



(12) 发明专利

(10) 授权公告号 CN 110659034 B

(45) 授权公告日 2022.09.20

(21) 申请号 201910904428.1

(22) 申请日 2019.09.24

(65) 同一申请的已公布的文献号  
申请公布号 CN 110659034 A

(43) 申请公布日 2020.01.07

(73) 专利权人 合肥工业大学  
地址 230001 安徽省合肥市包河区屯溪路  
193号

(72) 发明人 梁昌勇 徐健 顾东晓 赵树平  
陆文星 蒋丽 董骏峰 王彬有  
马一鸣 张驰 邵建芳 王颖  
闫晓涵

(74) 专利代理机构 北京润平知识产权代理有限公司 11283  
专利代理师 肖冰滨 刘兵

(51) Int. Cl.  
G06F 8/60 (2018.01)

(56) 对比文件

CN 110210806 A, 2019.09.06

CN 109684075 A, 2019.04.26

CN 109240768 A, 2019.01.18

CN 108304168 A, 2018.07.20

US 2010228819 A1, 2010.09.09

US 2019020657 A1, 2019.01.17

Bing Lin. A Time-Driven Data Placement Strategy for a Scientific Workflow Combining Edge Computing and Cloud Computing.《IEEE Transactions on Industrial Informatics》.2019,

束东等. 基于信任的云服务系统多目标任务分配模型.《计算机研究与发展》.2018, (第06期),

曹云梦等. 面向服务的云-端动态协作方法.《计算机工程与科学》.2019, (第04期),

审查员 黄晓亮

权利要求书4页 说明书15页 附图3页

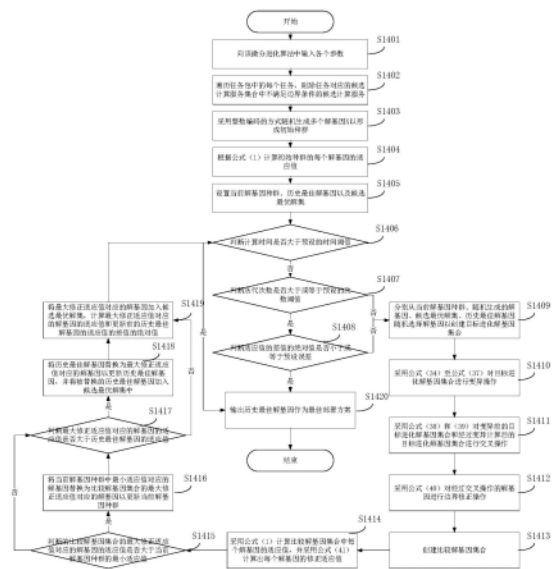
(54) 发明名称

云边混合计算服务的组合优化部署方法、系统及存储介质

(57) 摘要

本发明实施方式提供一种云边混合计算服务的组合优化部署方法、系统及存储介质,属于云边混合计算服务的优化部署技术领域。该云边混合计算服务的组合优化部署方法、系统及存储介质通过结合包括服务质量表现以外的云端服务满足和服务一致性等多种因素,采用改进的微分进化算法确定云边混合计算服务的最佳部署方案,解决了现有技术中在确定方案时由于仅结合单一的服务质量的因素而导致部署方案无法达到最优的技术问题,使得最终形成的部署方案能够最大程度地满足云边混合计算服务的要求。

CN 110659034 B



1. 一种云边混合计算服务的组合优化部署方法,其特征在于,所述组合优化部署方法包括:

获取所述云边混合计算服务的任务包,其中,所述任务包包括需要采用边缘计算服务完成的普通任务集合和需要采用云计算服务的特殊任务集合,所述普通任务集合和所述特殊任务集合均包括至少一个任务,每个所述任务之间通过Petri网结构连接;

确定能够完成每个所述任务的候选计算服务集合,其中,每个任务对应有用于完成所述任务的候选计算服务集合,每个所述候选计算服务集合包括至少一个候选计算服务,每个所述候选计算服务包括第一特征属性和第二特征属性,所述第一特征属性包括所述候选计算服务的类型以及用于完成所述候选计算服务的设备的品牌,所述第二特征属性包括完成所述候选计算服务的服务质量;

采用公式(1)作为预设的目标函数,

$$\text{Fit}(X) = \omega_1 \cdot \text{TQoS} + \omega_2 \cdot \text{TSR} + \omega_3 \cdot \text{TSC}, \quad (1)$$

其中,Fit(X)为部署方案X的所述目标函数, $\omega_1$ 、 $\omega_2$ 和 $\omega_3$ 为预设的权重,TQoS为总服务质量QoS表现值,TSR为采用公式(2)计算的特别计算服务满足程度值,

$$\text{TSR} = \left( e^{\ln 2 \times \frac{X_{\text{SRST}}}{N_{\text{SRST}}}} - 1 \right)^{\gamma_{\text{SRST}}}, \quad (2)$$

其中,TSR为所述特别计算服务满足程度值, $N_{\text{SRST}}$ 为所述特别任务集合包括的任务的数量, $X_{\text{SRST}}$ 是用于完成所述任务包的部署方案能够完成所述特别任务集合中的任务的数量, $\gamma_{\text{SRST}}$ 是预设的积聚强度值,且 $\gamma_{\text{SRST}} \geq 1$ ;TSC为采用公式(3)计算的服务一致性程度值,

$$\text{TSC} = \frac{\ln n + \sum_b \frac{X_{B=b}}{n} \ln \frac{X_{B=b}}{n}}{\ln n}, \quad (3)$$

其中,TSC为所述服务一致性程度值, $X_{B=b}$ 为在用于完成所述任务包的部署方案中,选择了品牌b的候选计算服务的数量,n为所述任务包包括的任务的数量;

采用微分进化算法根据所述任务包、所述候选计算服务集合和目标函数确定完成所述任务包的最佳部署方案。

2. 根据权利要求1所述的组合优化部署方法,其特征在于,所述组合优化部署方法进一步包括:

预设边界条件;

所述采用微分进化算法根据所述任务包、所述候选计算服务集合和目标函数确定完成所述任务包的最佳部署方案包括:

根据所述任务包、所述候选计算服务集合、目标函数以及所述边界条件确定完成所述任务包的最佳部署方案。

3. 根据权利要求2所述的组合优化部署方法,其特征在于,所述边界条件包括公式(4)和公式(5),

$$SQ_{\text{Ind},T_i} \geq \overline{SQ}_{\text{Ind},T_i}, \quad (4)$$

$$SQ_{\text{Ind},P} \geq \overline{SQ}_{\text{Ind},P}, \quad (5)$$

其中, $SQ_{\text{Ind},T_i}$ 为完成任务 $T_i$ 时指标Ind的标准化指标值, $\overline{SQ}_{\text{Ind},T_i}$ 为完成任务 $T_i$ 时指标

Ind的最低要求的标准化指标值,  $SQ_{Ind,P}$ 为完成任务包P时指标Ind的标准化指标值,  $\overline{SQ}_{Ind,P}$ 为完成任务包P时指标Ind的最低要求的标准化指标值。

4. 根据权利要求1所述的组合优化部署方法, 其特征在于, 所述组合优化部署方法进一步包括:

根据公式(6)计算所述总服务质量QoS表现值,

$$TQoS = \sum_{Ind} (\omega_{Ind} / \text{sum}(\omega_{Ind})) SQ_{Ind,P}, \quad (6)$$

其中,  $\omega_{Ind}$ 为所述候选计算服务的服务质量的指标Ind的权重,  $SQ_{Ind,P}$ 为完成任务包P时指标Ind的标准化指标值。

5. 根据权利要求2所述的组合优化部署方法, 其特征在于, 所述根据所述任务包、所述候选计算服务集合、目标函数以及所述边界条件确定完成所述任务包的最佳部署方案包括:

遍历所述任务包中的每个所述任务, 剔除所述任务对应的候选计算服务集合中不满足所述边界条件的所述候选计算服务;

采用整数编码的方式随机生成多个解基因X以形成初始种群, 其中,  $X = \{S_i^* | i = 1, \dots, n\}$ ,  $S_i^*$ 为完成任务 $T_i$ 所选择的候选计算服务;

根据公式(1)计算所述初始种群的每个所述解基因的适应值;

设置当前解基因种群、历史最佳解基因以及候选最优解集, 其中, 初始的所述当前解基因种群为所述初始种群, 所述历史最佳解基因为所述当前解基因种群中适应值最大的解基因, 所述候选最优解集初始为空集;

判断迭代次数是否大于或等于预设的次数阈值;

在判断所述迭代次数大于或等于所述次数阈值的情况下, 判断适应值的差值的绝对值是否小于或等于预设误差;

在判断所述绝对值大于所述预设误差和/或所述迭代次数小于所述次数阈值的情况下, 分别从所述当前解基因种群、随机生成的解基因、所述候选最优解集、所述历史最佳解基因随机选择解基因以创建目标进化解基因集合;

采用公式(7)至公式(10)对所述目标进化解基因集合进行变异操作,

$$U_1^{rand} = X^{rand} + F \cdot (X^{r1} - X^{r2}), \quad (7)$$

$$U_2^{rand} = X^{rand} + F \cdot (X^{r1} - X^{r2}) + F \cdot (X^{r3} - X^{r4}), \quad (8)$$

$$U_1^{best} = X^{best} + F \cdot (X^{r1} - X^{r2}), \quad (9)$$

$$U_2^{best} = X^{best} + F \cdot (X^{r1} - X^{r2}) + F \cdot (X^{r3} - X^{r4}), \quad (10)$$

其中,  $U_1^{rand}$ 、 $U_2^{rand}$ 、 $U_1^{best}$ 、 $U_2^{best}$ 为经过变异操作后的解基因,  $X^{rand}$ 为从所述当前解基因种群和随机生成的解基因中选择的解基因,  $X^{best}$ 为从所述候选最优解集和历史最佳解基因中随机选择的解基因,  $X^{r1}$ 、 $X^{r2}$ 、 $X^{r3}$ 、 $X^{r4}$ 为从当前解基因种群中随机选择的解基因;

