面向广域分布式计算环境的任务与资源动态双向匹配方法

尚晶[1]，肖利民[2,3]，肖智文[1]，王锦权[2,3]，武智晖[1]，李辉阳[2,3]，张逸飞[1]，宋尧[4]，王冀彬[1]
1. 中移动信息技术有限公司，北京 100033；
2. 复杂关键软件环境全国重点实验室，北京 100191；
3. 北京航空航天大学计算机学院，北京 100191；
4. 中国信息通信研究院，北京 100191

摘要
广域分布式计算环境可提供大规模的计算和存储资源，是支持算力互联和数据流转的重要基础设施。在广域分布式计算环境中，任务与资源的匹配对于提高系统性能具有重要意义。然而，任务与资源的多样性、地理位置分散的资源会增加二者匹配的复杂性。针对响应延迟高、匹配效率低等问题，提出了面向广域分布式计算环境的任务与资源动态匹配方法，通过建立统一的任务需求模型和资源能力模型来简化匹配过程，降低响应延迟。此外，定义了任务向匹配度和资源向匹配度以刻画任务视角和资源视角的偏好，并权衡二者；定义了任务和资源的双向综合匹配度以量化任务需求和资源能力的适配程度。最后通过动态计算每一组任务与资源间的双向综合匹配度以优化匹配效果。实验结果表明，与现有的方法相比，该方法可提升匹配效果，并大幅降低平均响应延迟。

关键词
广域协同调度；资源匹配；双向匹配；广域分布式计算环境

中图分类号：TP316　　文献标志码：A　　doi: 10.11959/j.issn.2096-0271.2024050

A dynamic bidirectional matching method of tasks and resources oriented to wide-area distributed computing

SHANG Jing[1], XIAO Limin[2,3], XIAO Zhiwen[1], WANG Jinquan[2,3], WU Zhihui[1], LI Huiyang[2,3], ZHANG Yifei[1], SONG Yao[4], WANG Jibin[1]
1. China Mobile Information Technology Center, Beijing 100033, China
2. State Key Laboratory of Complex & Critical Software Environment, Beijing 100191, China
3. Computer Science and Engineering, Beihang University, Beijing 100191, China
4. China Academy of Information and Communications Technology, Beijing 100191, China

Abstract
Due to the huge capacities of computing and storage resources, wide-area distributed computing environment has become important infrastructures supporting computing power and data interconnection. In wide-area distributed computing

environment, matching of tasks and resources is important to improve system performance. However, the diversity of tasks and resources and the geographical dispersion of resources increase the complexity of matching problems. To solve the problems of high response delay and low matching efficiency, a dynamic bidirectional matching method of tasks and resources oriented to wide-area distributed computing environments is proposed. The matching process is simplified and the response delay is mitigated by building a unified task requirement model and resource capability model. Moreover, the task-oriented and resource-oriented matching degrees are defined to express the preference of task-perspective and resource-perspective; the two-side comprehensive matching degree of tasks and resources is defined by the trade-off of the task-oriented and resource-oriented matching degree. The two-side comprehensive matching degrees are dynamically calculated for each task group and the resources to improve the matching quality. The experimental results show that the proposed method can effectively promoting the matching quality and significantly reduce the response delay compared with the existing methods.

Key words
wide-area collaborative scheduling, resource matching, bidirectional matching, wide-area distributed computing environment

0 引言

由于科学和工程计算问题日益复杂，应用需要协同广域环境中分散的算力资源以支持高效计算[1]。广域分布式计算环境可发挥广域分散资源的聚合效应，因而受到广泛关注。构建广域分布式计算环境以支持高效的算力互联和数据流转，已成为各国算力竞争的焦点[2-5]。各大研究机构和互联网公司纷纷通过网络联通、地理位置分散的超算中心、数据中心和边缘计算节点等算力中心，构建广域分布式计算环境[2-4,6]。近年来，我国也陆续开展“东数西算”工程[7]、中国算力网[8]、国家超算互联网[9]等重点工作，加速广域分布式计算环境建设。

在广域分布式计算环境中，任务与资源间的匹配通过分析任务需求和全局资源能力以针对任务选定一个或多个合适的调度目标，二者的匹配对系统性能具有重要作用[10]。匹配过程的复杂程度与任务需求、资源能力密切相关，即随着任务和资源多样性的增加，匹配问题的建模过程更加复杂且其求解搜索空间显著增长[11]。此外，由于任务和资源的多样性、地理位置分散的资源，在收集系统信息时广域网通信的开销大，这将影响任务和资源的匹配效率[12]。广域分布式计算环境中的匹配问题是一个NP难的组合优化问题[13]，其求解可能产生较高的响应延迟或较差的匹配效果，即任务需经过长时间等待才能被处理或总体任务完成时间较长。因此，在广域分布式计算环境中，实现任务需求和资源能力之间的高效匹配非常具有挑战性。

现有的匹配方法分为静态匹配[14-16]和动态匹配[17-20]两类。静态匹配方法适用于考虑静态任务和资源并预定义计算和通信行为的静态场景，具有全局视角，能够更好地进行匹配决策，但静态匹配会通过广域网通信多次获取任务需求和资源能力等系统信息，这会带来较高的响应延迟，影响服务质量（quality of service，QoS）。动态匹配方法使用启发式方法避免匹配过程响应延迟高的问题，并根据特定的需求定义任务和资源之间的匹配度。但现有的动态匹配方法主要关注负载均衡、保障QoS等方面，其匹配度定义缺乏对提升系统计算效率的关注。在面向多样的任务需求和

资源能力时，由于计算能力有限，动态匹配方法往往无法支撑对匹配度定义方式的迭代优化。因此，在动态匹配方法中定义高效、合理的匹配度十分具有挑战性。综上所述，在面向多样的任务需求和资源能力时，现有的匹配方法无法同时满足对响应延迟和匹配效果方面的要求。

针对上述挑战，本文提出一种面向广域分布式计算环境的任务与资源动态双向匹配方法。通过建立广域分布式计算环境中的任务与资源特征模型，对多样化的任务需求和资源能力进行统一的模型化描述；提出了一种动态双向匹配策略，综合考虑任务视角和资源视角的偏好，定义任务需求和资源能力间的双向综合匹配度，从而量化分析任务需求和资源能力的适配程度；基于双向综合匹配度生成任务调度决策，并动态反馈资源信息以支持后续匹配。本方法利用任务-资源双向综合匹配度以保障任务和资源间的适配性，通过简化资源匹配过程提升响应速度，从而实现任务需求和资源能力的高效匹配。

