data Analytics (ICCCBDA). Chengdu, China :IEEE,2021:299-304.
- [8] KUMARA B T G S, PAIK I, OHASHI H, et al. Web service filtering and visualization with context aware similarity to bootstrap clustering[C]//2013 International Joint Conference on Awareness Science and Technology & Ubi - Media Computing. Aizu - Wakamatsu, Japan;IEEE, 2013:220-226.
- [9] LIU Feng, HAN Min, LIU Jianwei. Credible Web Service Composition based on Improved multi-objective particle swarm optimization [C] //Proceedings of 40<sup>th</sup> Chinese Control

- Conference (CCC). Shanghai, China ;IEEE,2021; 2408–2413.
- [10] DAHAN F, BINSAEEDAN W, ALTAF M, et al. An efficient hybrid metaheuristic algorithm for QoS – Aware cloud service composition problem[J]. IEEE Access, 2021(9): 95208–95217.
- [11] LIU Li, GU Shuxian, ZHANG Miao, et al. A hybrid evolutionary algorithm for inter – cloud service composition [ C ]//9<sup>th</sup> International Conference on Modelling, Identification and Control (ICMIC). Kunming, China ;IEEE,2017;482–487.
- [12] FENG Jianzhou, KONG Lingfu. A fuzzy multi-objective genetic algorithm for QoS-based cloud service composition [ C ]// 11<sup>th</sup> International Conference on Semantics, Knowledge and Grids (SKG). Beijing, China ;IEEE,2015; 202–206.
- [13] JATOTH C, GANGADHARAN G R, BUYYYA R. Optimal fitness aware cloud service composition using an adaptive genotypes evolution based genetic algorithm[J]. Future Generation Computer Systems, 2019(94): 185–198.
- [14] ZHANG Hongxia, ZHANG Huajun. A new method for objective weights computing in multi-objective optimization [ C ]// 9<sup>th</sup> IEEE Conference on Industrial Electronics and Applications. Hangzhou, China;IEEE, 2014; 2019–2022.
- [15] RYU N, SEO M, MIN S. Multiobjective optimization of the Benchmark TEAM problem using gradient – based approach and adaptive weight determination [ J ]. IEEE Transactions on Magnetics, 2020, 56(3): 1–4.
- [16] RYU N, LIM S, MIN S, et al. Multi-objective optimization of magnetic actuator design using adaptive weight determination scheme [ J ]. IEEE Transactions on Magnetics, 2017, 53(6): 1–4.
- [17] 张岩, 吴水根. MATLAB 优化算法 [ M ]. 北京:清华大学出版社, 2017.
- [18] 柳焯. 最优化原理及其在电力系统中的应用 [ M ]. 北京:清华大学出版社, 2002.
- [19] SCHAFFER J D. Multiple objective optimization with vector evaluated genetic algorithms [ C ]// Proceedings of the 1<sup>st</sup> International Conference on Genetic Algorithms. Mahwah ,USA: dblp, 1985:93–100.
- [20] 周成龙, 陈玉明, 朱益冬. 粒 K 均值聚类算法 [ J ]. 计算机工程与应用, 2023, 59(13): 317–324.
- [21] LI Chengyang, LI Jun, CHEN Huiling. A meta-heuristic-based approach for Qos-aware service composition [ J ]. IEEE Access, 2020(8): 69579–69592.