采用公式(11)和(12)对变异前的所述目标进化解基因集合和经过变异计算后的所述目标进化解基因集合进行交叉操作,

$$v_i^{rand} = \begin{cases} u_i^{rand}, & \text{if } rand(0,1) \leq CR \text{ or } i = i_{rand}, \\ x_i^{rand}, & \text{otherwise} \end{cases}, \quad (11)$$

$$v_i^{best} = \begin{cases} u_i^{best}, & \text{if } rand(0,1) \leq CR \text{ or } i = i_{rand}, \\ x_i^{best}, & \text{otherwise} \end{cases}, \quad (12)$$

其中,  $v_i^{rand}$  为经过所述交叉操作后得到的解基因  $V^{rand}$  的第  $i$  个任务  $T_i$  的候选计算服务,  $u_i^{rand}$  为交叉操作前的解基因  $U_1^{rand}$ 、 $U_2^{rand}$  的第  $i$  个任务  $T_i$  的候选计算服务,  $rand(0,1)$  为从 0 和 1 之间随机产生的一个数,  $CR$  为预设的交叉算子,  $x_i^{rand}$  为变异和交叉前的解基因的候选计算服务,  $v_i^{best}$  为经过所述交叉操作后得到的解基因  $V^{best}$  的第  $i$  个任务  $T_i$  的候选计算服务,  $u_i^{best}$  为解基因  $U_1^{best}$ 、 $U_2^{best}$  的任务  $T_i$  的候选计算服务,  $x_i^{best}$  为变异和交叉前的解基因的候选计算服务;

采用公式 (13) 对经过交叉操作的解基因进行边界修正操作,

$$bmv_i = \begin{cases} n_i - \lfloor \text{mod}(1 - v_j, n_i - 1) \rfloor, & \text{if } v_i \leq 1 \\ v_j, & \text{if } 1 < v_i < n_i, \\ 1 + \lfloor \text{mod}(v_i - n_i, n_i - 1) \rfloor, & \text{if } v_i \geq n_i \end{cases}, \quad (13)$$

其中,  $bmv_i$  为经过边界修正操作后的任务  $T_i$  的候选计算服务的数量,  $v_j$  为进行边界修正操作前的解基因  $V$  的任务  $T_i$ ,  $\lfloor \text{mod}(1 - v_j, n_i - 1) \rfloor$  表示  $1 - v_j$  与  $n_i - 1$  相除的余数下取整,  $\lfloor \text{mod}(v_i - n_i, n_i - 1) \rfloor$  表示  $v_i - n_i$  与  $n_i - 1$  相除的余数下取整;

创建比较解基因集合, 所述比较解基因集合包括所述目标进化解基因集合和经过边界修正操作后的所有解基因;

采用公式 (1) 计算所述比较解基因集合中每个解基因的适应值, 并采用公式 (14) 计算出每个所述解基因的修正适应值,

$$MFit(X) = \begin{cases} Fit(X) & , \text{if } SQ_{Ind,P} \geq \overline{SQ}_{Ind,P} \\ bestfit - |bestfit - Fit(X)| \cdot \frac{vio(X)}{\max vio} & , \text{otherwise} \end{cases}, \quad (14)$$

其中,  $bestfit$  为修正前所述比较解基因集合中解基因的适应值的最大值,  $Vio(X)$  为采用公式 (15) 计算的违约程度,  $\max vio$  为比较解基因集合中解基因的最大违约程度;

$$Vio(X) = \begin{cases} 0 & , \text{if } SQ_{Ind,P} \geq \overline{SQ}_{Ind,P} \\ \sum_{Ind} \left( \frac{\omega_{Ind}}{\sum(\omega_{Ind})} \right) \cdot \frac{(\overline{SQ}_{Ind,P_i} - SQ_{Ind,P_i})}{\overline{SQ}_{Ind,P_i}} & , \text{otherwise} \end{cases}, \quad (15);$$

判断的所述比较解基因集合的最大修正适应值对应的解基因的适应值是否大于所述当前解基因种群的最小适应值;

在判断的所述比较解基因集合的最大修正适应值对应的解基因的适应值大于所述当前解基因种群的最小适应值的情况下, 将所述当前解基因种群中最小适应值对应的解基因替换为所述比较解基因集合的最大修正适应值对应的解基因以更新所述当前解基因种群;

判断所述最大修正适应值对应的解基因的适应值是否大于所述历史最佳解基因的适应值;

在判断所述最大修正适应值对应的解基因的适应值大于所述历史最佳解基因的适应值的情况下,将所述历史最佳解基因替换为所述最大修正适应值对应的解基因以更新所述历史最佳解基因,并将被替换的所述历史最佳解基因加入所述候选最优解集中;

在判断所述最大修正适应值对应的解基因的适应值小于或等于所述历史最佳解基因的适应值的情况下,将所述最大修正适应值对应的解基因加入所述候选最优解集,计算所述最大修正适应值对应的解基因的适应值和更新前的所述历史最佳解基因的适应值的差值的绝对值;

在判断所述绝对值大于所述预设误差的情况下,输出所述历史最佳解基因作为所述最佳部署方案。

6. 根据权利要求5所述的组合优化部署方法,其特征在于,所述根据所述任务包、所述候选计算服务集合、目标函数以及所述边界条件确定完成所述任务包的最佳部署方案包括:

确定所述组合优化部署方法的计算时间;

判断所述计算时间是否大于预设的时间阈值;

在判断所述计算时间大于所述时间阈值的情况下,输出当前的所述历史最佳解基因作为最佳部署方案。

7. 根据权利要求5所述的组合优化部署方法,其特征在于,所述输出所述历史最佳解基因作为所述最佳部署方案包括:

输出所述候选最优解集。

8. 根据权利要求7所述的组合优化部署方法,其特征在于,所述将所述最大修正适应值对应的解基因加入所述候选最优解集包括:

判断所述候选最优解集中的解基因的数量;

判断所述解基因的数量是否大于预设的数量值;

在判断所述解基因的数量大于所述数量值的情况下,删除所述候选最优解集中适应值最小的解基因。

9. 一种云边混合计算服务的组合优化部署系统,其特征在于,所述组合优化部署系统包括处理器,所述处理器用于执行如权利要求1至8任一所述的组合优化部署方法。

10. 一种存储介质,其特征在于,所述存储介质存储有指令,所述指令用于被机器读取以使得所述机器执行如权利要求1至8任一所述的组合优化部署方法。

## 云边混合计算服务的组合优化部署方法、系统及存储介质

### 技术领域

[0001] 本发明涉及云边混合计算服务的优化部署技术领域,具体地涉及一种云边混合计算服务的组合优化部署方法、系统及存储介质。

### 背景技术

[0002] 在5G网络的支持下能够有大量设备接入物联网实现万物互联,其中的边缘计算服务是更加靠近物和数据端的计算服务,能够满足连接海量异构设备,实现数字化的敏捷连接、实时业务和应用智能。而在实施过程中那些需要全局性、非实时、长周期的大数据处理与分析来支持复杂决策的计算服务则需要云计算服务中心的支持。同时在一些企业处于安全隐私、特殊专用计算的任务出可能搭建基于自身特点的候选计算服务。因此基于以上考虑如何在实施过程中混合以上特征类型的计算服务,来进行组合优化部署是实现未来万物互联的生产生活计算服务的重要基础,同时云边协同的计算服务可以更好地完成目标需求、放大数据智能应用价值、降低计算服务成本。

[0003] 在边缘计算、云计算和企业私有计算服务混合的环境中也带来了一定的问题,特别表现以下方面:

[0004] (1) 每个计算服务要满足任务的基本服务质量QoS的需求,而整体计算服务的组合既要满足整体实施过程的基本服务质量QoS的需求,又要能够提供更好的计算服务质量;

[0005] (2) 在整体实施过程中的任务中除了有大量的需要实时性敏捷性强的边缘计算,还有一些具有特殊需求的任务需要计算能力、存储能力和数据分析能力更强的云计算提供服务,所以在组合优化中需要对特殊需求进行有效考虑;

[0006] (3) 多种服务组合也将带来兼容性问题,特别是混合有边缘计算、云计算、私有计算等多种服务,由于其操作系统、服务接口、计算模式的不同会使得兼容性变得十分复杂,在云服务组合中常采用TOSCA(Kim D,Muhammad H,Kim E,et al.TOSCA-Based and Federation-Aware Cloud Orchestration for Kubernetes Container Platform[J].Applied Sciences-Basel,2019,9(1))和容器技术,如Docker和Kubernetes来处理不同服务之间兼容的技术问题。但是由于云边混合计算还处于初级阶段,相应的技术方法还不成熟,所以对于服务兼容性需要设计更具一般性的衡量方法。(4) 对于整体实施过程按照任务间不同的连接方式(S.Deng,H.Wu,et al.Service selection for composition with QoS correlations[J].IEEE Transactions on Services Computing,2016,9(2):291-303),需从整体实施过程的结构特征出发对任务服务质量聚合进行设计;

[0007] (5) 随着市场上边缘服务计算设备的增多,以及不同云计算服务提供商的加入使得计算服务的优化选择需要考虑更多的因素,例如服务类型和服务品牌特征;

[0008] (6) 由于计算服务组合的模型是一个组合优化模型,在算法上是一个NP-hard问题,因此多采用启发式进化算法来迭代求解。在针对云服务组合的研究中已有许多计算方法采用遗传算法(Lin Y-K,Chong CS.Fast GA-based project scheduling for computing resources allocation in a cloud manufacturing system[J].Journal Of

Intelligent Manufacturing,2017,28(5):1189-1201.)、微分进化算法(Qi J,Xu B,Xue Y,et al.Knowledge based differential evolution for cloud computing service composition[J].Journal of Ambient Intelligence and Humanized Computing,2018,9(3):565-574.)等进行求解,但往往只考虑服务质量一个单纯的影响因素。

### 发明内容

[0009] 本发明实施方式的目的是提供一种云边混合计算服务的组合优化部署方法、系统及存储介质,该组合优化部署方法、系统及存储介质可以克服现有技术中的不足,高效、合理地规划处云边混合计算服务的最佳部署方案。

[0010] 为了实现上述目的,本发明实施方式提供一种云边混合计算服务的组合优化部署方法,所述组合优化部署方法包括:

[0011] 获取所述云边混合计算服务的任务包,其中,所述任务包包括需要采用边缘计算服务完成的普通任务集合和需要采用云计算服务的特殊任务集合,所述普通任务集合和所述特殊任务集合均包括至少一个任务,每个所述任务之间通过Petri网结构连接;