本文的主要贡献如下。

- 提出了一种面向广域分布式计算环境的任务与资源动态双向匹配方法，以实现任务需求和资源能力的高效匹配。
- 设计并实现了一个基于任务与资源动态双向匹配方法的原型调度系统，可优化广域环境中的全局资源利用，支持计算任务的广域协同处理。
- 在广域位置分散的5个节点上部署了原型调度系统，并执行标准的MPI并行计算任务，实验结果表明，本文方法可有效提升匹配效果，并大幅降低平均响应延迟。

1 国内外研究现状

任务需求和资源能力间的适配性是影响计算性能的关键因素之一，任务与资源的匹配方法需要对任务需求和资源能力进行建模，针对每个任务选定一个或多个合适的调度目标。现有的匹配方法可以分为静态匹配[10,14-16]和动态匹配[17-20]方法。

静态匹配适用于考虑预定义任务集并已知其计算和通信行为的静态场景。静态匹配具有系统的全局视图，能够更好地进行匹配决策。在早期的研究工作[21-22]中，匹配问题通常被设计为组合优化问题，以任务需求和资源能力模型为约束条件，以任务调度目标选择和资源分配为决策变量，设置特定的优化目标，并使用混合整数线性规划（mixed integer linear programming，MILP）求解器、遗传算法等进行求解[15,23-24]。文献[14]将匹配问题建模为具有均衡约束的均衡问题，并提出了基于多对多匹配的雾节点资源分配模型。由于优化是在设计阶段进行的，静态匹配方法可以使用更全面的系统信息进行决策，获得更好的匹配效果。但是，静态匹配方法难以在运行时处理任务和资源的动态变化，如新任务进入系统或资源状态动态变化。针对动态变化，静态匹配方法通常定期收集资源信息，并定期对一个批次的任务进行匹配[21]。为了获取更好的匹配效果，静态匹配方法需要通过广域网收集全面的系统信息以支持匹配。但这种广域网通信非常缓慢，特别是在需要收集多轮信息以支持匹配策略的迭代优化时，广域网通信的开销使得静态匹配的响应延迟大幅提升。简而言之，在广域分布式计算环境中，面向多样、动态的任务和资源时，现有的静态匹配方法的问题建模和求解过程缓慢，响应延迟高。

随着任务需求和资源能力的多样化，匹配问题建模和求解的时间开销增加，难以满足用户对计算任务快速响应的需求。针对这样的动态场景，研究者发现需

要采用动态匹配方法，在较短的响应延迟内完成匹配决策，以满足用户的QoS需求[13,17-18.20,25-27]。动态匹配方法面向特定需求定义任务和资源之间的匹配度，如任务需求向量与资源能力向量间的相似程度[18,25]、任务对资源的公平占有度[20]、用户和资源的双向偏好程度[13, 17]等。因匹配度定义各不相同，动态资源匹配方法的优化效果不同。文献[18]提出了一种面向边缘计算中任务分配的资源匹配方法，将任务的存储、计算、内存需求和资源的存储、计算、内存能力分别建模为三元组模型，并将任务模型和资源模型间的余弦相似度作为匹配度。文献[26]提出了一种基于改进式蚁群算法的移动边缘计算资源调度方法，在用户任务和分散的服务器之间进行匹配和调度。上述动态匹配方法使用了典型的贪婪启发式算法，通过简单的匹配过程来减少响应延迟，但同时也会降低匹配效果。

为在保证响应延迟的同时优化任务和资源间的适配性，从任务和资源的双重视角进行资源匹配的双向匹配方法成为近年来的研究热点[13,17,19,28]。文献[17]分别从任务和资源视角设定偏好信息、构造双向的偏好矩阵，并计算双向的贴近度，最后将任务视角对资源的贴近度和资源视角对任务的贴近度整合为双向决策值，以判断任务和资源是否适配。文献[19]提出的双向匹配方法分别基于任务收益和资源收益定义了任务和资源的期望模型，然后根据期望模型定义了双边匹配度以进行双向匹配。文献[28]提出的双向综合匹配方法综合考虑了任务向匹配度和资源向匹配度，可有效提升匹配效果，但其匹配度的定义未考虑网络因素，而低带宽、高延迟的广域网络会造成调度信息延迟高、数据迁移迟缓等问题。因此，上述双向匹配度的定义仍然是低效的，这些定义缺乏对全局资源的统筹考虑，可能导致负载失衡。简而言之，由于匹配度定义未综合考虑多样任务和资源的状态，现有的动态匹配方法难以保障匹配效果。

综上所述，现有的资源匹配方法难以同时满足广域分布式计算环境中任务和资源之间匹配过程响应延迟和匹配效果的需求。静态匹配方法综合考虑了系统信息，可以获得更好的匹配效果，但存在较高的响应延迟。动态匹配方法可以简化匹配过程以减少响应延迟，但其局部视图难以保障匹配效果。此外，大多数动态匹配方法在定义匹配度时，并不直接关注如何提升系统效率，而侧重于负载均衡、保障QoS等，这导致动态匹配方法的匹配效果有限。因此，如何高效地进行多样化任务需求和资源能力之间的匹配是一个亟须解决的问题。

2 面向广域分布式计算环境的任务与资源动态双向匹配方法

本文提出了面向广域分布式计算环境的任务与资源动态双向匹配方法（dynamic bidirectional matching method，DBMM），该方法通过分析多样化的任务需求和资源能力特征，生成任务需求和资源能力的模型化描述，以简化任务与资源间的匹配过程；在此基础上，定义了任务向匹配度和资源向匹配度以刻画任务视角和资源视角的偏好，并基于二者的双向一致性和双向互补性定义了任务和资源的双向综合匹配度，从而量化任务需求和资源能力间的适配程度。DBMM充分考虑了全局任务需求和资源能力间的双向综合匹配度，同时避免了复杂的匹配过程，实现了广域环境中多样化任务需求和资源能力的高效匹配，如**图1**所示。

2.1 任务与资源特征建模

面向广域分布式计算环境中多样化的任务需求和资源能力，DBMM基于任务需求和资源能力的形式化描述，分别建立任务需求模型和资源能力模型，以消除多样化的任务需求和资源能力带来的复杂性。为方便描述，本文对任务和资源进行如下设定：①用任务矩阵 $\mathbb{T}=[t_1,t_2,t_3,\cdots,t_M]^{\mathrm{T}}$ 描述计算平台中现存的所有任务，设定共存在 M 个计算任务，且这些任务在后续的调度和执行过程中不可被再次切分；②用资源矩阵 $\mathbb{P}=[p_1,p_2,p_3,\cdots,p_N]^{\mathrm{T}}$ 描述计算平台中的所有算力中心，设定共存在 N 个算力中心，这些算力中心的地理位置分散但网络互联，且这些算力中心包含计算、存储、应用、网络等多种算力资源，因此每个算力中心可视为一个包含了多种算力资源的“资源聚合体”。

基于上述设定，DBMM针对任务侧的资源需求和中心侧的资源供给能力进行特征分析，以提取可用于支持匹配过程的特征。

任务需求由用户提出，是对所提交的计算任务需要占据的资源量、期望的资源状态等内容的描述，其中，资源量指的是算力资源的数量。因此，每个任务 t 可由计算核数需求 C_t 、存储容量需求 S_t 、传输带宽需求 B_t 、预估工作时长 $T_{t\mathrm{pred}}$ 、应用需求 A_t 组成的五元组来描述，如式（1）所示：

$$t=\left[C_t \quad S_t \quad B_t \quad T_{t\mathrm{pred}} \quad A_t\right] \tag{1}$$

为方便与任务需求的匹配，算力中心被描述为一个包含多种算力资源的“资源聚合体”，资源能力被用于描述算力中心包含的资源状态和应用部署情况等。具体而言，资源能力在计算资源方面包含算力中心的总核数 C_p 、允许的任务最大占有核数 $C_{p,\mathrm{limit}}$ 、正在运行的核数 $C_{p,\mathrm{run}}$ 和在该中心内排队的任务所需求的总核数 $C_{p,\mathrm{wait}}$ ，在存储资源方面包含算力中心的总存储容量 S_p 和已使用的存储容量 $S_{p,\mathrm{used}}$ ，在网络资源方面包含算力中心允许的最大数据传输带宽 B_p 和已被占用的网络带宽 $B_{p\mathrm{used}}$ ，在计算时长方面包含算力中心允许的最大计算时长 $T_{p,\mathrm{limit}}$ 。此外，算力中心的应用部署情况由集合 $A_p=\{A_{\mathrm{app1}},A_{\mathrm{app2}},A_{\mathrm{app3}},\cdots\}$ 表示。上述参数中，计算、存储、网络等算力资源的占用情况均由计算平台内各算力中心动态收集。因此，每个算力中心 p 都可采用一个十元组进行描述，如式（2）所示：

$$p=\left[C_p \quad C_{p,\mathrm{limit}} \quad C_{p,\mathrm{run}} \quad C_{p,\mathrm{wait}} \quad S_p \quad S_{p,\mathrm{used}} \quad B_p \quad B_{p,\mathrm{sed}} \quad T_{p,\mathrm{limit}} \quad A_p\right] \tag{2}$$

任务需求模型和资源能力模型可以对多样化的任务需求和计算平台的资源能力进行统一描述，是任务–资源双向匹配的基础，可简化资源匹配过程。

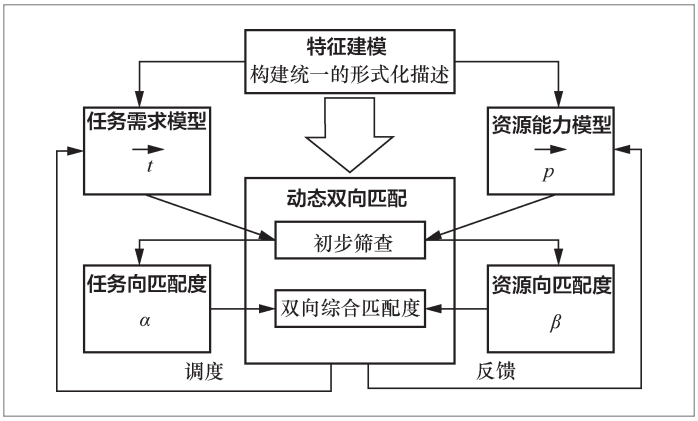


图1 DBMM 流程

2.2 动态双向匹配策略

2.2.1 初步筛查

在进行任务需求和资源能力的动态双向匹配时，首先需对任务需求模型和资

源能力模型进行初步筛查，以去除任务需求与资源能力不相符的任务-资源配对，从而简化后续的匹配过程。具体地，对于式(1)所示任务t和式(2)所示算力中心p，初步筛查限制条件如式(3)所示：

$$\begin{cases} C_t \leqslant C_{p,\text{limit}} \\ S_t \leqslant S_p - S_{p,\text{used}} \\ B_t \leqslant B_p - B_{p,\text{used}} \\ T_{t,\text{pred}} \leqslant T_{p,\text{limit}} \\ A_t \in \mathbb{A}_p \end{cases} \tag{3}$$

2.2.2 匹配度定义

在初步筛查后，剩下的每个任务需求和资源能力的配对都代表了一种可行的匹配方案，但可能存在一个任务需求匹配到多个资源的情况，因此需要从多种可行的匹配方案中选出更合适的方案。本文提出的DBMM考虑了任务需求和资源能力间的相似程度从而定义了任务向匹配度，考虑了资源负载情况与全局负载均衡从而定义了资源向匹配度，基于任务向匹配度和资源向匹配度计算任务-资源双向综合匹配度，并以双向综合匹配度为匹配方案的评价指标。

任务和资源的匹配可以看作一个时序变化的装箱问题[29]。任务向匹配度主要考虑任务在对应资源执行对全局资源利用的影响，为保证全局资源的利用率，需选择与任务相似程度较高的资源进行匹配。因此，DBMM构建了一个三维向量$\vec{t}$（包含任务的计算核数需求C_t、数据存储需求S_t、传输带宽需求B_t）以表示任务需求，如式(4)所示：

$$\vec{t} = (C_t, S_t, B_t) \tag{4}$$

同时，将算力中心的总核数C_p、总存储容量S_p、最大数据传输带宽B_p构建为一个三维向量$\vec{p}$以表示资源能力，如式(5)所示：

$$\vec{p} = (C_p, S_p, B_p) \tag{5}$$

任务向匹配度$\alpha_{t,p}$表示任务需求与资源能力的相似程度，其取值范围为(0,1)，如式(6)所示：

$$\alpha_{t,p} = S_c(\vec{t}, \vec{p}) = \frac{\vec{t} \cdot \vec{p}}{|\vec{t}| \times |\vec{p}|} \tag{6}$$