[0012] 确定能够完成每个所述任务的候选计算服务集合,其中,每个任务对应有用于完成所述任务的候选计算服务集合,每个所述候选计算服务集合包括至少一个候选计算服务,每个所述候选计算服务包括第一特征属性和第二特征属性,所述第一特征属性包括所述候选计算服务的类型以及用于完成所述候选计算服务的设备的品牌,所述第二特征属性包括完成所述候选计算服务的服务质量;

[0013] 采用公式(1)作为预设的目标函数,

[0014]  $Fit(X) = \omega_1 \cdot TQoS + \omega_2 \cdot TSR + \omega_3 \cdot TSC, (1)$

[0015] 其中,Fit(X)为部署方案X的所述目标函数, $\omega_1$ 、 $\omega_2$ 和 $\omega_3$ 为预设的权重,TQoS为总服务质量QoS表现值,TSR为采用公式(2)计算的特别计算服务满足程度值,

[0016]  $TSR = \left( e^{\ln 2 \times \frac{X_{SRST}}{N_{SRST}}} - 1 \right)^{\gamma_{SRST}}, (2)$

[0017] 其中,TSR为所述特别计算服务满足程度值, $N_{SRST}$ 为所述特别任务集合包括的任务的数量, $X_{SRST}$ 是用于完成所述任务包的部署方案能够完成所述特别任务集合中的任务的数量, $\gamma_{SRST}$ 是预设的积聚强度值,且 $\gamma_{SRST} \geq 1$ ;TSC为采用公式(3)计算的服务一致性程度值,

[0018]  $TSC = \frac{\ln n + \sum_b \frac{X_{B=b}}{n} \ln \frac{X_{B=b}}{n}}{\ln n}, (3)$

[0019] 其中,TSC为所述服务一致性程度值, $X_{B=b}$ 为在用于完成所述任务包的部署方案中,选择了品牌b的候选计算服务的数量,n为所述任务包包括的任务的数量;

[0020] 采用微分进化算法根据所述任务包、所述候选计算服务集合和目标函数确定完成所述任务包的最佳部署方案。

[0021] 可选地,所述组合优化部署方法进一步包括:

[0022] 预设边界条件;

[0023] 所述采用微分进化算法根据所述任务包、所述候选计算服务集合和目标函数确定完成所述任务包的最佳部署方案包括:

[0024] 根据所述任务包、所述候选计算服务集合、目标函数以及所述边界条件确定完成所述任务包的最佳部署方案。

[0025] 可选地,所述边界条件包括公式(4)和公式(5),

$$[0026] \quad SQ_{Ind,T_i} \geq \overline{SQ}_{Ind,T_i}, \quad (4)$$

$$[0027] \quad SQ_{Ind,P} \geq \overline{SQ}_{Ind,P}, \quad (5)$$

[0028] 其中,  $SQ_{Ind,T_i}$  为完成任务  $T_i$  时指标  $Ind$  的标准化指标值,  $\overline{SQ}_{Ind,T_i}$  为完成任务  $T_i$  时指标  $Ind$  的最低要求的标准化指标值,  $SQ_{Ind,P}$  为完成任务包  $P$  时指标  $Ind$  的标准化指标值,  $\overline{SQ}_{Ind,P}$  为完成任务包  $P$  时指标  $Ind$  的最低要求的标准化指标值。

[0029] 可选地,所述组合优化部署方法进一步包括:

[0030] 根据公式(6)计算所述总服务质量  $QoS$  表现值,

$$[0031] \quad TQoS = \sum_{Ind} (\omega_{Ind} / \sum(\omega_{Ind})) SQ_{Ind,P}, \quad (6)$$

[0032] 其中,  $\omega_{Ind}$  为所述候选计算服务的服务质量的指标  $Ind$  的权重,  $SQ_{Ind,P}$  为完成任务包  $P$  时指标  $Ind$  的标准化指标值。

[0033] 可选地,所述根据所述任务包、所述候选计算服务集合、目标函数以及所述边界条件确定完成所述任务包的最佳部署方案包括:

[0034] 遍历所述任务包中的每个所述任务,剔除所述任务对应的候选计算服务集合中不满足所述边界条件的所述候选计算服务;

[0035] 采用整数编码的方式随机生成多个解基因  $X$  以形成初始种群,其中,  $X = \{S_i^* | i = 1, \dots, n\}$ ,  $S_i^*$  为完成任务  $T_i$  所选择的候选计算服务;

[0036] 根据公式(1)计算所述初始种群的每个所述解基因的适应值;

[0037] 设置当前解基因种群、历史最佳解基因以及候选最优解集,其中,初始的所述当前解基因种群为所述初始种群,所述历史最佳解基因为所述当前解基因种群中适应值最大的解基因,所述候选最优解集初始为空集;

[0038] 判断迭代次数是否大于或等于预设的次数阈值;

[0039] 在判断所述迭代次数大于或等于所述次数阈值的情况下,判断适应值的差值的绝对值是否小于或等于预设误差;

[0040] 在判断所述绝对值大于所述预设误差和/或所述迭代次数小于所述次数阈值的情况下,分别从所述当前解基因种群、随机生成的解基因、所述候选最优解集、所述历史最佳解基因随机选择解基因以创建目标进化解基因集合;

[0041] 采用公式(7)至公式(10)对所述目标进化解基因集合进行变异操作,

$$[0042] \quad U_1^{rand} = X^{rand} + F \cdot (X^{r1} - X^{r2}), \quad (7)$$

$$[0043] \quad U_2^{rand} = X^{rand} + F \cdot (X^{r1} - X^{r2}) + F \cdot (X^{r3} - X^{r4}), \quad (8)$$

$$[0044] \quad U_1^{best} = X^{best} + F \cdot (X^{r1} - X^{r2}), \quad (9)$$

$$[0045] \quad U_2^{best} = X^{best} + F \cdot (X^{r1} - X^{r2}) + F \cdot (X^{r3} - X^{r4}), \quad (10)$$



[0046] 其中,  $U_1^{rand}$ 、 $U_2^{rand}$ 、 $U_1^{best}$ 、 $U_2^{best}$ 为经过变异操作后的解基因,  $X^{rand}$ 为从所述当前解基因种群和随机生成的解基因中选择解基因,  $X^{best}$ 为从所述候选最优解集和历史最佳解基因中随机选择的解基因,  $X^{r1}$ 、 $X^{r2}$ 、 $X^{r3}$ 、 $X^{r4}$ 为从当前解基因种群中随机选择的解基因;

[0047] 采用公式 (11) 和 (12) 对变异前的所述目标进化解基因集合和经过变异计算后的所述目标进化解基因集合进行交叉操作,

$$[0048] \quad v_i^{rand} = \begin{cases} u_i^{rand}, & ifrand(0,1) \leq CRori = i_{rand} \\ x_i^{rand}, & otherwise \end{cases}, \quad (11)$$

$$[0049] \quad v_i^{best} = \begin{cases} u_i^{best}, & ifrand(0,1) \leq CRori = i_{rand} \\ x_i^{best}, & otherwise \end{cases}, \quad (12)$$

[0050] 其中,  $v_i^{rand}$ 为经过所述交叉操作后得到的解基因 $V^{rand}$ 的第*i*个任务 $T_i$ 的候选计算服务,  $u_i^{rand}$ 为交叉操作前的解基因 $U_1^{rand}$ 、 $U_2^{rand}$ 的第*i*个任务 $T_i$ 的候选计算服务, rand(0,1)为从0和1之间随机产生的一个数, CR为预设的交叉算子,  $x_i^{rand}$ 为变异和交叉前的解基因的候选计算服务,  $v_i^{best}$ 为经过所述交叉操作后得到的解基因 $V^{best}$ 的第*i*个任务 $T_i$ 的候选计算服务,  $u_i^{best}$ 为解基因 $U_1^{best}$ 、 $U_2^{best}$ 的任务 $T_i$ 的候选计算服务,  $x_i^{best}$ 变异和交叉前的解基因的候选计算服务;

[0051] 采用公式 (13) 对经过交叉操作的解基因进行边界修正操作,

$$[0052] \quad bmv_i = \begin{cases} n_i - \lfloor \text{mod}(1 - v_j, n_i - 1) \rfloor, & if \ v_i \leq 1 \\ v_j, & if \ 1 < v_i < n_i, \\ 1 + \lfloor \text{mod}(v_i - n_i, n_i - 1) \rfloor, & if \ v_i \geq n_i \end{cases}, \quad (13)$$

[0053] 其中,  $bmv_i$ 为经过边界修正操作后的任务 $T_i$ 的候选计算服务,  $n_i$ 为任务 $T_i$ 的候选计算服务的数量,  $v_j$ 为进行边界修正操作前的解基因 $V$ 的任务 $T_i$ ,  $\lfloor \text{mod}(1 - v_j, n_i - 1) \rfloor$ 表示 $1 - v_j$ 与 $n_i - 1$ 相除的余数下取整,  $\lfloor \text{mod}(v_i - n_i, n_i - 1) \rfloor$ 表示 $v_i - n_i$ 与 $n_i - 1$ 相除的余数下取整;

[0054] 创建比较解基因集合, 所述比较解基因集合包括所述目标进化解基因集合和经过边界修正操作后的所有解基因;

[0055] 采用公式 (1) 计算所述比较解基因集合中每个解基因的适应值, 并采用公式 (14) 计算出每个所述解基因的修正适应值,

$$[0056] \quad MFit(X) = \begin{cases} Fit(X) & , \ if \ SQ_{Ind,P} \geq \overline{SQ}_{Ind,P} \\ bestfit - |bestfit - Fit(X)| \cdot \frac{Vio(X)}{\max Vio} & , \ otherwise \end{cases}, \quad (14)$$

[0057] 其中,  $bestfit$ 为修正前所述比较解基因集合中解基因的适应值的最大值,  $Vio(X)$ 为采用公式 (15) 计算的违约程度,  $\max Vio$ 为比较解基因集合中解基因的最大违约程度。

$$[0058] \quad Vio(X) = \begin{cases} 0 & , \text{ if } SQ_{Ind,P} \geq \overline{SQ}_{Ind,P} \\ \sum_{Ind} \left( \frac{\omega_{Ind}}{\text{sum}(\omega_{Ind})} \right) \cdot \frac{(\overline{SQ}_{Ind,P_i} - SQ_{Ind,P_i})}{\overline{SQ}_{Ind,P_i}} & , \text{ otherwise} \end{cases}, \quad (15);$$