任务向匹配度主要用于表示资源在满足任务需求的同时是否会造成算力资源使用不均，需判断任务需求向量与资源能力向量的相似程度，因此用余弦相似度定义任务向匹配度[18]。任务向匹配度越大，任务需求和资源能力的相似程度越高。此时向该资源代表的算力中心分配任务，算力中心的计算和存储资源仍被均衡使用，对其他任务分配的影响较小，并且在相同资源量的情况下对应平台可放入的任务数量更多。

资源向匹配度则从资源能力偏好出发，判断资源和任务的适配性。在广域分布式计算环境中，负载均衡是提升全局资源利用并发挥广域分散资源聚合效应的有效措施[30]。资源向匹配度主要考虑各算力中心的已用资源负载情况和全局负载均衡，用于判断任务被分配至该资源后对负载均衡的影响。

当任务t被分配至算力中心p执行时，该算力中心的计算、存储、网络资源占用率如式(7)所示：

$$\begin{cases} L_{t,p,\text{omp}} = \dfrac{C_{p,\text{run}} + C_{p,\text{wait}} + C_t}{C_p} \\ L_{t,p,\text{stor}} = \dfrac{S_{p,\text{used}} + S_t}{S_p} \\ L_{t,p,\text{et}} = \dfrac{B_{p,\text{used}} + B_t}{B_p} \end{cases} \tag{7}$$

对于多个算力中心，采用min-max归一化方法将每个算力中心的计算、存储、

网络资源占用率映射到取值范围(0,1]中，如式(8)所示：

$$L' = \frac{L - L_{\min}}{L_{\max} - L_{\min}} \tag{8}$$

其中，L 可为算力中心 p 的计算 $L_{t,p,\text{comp}}$、存储 $L_{t,p,\text{stor}}$、网络资源占用率 $L_{t,p,\text{net}}$，而 $L_{\max}$ 和 $L_{\min}$ 代表多个算力中心对某一类算力资源占用率的最大值和最小值。考虑到全局负载均衡和各类算力资源的负载情况，资源向匹配度 $\beta_{t,p}$ 的取值范围被设定为[0,1)，如式(9)所示：

$$\beta_{t,p} = \left| \left(\frac{1 - L'_{t,p,\text{comp}}}{\sqrt{3}}, \frac{1 - L'_{t,p,\text{stor}}}{\sqrt{3}}, \frac{1 - L'_{t,p,\text{net}}}{\sqrt{3}} \right) \right| \tag{9}$$

资源向匹配度的定义在min-max归一化过程中考虑了全局负载均衡的因素，并且归一化后采用向量的模求得资源的空闲程度，可直观体现各类资源的负载情况。

DBMM基于任务向匹配度和资源向匹配度定义了双向综合匹配度，综合考虑并权衡了任务和资源的偏好。双向综合匹配度 $M_{t,p}$ 的取值范围被设定为(0,1)，如式(10)所示：

$$M_{t,p} = \lambda \times \left(\frac{\alpha_{t,p} + \beta_{t,p}}{2} \right) + (1 - \lambda) \times \sqrt{\alpha_{t,p} \times \beta_{t,p}} \tag{10}$$

双向综合匹配度 $M_{t,p}$ 为任务向匹配度 $\alpha_{t,p}$ 和资源向匹配度 $\beta_{t,p}$ 的算数平均值和几何平均值加权求和，以代表二者在双向一致性和双向互补性之间的权衡。

2.2.3 调度方案设计

基于双向综合匹配度 $M_{t,p}$，DBMM可选择出合理适配的资源-任务配对，以生成高效的调度方案。由于匹配策略的设定，任务矩阵 $\mathbb{T}$ 中的每个任务与资源矩阵 $\mathbb{P}$ 中每个经过初筛的资源均可计算得到一个双向综合匹配度，匹配度高的资源更加适配任务，即任务被分配至该资源可获得更高的计算性能且系统资源利用更优。

双向综合匹配度 $M_{t,p}$ 权衡了任务向匹配度 $\alpha_{t,p}$ 和资源向匹配度 $\beta_{t,p}$，即在“任务视角下最适配的资源”和“资源视角下最适配的任务”之间进行权衡，并通过参数 λ 进行调整。

当 $\lambda = 0$ 时，式(10)表达了任务向匹配度和资源向匹配度的双向一致性，仅当任务向匹配度 $\alpha_{t,p}$ 和资源向匹配度 $\beta_{t,p}$ 取值都接近1、表达匹配程度高时，双向综合匹配度 $M_{t,p}$ 的取值才会接近1，否则双向综合匹配度 $M_{t,p}$ 的结果可能表达任务与资源不适配。例如，当任务向匹配度 $\alpha_{t,p} = 1$（即任务视角下资源完全适配），而资源向匹配度 $\beta_{t,p} = 0$（即资源视角下完全不适配）时，偏重双向一致的双向综合匹配度 $M_{t,p}$ 取值为0，表示任务与资源不匹配。此时，双向综合匹配度并不能体现任务向匹配度所表达的匹配偏好。任务向匹配度和资源向匹配度的双向一致性可保障任务和资源在偏好上的相似程度，从而避免调度方案不公平。

当参数 $\lambda = 1$ 时，式(10)表达了任务向匹配度和资源向匹配度的双向互补性，此时无论任务向匹配度 $\alpha_{t,p}$ 和资源向匹配度 $\beta_{t,p}$ 所表达的匹配偏好是何种状态，双向综合匹配度 $M_{t,p}$ 都能在一定程度上体现二者的匹配偏好。例如，在任务需求和资源能力不相似但资源空闲程度很大的情况下，任务向匹配度 $\alpha_{t,p}$ 的取值近乎为0，而资源向匹配度 $\beta_{t,p}$ 的取值接近1，双向互补模式下的双向综合匹配度仍会认为任务需求和资源能力间具有一定的适配性，从而将任务调度至空闲资源上执行。此机制可避免任务被长期搁置的情况。

双向一致性和双向互补性对定义双向

综合匹配度具有重要意义，有助于在各种资源和任务情况下综合判断任务和资源间的匹配度，避免出现任务分配不公平、任务长时间等待的情况。在实验中，参数 λ 的取值为0.5，旨在均衡地考虑任务向匹配度 $\alpha_{t,p}$ 和资源向匹配度 $\beta_{t,p}$ 之间的双向一致性和双向互补性。