[0059] 判断的所述比较解基因集合的最大修正适应值对应的解基因的适应值是否大于所述当前解基因种群的最小适应值；

[0060] 在判断的所述比较解基因集合的最大修正适应值对应的解基因的适应值大于所述当前解基因种群的最小适应值的情况下，将所述当前解基因种群中最小适应值对应的解基因替换为所述比较解基因集合的最大修正适应值对应的解基因以更新所述当前解基因种群；

[0061] 判断所述最大修正适应值对应的解基因的适应值是否大于所述历史最佳解基因的适应值；

[0062] 在判断所述最大修正适应值对应的解基因的适应值大于所述历史最佳解基因的适应值的情况下，将所述历史最佳解基因替换为所述最大修正适应值对应的解基因以更新所述历史最佳解基因，并将被替换的所述历史最佳解基因加入所述候选最优解集中；

[0063] 在判断所述最大修正适应值对应的解基因的适应值小于或等于所述历史最佳解基因的适应值的情况下，将所述最大修正适应值对应的解基因加入所述候选最优解集，计算所述最大修正适应值对应的解基因的适应值和更新前的所述历史最佳解基因的适应值的差值的绝对值；

[0064] 在判断所述绝对值大于所述预设误差的情况下，输出所述历史最佳解基因作为所述最佳部署方案。

[0065] 可选地，所述根据所述任务包、所述候选计算服务集合、目标函数以及所述边界条件确定完成所述任务包的最佳部署方案包括：

[0066] 确定所述组合优化部署方法的计算时间；

[0067] 判断所述计算时间是否大于预设的时间阈值；

[0068] 在判断所述计算时间大于所述时间阈值的情况下，输出当前的所述历史最佳解基因作为最佳部署方案。

[0069] 可选地，所述输出所述历史最佳解基因作为所述最佳部署方案包括：

[0070] 输出所述候选最优解集。

[0071] 可选地，所述将所述最大修正适应值对应的解基因加入所述候选最优解集包括：

[0072] 判断所述候选最优解集中的解基因的数量；

[0073] 判断所述解基因的数量是否大于预设的数量值；

[0074] 在判断所述解基因的数量大于所述数量值的情况下，删除所述最优解集中适应值最小的解基因。

[0075] 另一方面，本发明还提供一种云边混合计算服务的组合优化部署系统，所述组合优化部署系统包括处理器，所述处理器用于执行如上述任一所述的组合优化部署方法。

[0076] 再一方面，本发明还提供一种存储介质，所述存储介质存储有指令，所述指令用于被机器读取以使得所述机器执行如上述任一所述的组合优化部署方法。

[0077] 通过上述技术方案，本发明提供的云边混合计算服务的组合优化部署方法、系统及存储介质通过结合包括服务质量以外的多种因素，采用改进的微分进化算法确定云边混

合计算服务的最佳部署方案,解决了现有技术中在确定方案时由于仅结合单一的服务质量的因素而导致部署方案无法达到最优的技术问题,使得最终形成的部署方案能够最大程度地满足云边混合计算服务的要求。

[0078] 本发明实施方式的其它特征和优点将在随后的具体实施方式部分予以详细说明。

### 附图说明

[0079] 附图是用来提供对本发明实施方式的进一步理解,并且构成说明书的一部分,与下面的具体实施方式一起用于解释本发明实施方式,但并不构成对本发明实施方式的限制。在附图中:

[0080] 图1是根据本发明的一个实施方式的云边混合计算服务的组合优化部署方法的流程图;

[0081] 图2是根据本发明的一个实施方式的云边混合计算服务的组合优化部署方法的流程图;

[0082] 图3是根据本发明的一个实施方式的微分进化算法的流程图;以及

[0083] 图4是根据本发明的一个实施方式的微分进化算法的流程图。

### 具体实施方式

[0084] 以下结合附图对本发明实施方式的具体实施方式进行详细说明。应当理解的是,此处所描述的具体实施方式仅用于说明和解释本发明实施方式,并不用于限制本发明实施方式。

[0085] 在本发明实施方式中,在未作相反说明的情况下,使用的方位词如“上、下、顶、底”通常是针对附图所示的方向而言的或者是针对竖直、垂直或重力方向上而言的各部件相互位置关系描述用词。

[0086] 另外,若本发明实施方式中有涉及“第一”、“第二”等的描述,则该“第一”、“第二”等的描述仅用于描述目的,而不能理解为指示或暗示其相对重要性或者隐含指明所指示的技术特征的数量。由此,限定有“第一”、“第二”的特征可以明示或者隐含地包括至少一个该特征。另外,各个实施方式之间的技术方案可以相互结合,但是必须是以本领域普通技术人员能够实现为基础,当技术方案的结合出现相互矛盾或无法实现时应当认为这种技术方案的结合不存在,也不在本发明要求的保护范围之内。

[0087] 如图1所示是根据本发明的一个实施方式的云边混合计算服务的组合优化部署方法的流程图。在图1中,该组合优化部署方法可以包括:

[0088] 在步骤S10中,获取云边混合计算服务的任务包。其中,该任务包可以包括需要采用边缘计算服务完成的普通任务集合和需要采用云计算服务的特殊任务集合。普通任务集合和特殊任务集合均可以包括至少一个任务,每个任务之间可以通过Petri网结构连接。

[0089] 在步骤S11中,确定能够完成每个任务的候选计算服务集合。其中,每个任务对应有用于完成该任务的候选计算服务集合。每个候选计算服务集合包括至少一个候选计算服务。每个候选计算服务包括第一特征属性和第二特征属性。该第一特征属性可以包括候选计算服务的类型以及用于完成该候选计算服务的设备的品牌,第二特征属性可以包括完成候选计算服务的服务质量。其中,对于该类型,可以包括例如一般性边缘计算服务GE、云计

算服务SC以及企业私有计算服务PC。该服务质量可以是例如用于评价候选计算服务的各个指标,例如包括用于表示该候选计算服务 $S_{ij}$ 在服务执行时间内可以被唤醒的概率的可获得性 $ava_{S_{ij}}$ 、用于表示该候选计算服务 $S_{ij}$ 在服务执行时间内可以成功运行且不会发生错误中断的概率的可靠性 $rel_{S_{ij}}$ 、用于表示该候选计算服务 $S_{ij}$ 在执行时所使用的时间的运行时间 $tim_{S_{ij}}$ 、用于表示该候选计算服务 $S_{ij}$ 的价格的产品成本 $cos_{S_{ij}}$ 、用于表示该候选计算服务 $S_{ij}$ 的总体质量和品牌因素的综合高低程度的产品声望等级 $rep_{S_{ij}}$  (用来区别对产品本身的信任程度)。因此,针对每个候选计算服务,其服务质量均可以表示为指标集合 $QoS_{set} = \{ava, rel, tim, cos, rep\}$ 。在该实施方式中,对于任意一个指标,可以采用 $Q_{Ind}$ 来表示。相应地,对于任意一个任务 $T_i$ ,其对应的任意一个候选计算服务 $S_{ij}$ 的服务质量则可以采用 $Q_{Ind, S_{ij}}$ 来表示,且 $Ind \in QoS_{set}$ 。另外,由于服务质量的各个指标的优劣与数值大小的对应关系可能不同,以运行时间 $tim_{S_{ij}}$ 为例,其对应的数值越大,服务质量则越低;而以产品声望等级 $rep_{S_{ij}}$ 为例,其对应的数值越大,服务质量则越高。因此,为了便于统一计算服务质量,在该实施方式中,可以将每个指标 $Ind$ 分为正向计算服务质量 $Ind^+$ 和负向计算服务质量 $Ind^-$ 。其中, $Ind^+ = \{ava, rel, rep\}$ ,  $Ind^- = \{tim, cos\}$ 。

[0090] 在步骤S12中,采用公式(1)作为预设的目标函数,

$$[0091] \quad Fit(X) = \omega_1 \cdot TQoS + \omega_2 \cdot TSR + \omega_3 \cdot TSC, \quad (1)$$

[0092] 其中, $Fit(X)$ 为部署方案 $X$ 的目标函数, $\omega_1$ 、 $\omega_2$ 和 $\omega_3$ 为预设的权重, $TQoS$ 为总服务质量 $QoS$ 表现值, $TSR$ 为采用公式(2)计算的特定计算服务满足程度值,

$$[0093] \quad TSR = \left( e^{\ln 2 \times \frac{X_{SRST}}{N_{SRST}}} - 1 \right)^{\gamma_{SRST}}, \quad (2)$$

[0094] 其中, $TSR$ 为特定计算服务满足程度值, $N_{SRST}$ 为特殊任务集合包括的任务的数量, $X_{SRST}$ 是用于完成任务包的部署方案能够完成特殊任务集合中的任务的数量, $\gamma_{SRST}$ 是预设的积聚强度值,且 $\gamma_{SRST} \geq 1$ ;  $TSC$ 为采用公式(3)计算的服务一致性程度值,

$$[0095] \quad TSC = \frac{\ln n + \sum_b \frac{X_{B=b}}{n} \ln \frac{X_{B=b}}{n}}{\ln n}, \quad (3)$$

[0096] 其中, $TSC$ 为服务一致性程度值, $X_{B=b}$ 为在用于完成任务包的部署方案中,选择了品牌 $b$ 的候选计算服务的数量, $n$ 为任务包包括的任务的数量。另外,对于该总服务质量 $QoS$ 表现值,其计算方式可以是本领域人员所知的多种形式。在本发明的一个示例中,可以根据公式(4)计算总服务质量 $QoS$ 表现值,

$$[0097] \quad TQoS = \sum_{Ind} (\omega_{Ind} / \sum (\omega_{Ind})) SQ_{Ind, P}, \quad (4)$$

[0098] 其中, $\omega_{Ind}$ 为候选计算服务的服务质量的指标 $Ind$ 的权重, $SQ_{Ind, P}$ 为完成任务包 $P$ 时指标 $Ind$ 的标准化指标值。在该示例中,考虑到任务包中的各个任务之间可能存在串行模式、并行模式、选择模式以及循环模式的四种关系,其中,串行模式表示其中的任务需要按照先后次序完成,并行模式表示其中的任务需要同时完成,选择模式表示其中的任务只能同时选择一者或多者完成,循环模式表示其中的任务需要循环完成。因此,在计算该目标函

数前,可以先将该任务包中的任务分成:

[0099] 串行模式任务组集合  $SEQ = \{seq_1, \dots, seq_{k_{seq}}, \dots, seq_{n_{seq}}\}$ 、

[0100] 并行模式任务组集合  $PAR = \{par_1, \dots, par_{k_{par}}, \dots, par_{n_{par}}\}$ 、

[0101] 选择模式任务组集合  $SEL = \{sel_1, \dots, sel_{k_{sel}}, \dots, sel_{n_{sel}}\}$  以及

[0102] 循环模式任务组集合  $CIR = \{cir_1, \dots, cir_{k_{cir}}, \dots, cir_{n_{cir}}\}$ 。

[0103] 其中,  $seq_{k_{seq}}$  为串行模式任务组集合 SEQ 的第  $k_{seq}$  个串行任务组,  $n_{seq}$  为串行模式任务组集合 SEQ 的串行任务组的个数;  $par_{k_{par}}$  为并行模式任务组集合 PAR 的第  $k_{par}$  个并行任务组,  $n_{par}$  为并行任务集合 PAR 的并行任务组的个数;  $sel_{k_{sel}}$  为选择模式任务组集合 SEL 的第  $k_{sel}$  个选择任务组,  $n_{sel}$  为选择模式任务组集合 SEL 的选择任务组的个数;  $cir_{k_{cir}}$  为循环模式任务组集合 CIR 的第  $k_{cir}$  个循环任务组,  $n_{cir}$  为循环模式任务组集合中的循环任务组的个数。依据本领域人员所知 (H.Liang, Y.Du. Dynamic service selection with QoS constraints and inter-service correlations using cooperative coevolution [J] Future Generation Computer Systems. 2017, 76:119-135), 并行模式任务组集合、选择模式任务组集合以及循环模式任务组集合均可以通过预设的规则最终转化为串行模式任务组集合中的一部分。所以, 在计算该总服务质量 QoS 表现值时, 可以首先对各个任务集合进行聚合计算。具体地, 可以根据公式 (5) 至公式 (9) 对串行模式任务组集合的任务进行聚合计算,

$$[0104] \quad SQ_{ava, seq_k} = \prod_{i \in seq_k} SQ_{ava, T_i}, \quad (5)$$

$$[0105] \quad SQ_{rel, seq_k} = \prod_{i \in seq_k} SQ_{rel, T_i}, \quad (6)$$

$$[0106] \quad SQ_{tim, seq_k} = \sum_{i \in seq_k} SQ_{tim, T_i}, \quad (7)$$

$$[0107] \quad SQ_{cos, seq_k} = \sum_{i \in seq_k} SQ_{cos, T_i}, \quad (8)$$

$$[0108] \quad SQ_{rep, seq_k} = \sum_{i \in seq_k} SQ_{rep, T_i} / n_{seq_k} \quad (9)$$

[0109] 其中,  $SQ_{ava, seq_k}$  为完成串行模式任务组集合 SEQ 的 (串行) 任务组  $seq_k$  的可获得性的标准化指标值, 任务组  $seq_k$  包括处于同一个串行模式的多个任务,  $SQ_{ava, T_i}$  为完成任务  $T_i$  的可获得性的标准化指标值,  $SQ_{rel, seq_k}$  为完成串行模式任务组集合 SEQ 的任务组  $seq_k$  的可靠性的标准化指标值,  $SQ_{rel, T_i}$  为完成串行模式任务组集合 SEQ 的任务组  $seq_k$  的运行时间的标准化指标值,  $SQ_{tim, T_i}$  为完成任务  $T_i$  的运行时间的标准化指标值,  $SQ_{cos, seq_k}$  为完成串行模式任务组集合 SEQ 的任务组  $seq_k$  的产品成本的标准化指标值,  $SQ_{cos, T_i}$  为完成任务  $T_i$  的产品成本的标准化指标值,  $SQ_{rep, seq_k}$  为完成串行模式任务组集合 SEQ 的任务组  $seq_k$  的

的产品声望等级的标准化指标值,  $SQ_{rep,T_i}$  为完成任务  $T_i$  的产品声望等级的标准化指标值,  $n_{seq_k}$  是任务组  $seq_k$  中任务的个数,  $seq_k \in SEQ$ ;

[0110] 根据公式 (10) 至公式 (14) 对并行模式任务组集合的任务进行聚合计算,

$$[0111] \quad SQ_{ava,par_k} = \prod_{i \in par_k} SQ_{ava,T_i}, \quad (10)$$

$$[0112] \quad SQ_{rel,par_k} = \prod_{i \in par_k} SQ_{rel,T_i}, \quad (11)$$

$$[0113] \quad SQ_{tim,par_k} = \max_{i \in par_k} (SQ_{tim,T_i}), \quad (12)$$

$$[0114] \quad SQ_{cos,par_k} = \sum_{i \in par_k} SQ_{cos,T_i}, \quad (13)$$

$$[0115] \quad SQ_{rep,par_k} = \sum_{i \in seq_k} SQ_{rep,T_i} / n_{par_k}, \quad (14)$$

[0116] 其中,  $SQ_{ava,par_k}$  为完成并行模式任务组集合 PAR 的 (并行) 任务组  $par_k$  的可获得性的标准化指标值, 任务组  $par_k$  包括处于同一个并行模式的多个任务,  $SQ_{ava,T_i}$  为完成任务  $T_i$  的可获得性的标准化指标值,  $SQ_{rel,par_k}$  为完成任务并行模式任务组集合 PAR 的任务组  $par_k$  的可靠性的标准化指标值,  $SQ_{rel,T_i}$  为完成任务  $T_i$  的可靠性的标准化指标值,  $SQ_{tim,par_k}$  为完成并行模式任务组集合 PAR 的任务组  $par_k$  的运行时间的标准化指标值,  $SQ_{tim,T_i}$  为完成任务  $T_i$  的运行时间的标准化指标值,  $SQ_{cos,par_k}$  为完成并行模式任务组集合 PAR 的任务组  $par_k$  的产品成本的标准化指标值,  $SQ_{cos,T_i}$  为完成任务  $T_i$  的产品成本的标准化指标值,  $SQ_{rep,par_k}$  为完成并行模式任务组集合 PAR 的任务组  $par_k$  的声望等级的标准化指标值,  $SQ_{rep,T_i}$  为完成任务  $T_i$  的声望等级的标准化指标值,  $n_{par_k}$  是任务组  $par_k$  的任务个数,  $par_k \in PAR$ ;

[0117] 根据公式 (15) 至公式 (19) 对选择模式任务组集合的任务进行聚合计算,

$$[0118] \quad SQ_{ava,sel_k} = \sum_{i \in sel_k} (SQ_{ava,T_i} \times \theta_i), \quad (15)$$

$$[0119] \quad SQ_{rel,sel_k} = \sum_{i \in sel_k} (SQ_{rel,T_i} \times \theta_i), \quad (16)$$

$$[0120] \quad SQ_{tim,sel_k} = \sum_{i \in sel_k} (SQ_{tim,T_i} \times \theta_i), \quad (17)$$

$$[0121] \quad SQ_{cos,sel_k} = \sum_{i \in sel_k} SQ_{cos,T_i}, \quad (18)$$

$$[0122] \quad SQ_{rep,sel_k} = \sum_{i \in sel_k} SQ_{rep,T_i} / n_{sel_k}, \quad (19)$$

[0123] 其中,  $SQ_{ava,sel_k}$  为完成选择模式任务组集合 SEL 的 (选择) 任务组  $sel_k$  的可获得性的标准化指标值,  $SQ_{ava,T_i}$  为完成任务  $T_i$  的可获得性的标准化指标值,  $\theta_i$  为任务  $T_i$  在任务组  $sel_k$  中被选择执行的概率,  $SQ_{rel,sel_k}$  为完成选择模式任务组集合 SEL 的任务组  $sel_k$  的可靠性的标准化指标值,  $SQ_{rel,T_i}$  为完成任务  $T_i$  的可靠性的标准化指标值,  $SQ_{tim,sel_k}$  为完成选

择模式任务组集合SEL的任务组 $sel_k$ 的运行时间的标准化指标值,  $SQ_{tim,T_i}$ 为完成任务 $T_i$ 的运行时间的标准化指标值,  $SQ_{cos,sel_k}$ 为完成选择模式任务组集合SEL的任务组 $sel_k$ 的产品成本的标准化指标值,  $SQ_{cos,T_i}$ 为完成任务 $T_i$ 的产品成本的标准化指标值,  $SQ_{rep,sel_k}$ 为完成选择模式任务组集合SEL的任务组 $sel_k$ 的声望等级的标准化指标值,  $SQ_{rep,T_i}$ 为完成任务 $T_i$ 的声望等级的标准化指标值,  $n_{sel_k}$ 为任务组 $sel_k$ 的任务个数,  $sel_k \in SEL$ ;

[0124] 根据公式(20)至公式(24)对循环模式任务组集合的任务循环进行聚合计算,

$$[0125] \quad SQ_{ava,cir_k} = \prod_{i \in cir_k} SQ_{ava,T_i}, \quad (20)$$

$$[0126] \quad SQ_{rel,cir_k} = \prod_{i \in cir_k} SQ_{rel,T_i}, \quad (21)$$

$$[0127] \quad SQ_{tim,cir_k} = \gamma \times \sum_{i \in cir_k} SQ_{tim,T_i}, \quad (22)$$

$$[0128] \quad SQ_{cos,cir_k} = \gamma \times \sum_{i \in cir_k} SQ_{cos,T_i}, \quad (23)$$

$$[0129] \quad SQ_{rep,cir_k} = \sum_{i \in cir_k} SQ_{rep,T_i} / n_{cir_k}, \quad (24)$$