在计算出任务需求和资源能力之间的双向综合匹配度后，需为每个任务选择出合适的资源，双向综合匹配度较大的任务–资源配对代表当前更为合理的调度方案，即任务被分配至该资源可获得更高的计算性能且系统资源利用更优。因此，在生成资源匹配方案时，以系统中任务的提交时间为优先级，按序进行资源匹配。对于每个任务，DBMM将其与所有资源的双向综合匹配度进行降序排列，从而设定该任务的可选调度目标集合，将双向综合匹配度作为调度方案生成的依据。当任务与多个资源的双向综合匹配度相同时，则将资源向匹配度值作为排序依据。对于资源向匹配度大的资源，由于其空闲程度较高，选择该资源作为调度目标可有效均衡全局资源负载。若资源向匹配度也相同，则可任选其一作为调度目标。在一个任务的资源匹配方案被确定后，DBMM将其向计算平台反馈，并对计算平台中的资源占用情况进行动态更新。

2.3 原型系统设计

DBMM的原型系统设计如图2所示。DBMM所需的模块被部署在中心调度器和子调度器两个组件上。在资源匹配过程中，中心调度器用于统筹管理和协调全局资源并生成匹配方案，子调度器分布在各算力中心之内，用于管理和监控算力中心内的任务和资源。在资源匹配过程中各模块的具体功能如下。

- 任务队列模块：用户向计算平台提交任务后，该模块会收集任务信息并对任务进行统一描述，然后将任务补充到任务需求模型中。
- 资源信息汇总模块：该模块根据调度决策模块及各算力中心的反馈，收集计算平台中的资源信息并对资源进行统一描

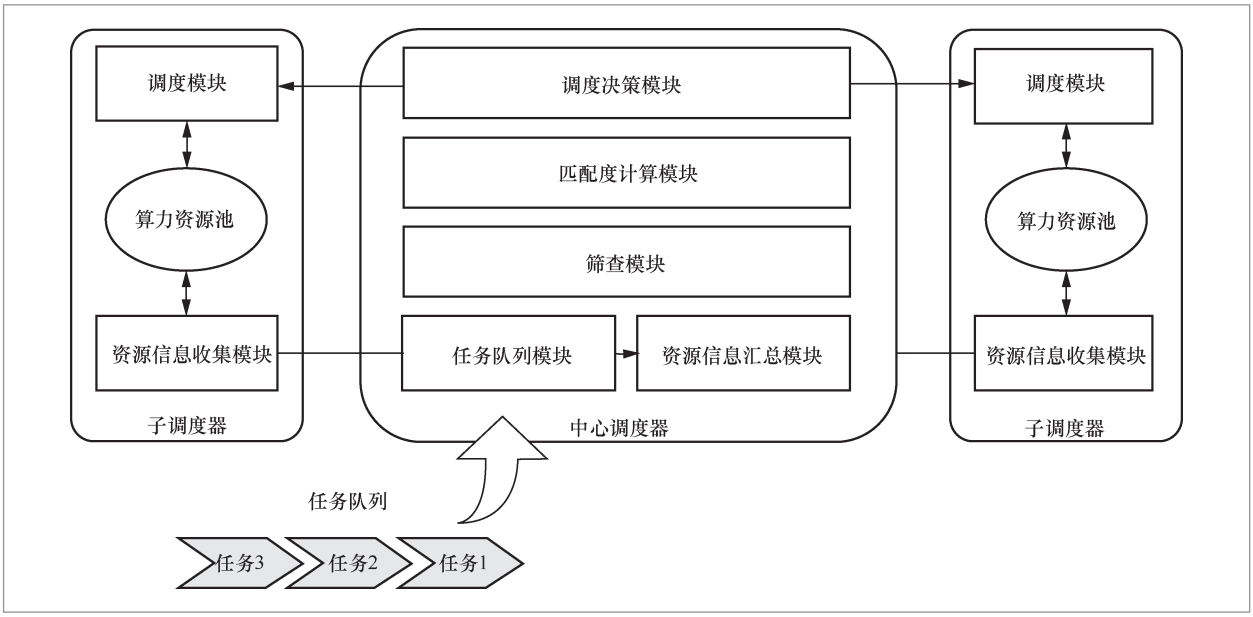


图 2 任务与资源双向匹配原型系统设计

述，然后更新资源信息。

● 筛查模块：根据任务需求模型和资源能力模型的输入，对二者进行初步筛查，去除不符合条件的配对，从而简化后续的匹配过程。

● 匹配度计算模块：在初步筛查后，该模块对剩余的任务–资源配对分别计算任务向匹配度和资源向匹配度，并生成这些配对的双向综合匹配度。

● 调度决策模块：在生成任务需求和资源能力间的双向综合匹配度后，该模块对于系统中的任务，按其与所有资源的双向综合匹配度的降序排列生成该任务的调度目标集合，并从中选取适合的调度目标生成调度决策。

● 调度模块：该模块对被分配至本地算力中心的任务实施进一步的编排和控制，是调度决策的实际执行者。

● 资源信息收集模块：该模块定时收集本地算力中心的资源信息，并发送给中心调度器中的资源信息汇总模块，以更新资源信息。

3 实验结果与分析

3.1 实验环境

本文通过广域分布式计算环境中多样的任务需求和资源能力的匹配任务，测试DBMM在响应延迟和匹配效果等方面的性能。如**图3**所示，实验中的广域分布式计算环境由地理位置分散的5个算力中心构成，各算力中心之间的平均网络带宽在图中进行了标注。

在构建的广域分布式计算环境中，各算力中心提供多样的存储、计算、网络等资源能力，见**表1**。

本文在多个标准的MPI并行计算任务

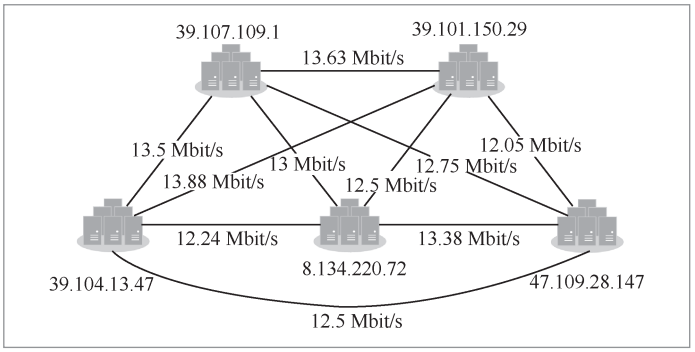


图 3 构建的广域分布式计算环境

表 1 测试环境

算力中心	总计算核数/个	总存储容量/GB
算力中心1（IP：39.107.109.1）	40	35
算力中心2（IP：39.101.150.29）	22	30
算力中心3（IP：39.104.13.47）	40	15
算力中心4（IP：8.134.220.72）	28	40
算力中心5（IP：47.109.28.147）	35	35

及其相应输入数据上进行实验，每个任务所需的计算核数为2~8个，所需的输入数据大小范围为2~200 MB。本实验将所有输入数据都部署到算力中心1中，在任务执行阶段通过节点间的链路按需对输入数据进行迁移。

本实验以典型的静态匹配方法——多对多匹配算法（many-to-many matching algorithm，MMMA）[14]和动态匹配方法——数据放置和任务调度协同算法（collaborative algorithm for data placement and task scheduling，CADT）[18]为对比基准算法。MMMA面向云网融合场景下雾计算的快速处理需求，为资源提供方（即算力中心）和资源使用方（即任务）规划合理的匹配方案。MMMA以降低任务完成时间和提升资源利用率为目标，对匹配方案进行迭代优化，从而提升匹配效果。CADT面向边缘计算场景下的用户体验，基于任务需求和资源能力间的相似程

度定义匹配度，并根据优先级和匹配度将任务分配至适合的资源以提升系统效率。同时，本实验还采用了笔者之前的研究工作——动态双向匹配（dynamic two-side matching，DTSM）[28]算法作为对比基准算法之一。DTSM算法分析任务和资源间的匹配度，针对每个任务生成优选调度目标集合，并根据资源状态动态地调整优选调度目标集合中各算力中心的优先级。相较于DTSM算法，本文提出的DBMM在匹配度定义中考虑了网络资源的因素，并简化了资源选择阶段，进一步提升了匹配效果并降低响应延迟。DBMM与各基准算法的特征见**表2**。

3.2 实验结果分析

本实验从资源匹配效果和响应延迟等方面分析了DBMM的匹配效率。匹配效果表示匹配方法生成的匹配方案能否保障应用的计算效率，本实验通过系统的总体任务完成时间量化分析匹配效果。响应延迟可反映用户提交的任务在系统中的滞留时间，是影响用户体验的重要因素。

3.2.1 匹配效果

本实验在广域分布式计算环境中分别提交10、20、30、40、50个任务（对系统施加不同情况的负载），并分别采用不同匹配方法进行匹配，然后根据各匹配方法生成的匹配方案执行任务，并统计系统总体任务完成时间。由于各组实验的执行相互隔离，即每组实验对资源完全解除占用后才会开始下一组实验，因此本实验认为各组实验中的计算、存储、网络等资源的初始条件是一致的。各算法在匹配效果方面的表现如**图4**所示。

如**图4**所示，在各种负载情况下，采用DBMM的总体任务完成时间均少于其他对比算法。其中，当任务数量为30时，采用DBMM的总体任务完成时间的优化效果最明显，分别优于采用MMMA、CADT和DTSM算法41.83%、72.36%和28.31%。因此，DBMM的匹配效果优于其他对比算法。

MMMA是一种以优化匹配效果为主要目标的静态匹配方法，因此MMMA在总体任务完成时间上的表现仅次于DBMM。然而，静态匹配方法适用于已知其计算和通信行为的静态场景，因此在广域分布式计算环境中，MMMA为了获取更优的匹配效果，需通过广域网环境多次收集任务需求和资源信息以支撑迭代优化求解过程。但是这种广域网收集信息的大量时间开销可能引起资源信息收集滞后、任务积压等，从而降低匹配效果。CADT是一种适用于广域场景的动态匹配方法，为了满足用户对QoS的需求而在匹配效果方面进行了权衡。CADT以任务需求和资源能力间的相似程度为匹配度定义，通过保障单个算力中心内计算和存储资源的均衡利用以保障

表2 各算法的特征

算法名称	匹配类型		匹配特征		
	静态匹配	动态匹配	响应延迟低	匹配效果好	双向匹配
MMMA	√			√	
CADT		√	√		
DTSM		√	√	√	√
DBMM		√	√	√	√

执行效率。然而这类匹配度定义可能造成全局资源负载不均衡、无法充分利用，因此，CADT在总体任务完成时间上的表现相对较差。在DTSM算法中，任务与资源间的匹配度是基于任务的存储、计算资源需求和算力中心的存储、计算资源负载情况定义的，然而DTSM算法未考虑网络资源情况可能造成的影响。在真实的广域分布式计算环境中，复杂多变的广域网络可能造成调度信息延迟大、数据迁移迟缓等问题。此外，DTSM算法在资源选择阶段采取的动态更新资源信息、调整替换优选调度目标等机制会导致调度响应延迟小幅度增长，但在广域网络性能极不稳定、资源状态动态变化的情况下，这种响应延迟增长也可能影响到匹配效果。而本文提出的DBMM在匹配度定义时更加全面地考虑了网络资源因素，并且简化了冗余的资源选择阶段。DBMM充分考虑计算、存储、网络等资源情况，采用匹配度感知的双向匹配，从任务视角出发注重提升执行效率、从资源视角出发注重全局负载均衡，在避免复杂匹配的同时保障匹配效果，匹配效果优于其他方法。

3.2.2 平均响应延迟

本实验统计了不同负载情况下系统中任务的平均响应延迟作为实验结果。平均响应延迟反映了任务在中心级调度器中滞留的时间，即从任务被提交到任务调度方案产生的时间。各算法在平均响应延迟上的表现如**图5**所示。

如**图5**所示，在大部分负载情况下，采用DBMM的平均响应延迟均少于其他对比算法。其中，当任务数量为20时，采用DBMM的平均响应延迟的优化效果最明显，分别优于采用MMMA、CADT和DTSM算法29.18%、12.08%和7.09%；当