[0130] 其中,  $SQ_{ava,cir_k}$ 为完成循环模式任务组集合CIR的(循环)任务组 $cir_k$ 的可获得性的标准化指标值,任务组 $cir_k$ 包括处于同一个循环模式的多个任务,  $SQ_{ava,T_i}$ 为完成任务 $T_i$ 的可获得性的标准化指标值,  $SQ_{rel,cir_k}$ 为完成循环模式任务组集合CIR的任务组 $cir_k$ 的可靠性的标准化指标值,  $SQ_{rel,T_i}$ 为完成任务 $T_i$ 的可靠性的标准化指标值,  $SQ_{tim,cir_k}$ 为完成循环模式任务组集合CIR的任务组 $cir_k$ 的运行时间的标准化指标值,  $\gamma$ 为任务组 $cir_k$ 的循环次数,  $SQ_{tim,T_i}$ 为完成任务 $T_i$ 的运行时间的标准化指标值,  $SQ_{cos,cir_k}$ 为完成循环模式任务组集合CIR的任务组 $cir_k$ 的产品成本的标准化指标值,  $SQ_{cos,T_i}$ 为完成任务 $T_i$ 的产品成本的标准化指标值,  $SQ_{rep,cir_k}$ 为完成循环模式任务组集合CIR的任务组 $cir_k$ 的声望等级的标准化指标值,  $SQ_{rep,T_i}$ 为完成任务 $T_i$ 的声望等级的标准化指标值,  $n_{cir_k}$ 为任务组 $cir_k$ 的任务个数,  $cir_k \in CIR$ 。

[0131] 再对聚合计算的结果形成的整体ALLMODE进行聚合计算。具体地,可以根据公式(25)至公式(29)进行聚合计算,

$$[0132] \quad SQ_{ava,P} = \prod SQ_{ava,ALLMODE}, \quad (25)$$

$$[0133] \quad SQ_{rel,P} = \prod SQ_{rel,ALLMODE}, \quad (26)$$

$$[0134] \quad SQ_{tim,P} = \sum SQ_{tim,ALLMODE}, \quad (27)$$

$$[0135] \quad SQ_{cos,P} = \sum SQ_{cos,ALLMODE}, \quad (28)$$

$$[0136] \quad SQ_{rep,P} = \sum SQ_{rep,ALLMODE} / n_{ALLMODE}, \quad (29)$$

[0137] 其中,  $SQ_{ava,P}$ 完成任务包P的可获得性的标准化指标值,  $SQ_{ava,ALLMODE}$ 为完成整体ALLMODE中的一个元素(任务)的可获得性的标准化指标值,  $SQ_{rel,P}$ 为完成任务包P的可靠性的标准化指标值,  $SQ_{rel,ALLMODE}$ 为完成整体ALLMODE中的一个元素的可靠性的标准化指标值,  $SQ_{tim,P}$ 为完成任务包P的运行时间的标准化指标值,  $SQ_{tim,E}$ 为完成整体ALLMODE中的一个元

素的运行时间的标准化指标值,  $SQ_{\cos,P}$  完成任务包P的产品成本的标准化指标值,  $SQ_{\cos,ALLMODE}$  为完成整体ALLMODE中的一个元素的产品成本的标准化指标值,  $SQ_{rep,P}$  完成任务包P的声望等级的标准化指标值,  $SQ_{rep,ALLMODE}$  为完成整体ALLMODE中的一个元素的声望等级的标准化指标值,  $n_{ALLMODE}$  为整体ALLMODE中的元素的个数。

[0138] 考虑到每个指标Ind分为正向计算服务质量 $Ind^+$ 和负向计算服务质量 $Ind^-$ 。其中,  $Ind^+ = \{ava, rel, rep\}$ ,  $Ind^- = \{tim, cos\}$ 。那么, 在计算出各个标准化指标值后, 可以进一步采用公式(30)和公式(31)对计算出的标准化指标值进行进一步地标准化处理,

$$[0139] \quad SQ_{Ind^+,T_i} = \begin{cases} \frac{Q_{Ind^+,T_i} - \min_j(Q_{Ind^+,S_{ij}})}{\max_j(Q_{Ind^+,S_{ij}}) - \min_j(Q_{Ind^+,S_{ij}})}, & \text{if } \max_j(Q_{Ind^+,S_{ij}}) \neq \min_j(Q_{Ind^+,S_{ij}}), \\ 1, & \text{if } \max_j(Q_{Ind^+,S_{ij}}) = \min_j(Q_{Ind^+,S_{ij}}) \end{cases}, \quad (30)$$

$$[0140] \quad SQ_{Ind^-,T_i} = \begin{cases} \frac{\max_j(Q_{Ind^-,S_{ij}}) - Q_{Ind^-,T_i}}{\max_j(Q_{Ind^-,S_{ij}}) - \min_j(Q_{Ind^-,S_{ij}})}, & \text{if } \max_j(Q_{Ind^-,S_{ij}}) \neq \min_j(Q_{Ind^-,S_{ij}}), \\ 1, & \text{if } \max_j(Q_{Ind^-,S_{ij}}) = \min_j(Q_{Ind^-,S_{ij}}) \end{cases}, \quad (31)$$

[0141] 其中,  $SQ_{Ind^+,T_i}$  为标准化处理后的任务 $T_i$ 的正向计算服务质量的标准化指标值,  $Q_{Ind^+,T_i}$  为标准化处理前的任务 $T_i$ 的正向计算服务质量的标准化指标值,  $\max_j(Q_{Ind^+,S_{ij}})$  和  $\max_j(Q_{Ind^-,S_{ij}})$  为任务 $T_i$ 的候选计算服务中最大的服务质量 $Q_{Ina}$  (正向计算服务质量或负向计算服务质量) 的值,  $\min_j(Q_{Ind^+,S_{ij}})$  和  $\min_j(Q_{Ind^-,S_{ij}})$  为任务 $T_i$ 的候选计算服务中最大的服务质量 $Q_{Ind}$  的值, 以可获得性 $ava_{S_{ij}}$ 为例, 其服务质量 $Q_{ava}$  为正向计算服务质量, 即:

$$[0142] \quad \max_j(Q_{ava,S_{ij}}) = \max \{Q_{ava,S_{ij}} \mid \forall S_{ij} \in CS_i\},$$

$$[0143] \quad \min_j(Q_{ava,S_{ij}}) = \min \{Q_{ava,S_{ij}} \mid \forall S_{ij} \in CS_i\},$$

[0144] 通过这样的方式使得最终计算出的总服务质量 $QoS$ 表现值均位于 $[0, 1]$ 之间, 从而使得五个 $QoS$ 表现值不出现数值量纲和单位的不统一, 也便于微分进化算法的计算。

[0145] 在步骤S13中, 采用微分进化算法根据任务包、候选计算服务集合和目标函数确定完成任务包的最佳部署方案。

[0146] 在本发明的一个实施方式中, 为了提高微分进化算法所确定的最佳部署方案的效果, 该组合优化部署方法可以包括如图2中所示出的步骤。与图1中示出的组合优化部署方法的不同之处在于, 在图2中, 该组合优化部署方法还包括:

[0147] 在步骤S13中, 预设边界条件。对于该边界条件, 可以是本领域人员结合实际的用户需求来制定。在本发明的一个示例中, 该边界条件可以包括公式(32)和公式(33),

$$[0148] \quad SQ_{Ind,T_i} \geq \overline{SQ}_{Ind,T_i}, \quad (32)$$

$$[0149] \quad SQ_{Ind,P} \geq \overline{SQ}_{Ind,P}, \quad (33)$$



[0150] 其中,  $SQ_{Ind,T_i}$  为完成任务  $T_i$  时指标  $Ind$  的标准化指标值,  $\overline{SQ}_{Ind,T_i}$  为完成任务  $T_i$  时指标  $Ind$  的最低要求的标准化指标值,  $SQ_{Ind,P}$  为完成任务包  $P$  时指标  $Ind$  的标准化指标值,  $\overline{SQ}_{Ind,P}$  为完成任务包  $P$  时指标  $Ind$  的最低要求的标准化指标值。

[0151] 相应的, 该步骤 S14 则可以为根据该任务包、候选计算服务集合、目标函数以及边界条件确定完成任务包的最佳部署方案。在该步骤 S14 中, 采用微分进化算法根据该任务包、候选计算服务集合、目标函数以及边界条件来确定最佳部署方案的具体细节可以是本领域人员所公知的步骤。在本发明的一个示例中, 该步骤 S14 可以具体包括如图 3 中所示出的步骤。在图 3 中, 该步骤 S14 可以具体包括:

[0152] 在步骤 S1401 中, 向该微分进化算法中输入各个参数。

[0153] 在步骤 S1402 中, 遍历任务包中的每个任务, 剔除任务对应的候选计算服务集合中不满足边界条件的候选计算服务。在该实施方式中, 考虑到该微分进化算法需要对每个候选计算服务进行计算。因此, 候选计算服务的数量直接影响到整个微分进化算法的计算速度。在该步骤 S1402 中, 在执行该微分进化算法前, 根据预设的边界条件 (公式 (31) 和公式 (32)) 剔除每个任务对应的候选计算服务集合中的不满足条件的候选计算服务可以减少候选计算服务的数量, 从而提高整体算法的运行速度, 也降低了执行计算的设备的负载。

[0154] 在步骤 S1403 中, 采用整数编码的方式随机生成多个解基因  $X$  以形成初始种群。其中,  $X = \{S_i^* | i = 1, \dots, n\}$ ,  $S_i^*$  为完成任务  $T_i$  所选择的候选计算服务;

[0155] 在步骤 S1404 中, 根据公式 (1) 计算初始种群的每个解基因的适应值;

[0156] 在步骤 S1405 中, 设置当前解基因种群、历史最佳解基因以及候选最优解集。其中, 初始的当前解基因种群为初始种群, 历史最佳解基因为当前解基因种群中适应值最大的解基因, 候选最优解集初始为空集;

[0157] 在步骤 S1406 中, 判断迭代次数是否大于或等于预设的次数阈值。其中, 初始的迭代次数可以为 0;

[0158] 在步骤 S1407 中, 在判断迭代次数大于或等于次数阈值的情况下, 判断适应值的差值的绝对值是否小于或等于预设误差;

[0159] 在步骤 S1408 中, 在判断绝对值大于预设误差和/或迭代次数小于次数阈值的情况下, 分别从当前解基因种群、随机生成的解基因、候选最优解集、历史最佳解基因随机选择解基因以创建目标进化解基因集合。相较于传统微分进化算法只随机选择当前解基因种群中的基因和历史最佳基因, 本发明提供的该微分进化算法通过设置目标进化解基因集合为后续筛选提供了更加丰富的解基因, 并且, 从候选最优解集中随机选择的基因相对于其他集合中的基因具有更好的适应值, 大大减少了算法的运行时间。另外, 通过随机选择的方式来创建该目标进化解基因集合也使得该目标进化解基因集合更具一般性, 这也提高了算法计算的准确性。