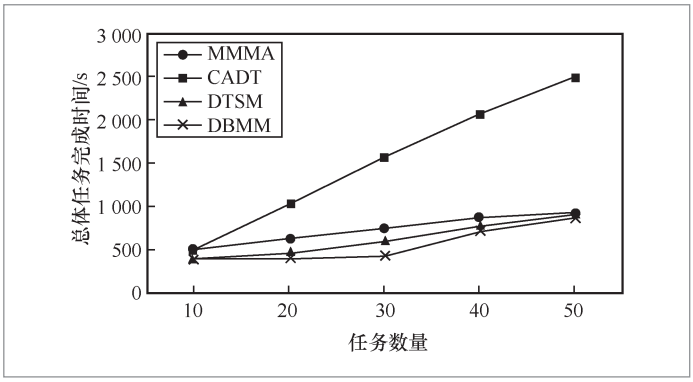


图 4　匹配效果实验结

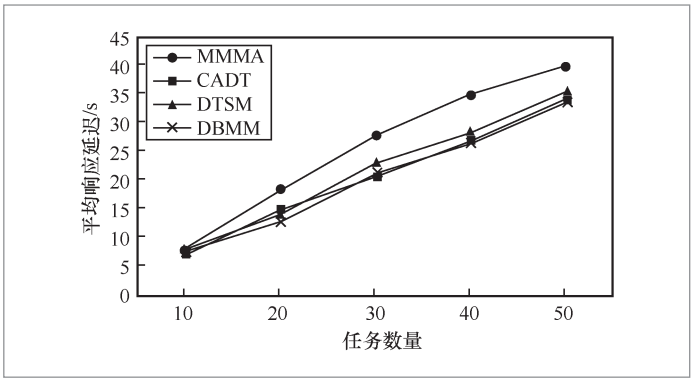


图 5　平均响应延迟实验结果

任务数量为10时，DBMM的平均响应延迟的优化效果最不明显，分别优于采用MMMA和DTSM算法4.87%和3.71%，弱于采用CADT算法8.33%。因此，DBMM的平均响应延迟与CADT相似，且优于DTSM算法和MMMA。

如**表2**中的分析，CADT、DTSM算法、DBMM作为动态匹配方法，采用匹配度感知的形式生成匹配方案，更加适用于广域分布式计算环境中满足用户的快速响应需求，在平均响应延迟方面都具有较优的性能。但DTSM算法在资源选择阶段为保障匹配效果，在调度过程中根据资源负载情况动态更新任务的优选调度目标集合，在广域网环境中动态收集资源负载情况在一定程度上会增加响应延迟，因此DTSM算法的平均响应延迟实验结果略逊于CADT和

DBMM。而MMMA作为静态匹配方法，在匹配过程中为了保障匹配效果需通过广域网收集任务和资源信息并进行多轮迭代优化，因此在平均响应延迟方面的表现较差。综上所述，本文提出的DBMM相较于其他对比基准算法可提升匹配效果并降低平均响应延迟，进而实现广域分布式计算环境中多样任务需求和资源能力的高效匹配。

4 结束语

针对广域分布式计算环境中任务需求和资源能力多样化导致匹配效率低的问题，本文提出了任务与资源动态双向匹配方法。本文提出的DBMM通过分析多样化任务需求和资源能力的模型化描述，建立了统一的任务需求模型和资源能力模型，简化了匹配过程。此外，DBMM定义了任务向匹配度和资源向匹配度以表述任务视角和资源视角的匹配偏好，基于二者的双向一致性和双向互补性定义了任务和资源的双向综合匹配度，从而优化匹配效果。最后，DBMM基于双向综合匹配度生成任务调度决策，并动态反馈资源信息以支持后续匹配过程。本文提出的方法充分考虑了全局任务需求和资源能力之间的双向综合匹配度，同时避免了复杂的匹配过程，从而实现了广域环境中多样化任务需求和资源能力的高效匹配。实验结果表明，本文方法可有效提升匹配效果并大幅降低平均响应延迟，进而实现了广域分布式计算环境中任务需求和资源能力间的高效匹配。

参考文献:

[1] CHEN Q, ZHENG Z M, HU C, et al. On-edge multi-task transfer learning: model and practice with data-driven task allocation[J]. IEEE Transactions on Parallel and Distributed Systems, 2020, 31(6): 1357-1371.

[2] TOWNS J, COCKERILL T, DAHAN M, et al. XSEDE: accelerating scientific discovery[J]. Computing in Science & Engineering, 2014, 16(5): 62-74.

[3] GAGLIARDI F. The EGEE European grid infrastructure project[M]//High Performance Computing for Computational Science - VECPAR 2004. Heidelberg: Springer 2005: 194-203.

[4] DEPEI Q. CNGrid: a test-bed for Grid technologies in China[C]//Proceedings of 10th IEEE International Workshop on Future Trends of Distributed Computing Systems. Piscataway: IEEE Press 2004: 135-139.

[5] KANG S, VEERAVALLI B, AUNG K M M. Dynamic scheduling strategy with efficient node availability prediction for handling divisible loads in multi-cloud systems[J]. Journal of Parallel and Distributed Computing, 2018, 113: 1-16.

[6] CHOUDHARY A. A walkthrough of Amazon elastic compute cloud (amazon EC2): a review[J]. International Journal for Research in Applied Science and Engineering Technology, 2021, 9(11): 93-97.

[7] 王建冬，于施洋，窦悦. 东数西算：我国数据跨域流通的总体框架和实施路径研究[J]. 电子政务, 2020(3): 13-21.
WANG J D, YU S Y, DOU Y. East digital computing and west computing: research on the overall framework and implementation path of cross-domain data circulation in China[J]. E-Government, 2020(3): 13-21.

[8] 高文. 中国算力网的机遇与挑战[J]. 中国计算机学会通讯, 2023, 1: 1-6.
GAO W. The opportunities and challenges of China's computing power network[J]. Communictions of the CCF, 2023(2): 1-6.

[9] 钱德沛，栾钟治，刘轶. 从网格到“东数西算”：构建国家算力基础设施[J]. 北京航空航天大学学报, 2022, 48(9): 1561-1574.

QIAN D P, LUAN Z Z, LIU Y. From grid to "East-west Computing Transfer": constructing national computing infrastructure[J]. Journal of Beijing University of Aeronautics and Astronautics, 2022, 48(9): 1561-1574.

[10] XU K, LYU L, LI T, et al. Minimizing tardiness for data-intensive applications in heterogeneous systems: a matching theory perspective[J]. IEEE Transactions on Parallel and Distributed Systems, 2020, 31(1): 144-158.

[11] 毕娅, 原惠群, 初叶萍, 等. 大数据环境下基于公共服务平台的资源多级智能寻租与匹配策略和价值创造[J]. 计算机科学, 2019, 46(2): 42-49.
BI Y, YUAN H Q, CHU Y P, et al. Multilevel and intelligent rent-seeking and matching resource strategy and value creation of public service platform in big data environment[J]. Computer Science, 2019, 46(2): 42-49.