[0160] 在步骤 S1409 中, 采用公式 (34) 至公式 (37) 对目标进化解基因集合进行变异操作,

$$[0161] \quad U_1^{rand} = X^{rand} + F \cdot (X^{r1} - X^{r2}), \quad (34)$$

$$[0162] \quad U_2^{rand} = X^{rand} + F \cdot (X^{r1} - X^{r2}) + F \cdot (X^{r3} - X^{r4}), \quad (35)$$

$$[0163] \quad U_1^{best} = X^{best} + F \cdot (X^{r1} - X^{r2}), \quad (36)$$

$$[0164] \quad U_2^{best} = X^{best} + F \cdot (X^{r1} - X^{r2}) + F \cdot (X^{r3} - X^{r4}), \quad (37)$$

[0165] 其中,  $U_1^{rand}$ 、 $U_2^{rand}$ 、 $U_1^{best}$ 、 $U_2^{best}$ 为经过变异操作后的解基因,  $X^{rand}$ 为从当前解基因种群和随机生成的解基因中选择的解基因,  $X^{best}$ 为从候选最优解集和历史最佳解基因中随机选择的解基因,  $X^{r1}$ 、 $X^{r2}$ 、 $X^{r3}$ 、 $X^{r4}$ 为从当前解基因种群中随机选择的解基因;

[0166] 在步骤S1410中,采用公式(38)和(39)对变异前的目标进化解基因集合和经过变异计算后的目标进化解基因集合进行交叉操作,

$$[0167] \quad v_i^{rand} = \begin{cases} u_i^{rand}, & \text{ifrand}(0,1) \leq CRori = i_{rand} \\ x_i^{rand}, & \text{otherwise} \end{cases}, \quad (38)$$

$$[0168] \quad v_i^{best} = \begin{cases} u_i^{best}, & \text{ifrand}(0,1) \leq CRori = i_{rand} \\ x_i^{best}, & \text{otherwise} \end{cases}, \quad (39)$$

[0169] 其中,  $v_i^{rand}$ 为经过交叉操作后得到的解基因 $V^{rand}$ 的第*i*个任务 $T_i$ 的候选计算服务,  $u_i^{rand}$ 为交叉操作前的解基因 $U_1^{rand}$ 、 $U_2^{rand}$ 的第*i*个任务 $T_i$ 的候选计算服务, rand(0,1)为从0和1之间随机产生的一个数, CR为预设的交叉算子,  $x_i^{rand}$ 为变异和交叉前的解基因的候选计算服务,  $v_i^{best}$ 为经过交叉操作后得到的解基因 $V^{best}$ 的第*i*个任务 $T_i$ 的候选计算服务,  $u_i^{best}$ 为解基因 $U_1^{best}$ 、 $U_2^{best}$ 的任务 $T_i$ 的候选计算服务,  $x_i^{best}$ 变异和交叉前的解基因的候选计算服务。相对于传统微分进化算法提供的单一的随机数的交叉方式,在本发明提供的该微分进化算法还进一步采用了随机点法(公式(38)和公式(39))的交叉方式,从而使得经过交叉操作的解基因种类更加丰富;

[0170] 在步骤S1411中,采用公式(40)对经过交叉操作的解基因进行边界修正操作,

$$[0171] \quad bmv_i = \begin{cases} n_i - \lfloor \text{mod}(1 - v_j, n_i - 1) \rfloor, & \text{if } v_i \leq 1 \\ v_j, & \text{if } 1 < v_i < n_i, \\ 1 + \lfloor \text{mod}(v_i - n_i, n_i - 1) \rfloor, & \text{if } v_i \geq n_i \end{cases}, \quad (40)$$

[0172] 其中,  $bmv_i$ 为经过边界修正操作后的任务 $T_i$ 的候选计算服务,  $n_i$ 为任务 $T_i$ 的候选计算服务的数量,  $v_j$ 为进行边界修正操作前的解基因 $V$ 的任务 $T_i$ ,  $\lfloor \text{mod}(1 - v_j, n_i - 1) \rfloor$ 表示 $1 - v_j$ 与 $n_i - 1$ 相除的余数下取整,  $\lfloor \text{mod}(v_i - n_i, n_i - 1) \rfloor$ 表示 $v_i - n_i$ 与 $n_i - 1$ 相除的余数下取整。在该实施方式中,由于考虑到在进行变异操作后,解基因的某些位置上的整数编码可能超出候选计算服务序号的范围。因此在该实施方式中,采用循环编码(公式(40))的方式对变异操作后的解基因进行边界修正。

[0173] 在步骤S1412中,创建比较解基因集合,比较解基因集合包括目标进化解基因集合和经过边界修正操作后的所有解基因;

[0174] 在步骤S1413中,采用公式(1)计算比较解基因集合中每个解基因的适应值,并采

用公式(41)计算出每个解基因的修正适应值,

$$[0175] \quad MFit(X) = \begin{cases} Fit(X) & , \text{ if } SQ_{Ind,P} \geq \overline{SQ}_{Ind,P} \\ bestfit - |bestfit - Fit(X)| \cdot \left| \frac{Vio(X)}{maxVio} \right| & , \text{ otherwise} \end{cases}, \quad (41)$$

[0176] 其中,bestfit为修正前比较解基因集合中解基因的适应值的最大值,Vio(X)为采用公式(42)计算的违约程度,maxVio为比较解基因集合中解基因的最大违约程度。

$$[0177] \quad Vio(X) = \begin{cases} 0 & , \text{ if } SQ_{Ind,P} \geq \overline{SQ}_{Ind,P} \\ \sum_{Ind} \left( \frac{\omega_{Ind}}{sum(\omega_{Ind})} \right) \cdot \frac{(\overline{SQ}_{Ind,P_i} - SQ_{Ind,P_i})}{\overline{SQ}_{Ind,P_i}} & , \text{ otherwise} \end{cases}, \quad (42);$$

[0178] 在步骤S1414中,判断的比较解基因集合的最大修正适应值对应的解基因的适应值是否大于当前解基因种群的最小适应值;

[0179] 在步骤S1415中,在判断的比较解基因集合的最大修正适应值对应的解基因的适应值大于当前解基因种群的最小适应值的情况下,将当前解基因种群中最小适应值对应的解基因替换为比较解基因集合的最大修正适应值对应的解基因以更新当前解基因种群并执行步骤S1416;

[0180] 在步骤S1416中,判断最大修正适应值对应的解基因的适应值是否大于历史最佳解基因的适应值;

[0181] 在步骤S1417中,在判断最大修正适应值对应的解基因的适应值大于历史最佳解基因的适应值的情况下,将历史最佳解基因替换为最大修正适应值对应的解基因以更新历史最佳解基因,并将被替换的历史最佳解基因加入候选最优解集中并执行步骤S1418;

[0182] 在步骤S1418中,在判断最大修正适应值对应的解基因的适应值小于或等于历史最佳解基因的适应值的情况下,将最大修正适应值对应的解基因加入候选最优解集,计算最大修正适应值对应的解基因的适应值和更新前的历史最佳解基因的适应值的差值的绝对值并执行步骤S1406;

[0183] 在步骤S1419中,在判断绝对值大于预设误差的情况下,输出历史最佳解基因作为最佳部署方案。

[0184] 进一步地,为了避免算法的计算时间过长,该微分进化算法也可以包括如图4所示出的步骤。与图3中示出的微分进化算法的不同之处在于,在图4中,该微分进化算法进一步包括:

[0185] 在步骤S1406中,确定组合优化部署方法的计算时间,判断该计算时间是否大于预设的时间阈值;

[0186] 在步骤S1420中,在判断计算时间大于时间阈值的情况下,输出当前的历史最佳解基因作为最佳部署方案。

[0187] 另外,为了便于工作人员选择最佳部署方案,在步骤S1420中,在输出历史最佳解基因作为最佳部署方案的同时,也可以输出候选最优解集以供工作人员选择。更进一步地,在执行如图1至图3中任一的组合优化部署方法时,考虑到产生的该候选最优解集的数量可能较多,为了避免内存占用过大,在将最大修正适应值对应的解基因加入候选最优解集时,也可以执行以下步骤:

[0188] 步骤1、判断候选最优解集中的解基因的数量;

[0189] 步骤2、判断解基因的数量是否大于预设的数量值；

[0190] 步骤3、在判断解基因的数量大于该数量值的情况下，删除最优解集中适应值最小的解基因。

[0191] 另一方面，本发明还提供一种云边混合计算服务的组合优化部署系统，该组合优化部署系统可以包括处理器，该处理器可以用于执行如上述任一的组合优化部署方法。

[0192] 再一方面，本发明还提供一种存储介质，该存储介质可以存储有指令，该指令可以用于被机器读取以使得该机器执行如上述任一的组合优化部署方法。

[0193] 通过上述技术方案，本发明提供的云边混合计算服务的组合优化部署方法、系统及存储介质通过结合包括服务质量以外的多种因素，采用改进的微分进化算法确定云边混合计算服务的最佳部署方案，解决了现有技术中在确定方案时由于仅结合单一的服务质量的因素而导致部署方案无法达到最优的技术问题，使得最终形成的部署方案能够最大程度地满足云边混合计算服务的要求。

[0194] 以上结合附图详细描述了本发明例的可选实施方式，但是，本发明实施方式并不限于上述实施方式中的具体细节，在本发明实施方式的技术构思范围内，可以对本发明实施方式的技术方案进行多种简单变型，这些简单变型均属于本发明实施方式的保护范围。

[0195] 另外需要说明的是，在上述具体实施方式中所描述的各个具体技术特征，在不矛盾的情况下，可以通过任何合适的方式进行组合。为了避免不必要的重复，本发明实施方式对各种可能的组合方式不再另行说明。

[0196] 本领域技术人员可以理解实现上述实施方式方法中的全部或部分步骤是可以通程序来指令相关的硬件来完成，该程序存储在一个存储介质中，包括若干指令用以使得一个（可以是单片机，芯片等）或处理器（processor）执行本申请各个实施方式所述方法的全部或部分步骤。而前述的存储介质包括：U盘、移动硬盘、只读存储器（ROM, Read-Only Memory）、随机存取存储器（RAM, Random Access Memory）、磁碟或者光盘等各种可以存储程序代码的介质。