[12] ZHAO L P, YANG Y N, MUNIR A, et al. Optimizing geo-distributed data analytics with coordinated task scheduling and routing[J]. IEEE Transactions on Parallel and Distributed Systems, 2020, 31(2): 279-293.

[13] HE L, QIAN Z C. Intent-based resource matching strategy in cloud[J]. Information Sciences, 2020, 538: 1-18.

[14] RAVEENDRAN N, ZHANG H Q, SONG L Y, et al. Pricing and resource allocation optimization for IoT fog computing and NFV: an EPEC and matching based perspective[J]. IEEE Transactions on Mobile Computing, 2022, 21(4): 1349-1361.

[15] FENG W J, ZHENG J L, JIANG W H. Joint pilot and data transmission power control and computing resource allocation algorithm for massive MIMO-MEC networks[J]. IEEE Access, 2020, 8: 80801-80811.

[16] CHEN Y F, LI Z Y, YANG B, et al. A Stackelberg game approach to multiple resources allocation and pricing in mobile edge computing[J]. Future Generation Computer Systems, 2020, 108: 273-287.

[17] LI B D, YANG Y, SU J F, et al. Two-sided matching decision-making model with hesitant fuzzy preference information for configuring cloud manufacturing tasks and resources[J]. Journal of Intelligent Manufacturing, 2020, 31(8): 2033-2047.

[18] LI C L, BAI J P, TANG J H. Joint optimization of data placement and scheduling for improving user experience in edge computing[J]. Journal of Parallel and Distributed Computing, 2019, 125: 93-105.

[19] CHEN L T, CHEN S Q. Volunteer multi-person multi-task optimization dispatch method considering two-sided matching[J]. Soft Computing, 2022, 26(8): 3837-3861.

[20] WANG W, LI B C, LIANG B, et al. Multi-resource fair sharing for datacenter jobs with placement constraints[C]// Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. Piscataway: IEEE Press, 2016: 1003-1014.

[21] LEONG S H, PARODI A, KRANZLMÜLLER D. A robust reliable energy-aware urgent computing resource allocation for flash-flood ensemble forecasting on HPC infrastructures for decision support[J]. Future Generation Computer Systems, 2017, 68: 136-149.

[22] KAMECKE U, ROTH A E, SOTOMAYOR M A O. Two sided matching: a study in game-theoretic modeling and analysis[J]. Economica, 1992, 59(236): 487.

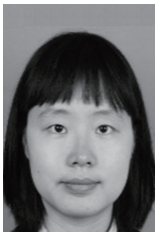
[23] MA Y T, WANG H J, XIONG J, et al. Joint allocation on communication and computing resources for fog radio access networks[J]. IEEE Access, 2020, 8: 108310-108323.

[24] YUAN Y L, YANG T, HU Y L, et al. Two-timescale resource allocation for cooperative D2D communication: a matching game approach[J]. IEEE Transactions on Vehicular Technology, 2021, 70(1): 543-557.

[25] DING D, FAN X C, LUO S W. User-

oriented cloud resource scheduling with feedback integration[J]. The Journal of Supercomputing, 2016, 72(8): 3114-3135.
[26] WANG Y F, LIU J, TONG Y, et al. Resource scheduling in mobile edge computing using improved ant colony algorithm for space information network[J]. International Journal of Satellite Communications and Networking, 2023, 41(4): 331-356.
[27] SHEN H X, LI S G, LIANG Y Y. Faster algorithms for bicriteria scheduling of identical jobs on uniform machines[J]. Journal of Industrial and Management Optimization, 2023, 19(7): 5398-5406.
[28] SONG Y, WANG L, XIAO L M, et al. Dynamic two-side matching of tasks and resources in wide-area distributed computing environments[J]. The Journal of Supercomputing, 2023, 79(9): 10208-10231.
[29] LI C Y, TANG X Y. On fault-tolerant Bin packing for online resource allocation[J]. IEEE Transactions on Parallel and Distributed Systems, 2020, 31(4): 817-829.
[30] KHIYAITA A, EL BAKKALI H, ZBAKH M, et al. Load balancing cloud computing: state of art[C]//Proceedings of the 2012 National Days of Network Security and Systems. Piscataway: IEEE Press, 2012: 106-109.

作者简介



尚晶(1978-),女,博士,中国移动信息技术中心高级工程师,中国移动首席专家,主要研究方向为大数据、云计算、数据库。

肖利民(1970-),男,博士,北京航空航天大学教授、博士生导师,计算机科学技术系主任,系统结构研究所所长,中国计算机学会大数据专委会委员、高性能计算专委会委员、容错计算专委会委员,中国电子学会云计算专委会委员,主要研究方向为计算机体系结构、大数据存储、高性能计算等。曾获国家科技进步二等奖4项、省部级科技一等奖4项及其他省部级奖5项。

肖智文(1993-),男,博士,中国移动信息技术中心中级工程师,主要研究方向为大数据、机器学习、云计算。

王锦权(1998-),男,北京航空航天大学博士生,主要研究方向为分布式存储系统、分布式调度系统、高性能计算等。

武智晖（1978– ），男，中国移动信息技术中心高级工程师、架构师，主要研究方向为大数据平台、数据处理、数据库。

李辉阳（2000– ），男，北京航空航天大学硕士生，主要研究方向为分布式调度系统、大数据存储等。

张逸飞（1996– ），男，博士，中国移动信息技术中心中级工程师、项目经理，主要研究方向为大数据、机器学习、云计算。

宋尧（1994– ），男，博士，中国信息通信研究院技术与标准研究所中级工程师，主要研究方向为高性能计算、边缘计算、算网融合、分布式存储、分布式调度系统、存算联动调度等。

王冀彬（1980– ），男，中国移动信息技术中心高级工程师、大数据事业部总经理，主要研究方向为大数据、数据分析。

收稿日期： 2024-05-20

通信作者： 宋尧，songyao@caict.ac.cn

基金项目： 国家重点研发计划项目（No.2023YFB4503100）；国家自然科学基金项目（No.U23B2027）；中国移动“联创+”资助项目（No.R23103E4）

Foundation Items: The National Key Research and Development Program of China(No.2023YFB4503100), The National Natural Science Foundation of China (No.U23B2027), China Mobile Joint R&D Project (No.R23103E4)