[0197] 此外，本发明实施方式的各种不同的实施方式之间也可以进行任意组合，只要其不违背本发明实施方式的思想，其同样应当视为本发明实施方式所公开的内容。

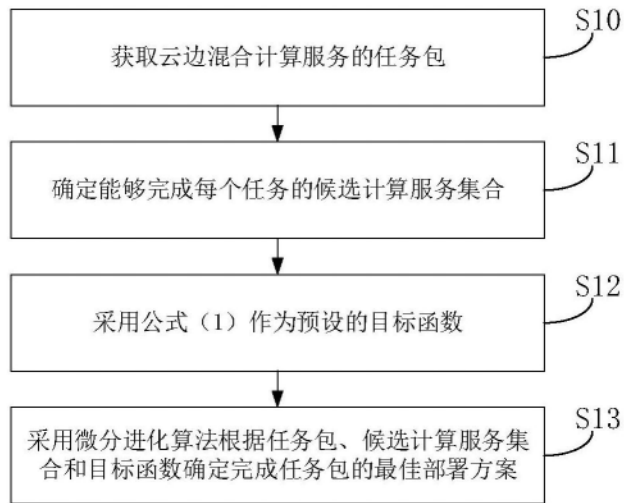


图1

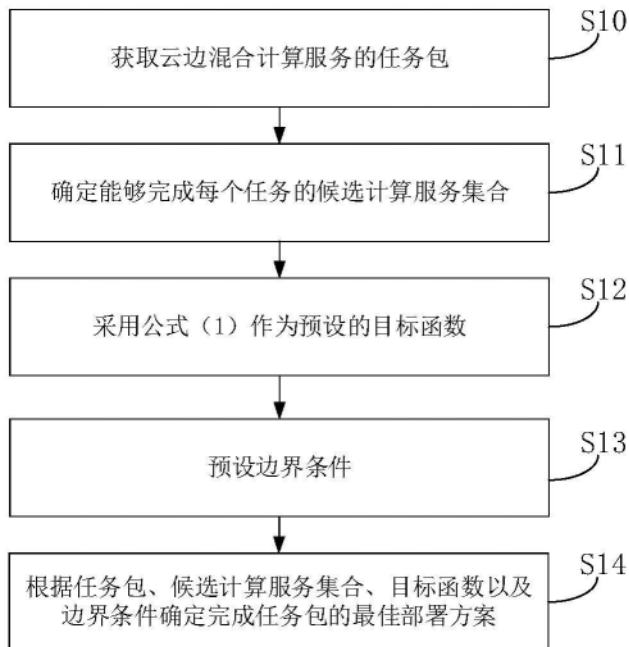


图2

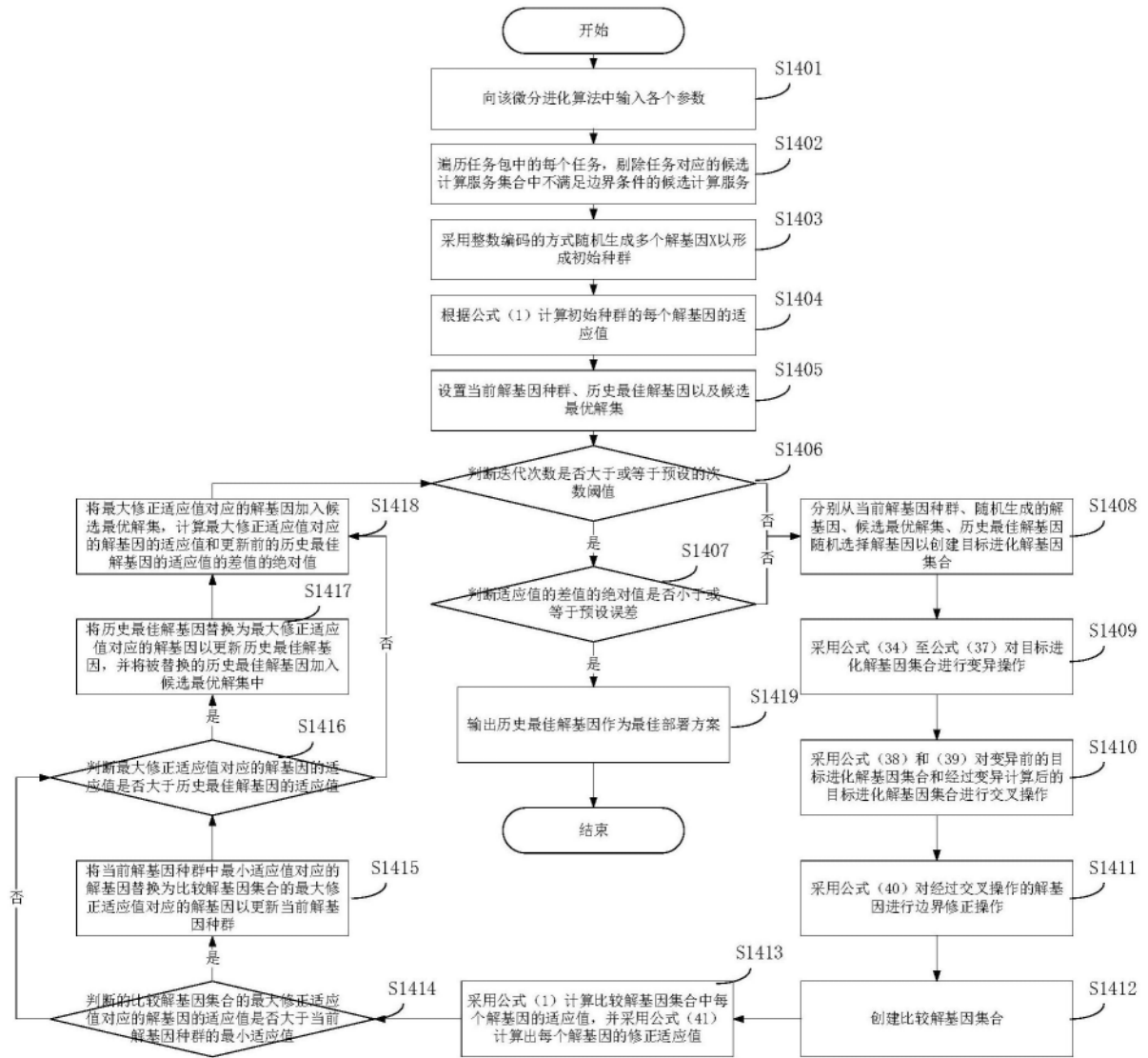


图3

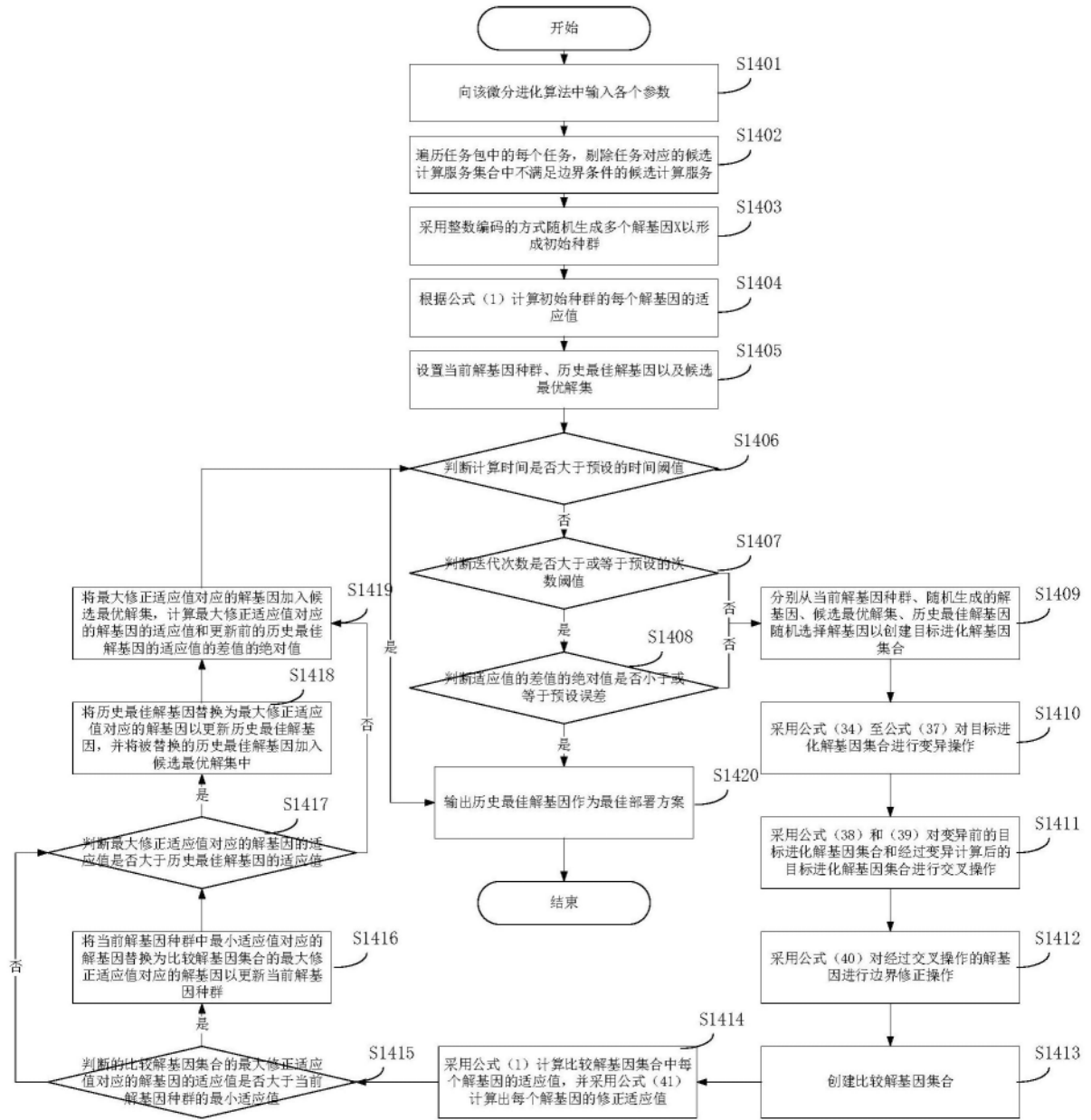


图4