

基于博弈合作的跨服务协同分配研究

张 笛¹
ZHANG Di

摘 要

随着城市服务需求的多样化, 跨服务场景中资源配置和任务协同关系日益紧密, 文章针对传统单一任务分配模式无法适应跨服务协同需求的问题, 提出了一种基于博弈论与合作机制的跨服务协同分配方法, 旨在解决多服务间的资源分配与任务协同问题。通过引入合作质量得分, 量化跨服务任务的协同打包效果, 并通过设计合作意识驱动的博弈匹配算法 (CGTM), 实现纳什均衡解。实验证明, 该算法在提高总收益和缩短响应时间方面均优于传统方法, 能够显著提升资源利用效率与公平性。此外, 文章还提出了提前终止准则和玩家最佳响应策略的延迟更新策略, 以进一步加速算法收敛, 确保在动态环境中的灵活性与高效性。

关键词

城市计算; 跨服务任务; 博弈论; 协同分配; 纳什均衡

doi: 10.3969/j.issn.1672-9528.2025.02.035

0 引言

在现代智能城市建设中, 如何实现多服务之间的资源高效配置和协同分配已成为城市计算领域的重要研究课题。随着城市服务需求日益多样性, 不同业务场景之间的资源需求和任务协同关系愈发紧密, 如何在确保服务效率的同时, 提高资源利用率和城市运行效率已成为亟待解决的关键问题。随着智能城市服务对跨服务协同的需求日益凸显, 诸多研究者们开始探索多服务间的协作分配模型, 以有效应对资源分布动态变化的挑战。但在城市计算场景中, 服务需求多样且具有动态性, 传统的单一业务分配模式已难以满足跨服务协同需求下的高效资源调配。

在此背景下, 出行与物流行业的协同需求显得尤为突出, 特别是城市出行服务和物流配送的交叉应用不断增加。例如, 基于乘车共享的包裹派送模式^[1-2], 实现了包裹和乘客共用同一车辆资源, 这种创新模式有效提升了物流运输效率, 同时降低了整体运营成本。Chen 等人^[3]通过在出租车与乘客之间建立中继机制, 提出了一种同时满足包裹运输需求且不影响乘客满意度的解决方案。该方法在沿途设置寄收仓库作为中转站, 使包裹借助多辆出租车的协同配合经由多个中转站点进行传递与配送。然而, 现有方法多偏重于单一业务的优化策略^[4], 难以平衡不同服务间的资源需求, 从而影响跨服务场景下的协作效率和公平性。此外, 物流配送中的众包模式^[5]也显示出在协作性和成本效益方面的优势, 但现有方

法多依赖固定仓储等设施, 灵活性差, 无法应对动态环境下服务需求的变化。

为了更好地实现跨服务的协同分配, 本文提出了一种基于博弈论与合作机制的跨服务协同分配方法, 以解决多服务间的资源分配与任务协同问题。具体而言, 本文利用博弈论建模服务间的合作关系, 通过合作博弈策略在满足不同服务需求的同时实现最大化资源分配效率。该方法旨在实现服务间需求的动态调整, 并通过优化合作模型确保在动态环境中资源的最优配置。相比传统的单一业务分配方案, 本文方法能够动态应对多服务需求变化, 提高资源利用公平性和高效率, 解决传统方法在跨服务协作中缺乏灵活性的问题。

1 合作感知的博弈协作匹配算法

为解决跨服务任务分配过程中因时空约束带来的效率瓶颈, 本文提出了一种改进的跨服务任务打包和分配策略。通过引入跨服务任务的合作质量得分, 对任务间的协同打包进行量化评价, 并结合博弈论中的效用函数, 构建打包与分配流程的统一框架。该框架不仅可以在任务分配中实现不同类型任务的合理打包, 还能在分配时保障工人 (如司机、快递员等) 之间的公平性。本节详细介绍了合作质量得分的计算方法, 阐述基于合作感知的博弈协作匹配算法, 并提出针对该算法的两个优化策略, 进一步提升其效率和公平性。

1.1 跨服务任务合作得分

在跨服务任务共享与分配系统中, 每个工人 w_i 在时间片 t 时进入系统, 报告其当前位置 $w_i.l_w$ 。工人倾向于选择距离较近的任务, 以减少不必要的移动, 因此更愿意接受位于其工作半径 r_i 范围内的跨服务任务。跨服务任务之间存在一个

1. 郑州财税金融职业学院信息技术系 河南郑州 450048

[基金项目] 河南省高等学校重点研究项目计划支持 (23B1520048)

合作质量得分 q ，且该得分因工人而异。具体而言，合作质量得分 q 包含两部分：一是历史平均合作质量得分，由系统基于以往任务完成情况生成。工人在完成特定区域的跨服务任务后会对其体验进行评分，从而得到该区域的平均合作质量得分。例如，若工人 w_i 完成了路网节点 l_i 和 l_j 附近的两个跨服务任务，可对这两个任务的合作体验分别评分，以供系统记录。二是实时位置合作质量得分，由系统根据工人与任务之间的相对位置动态计算。当某一时间片内出现多个跨服务任务时，系统依据任务与工人的相对距离生成合作质量得分，距离较近的任务对该工人的合作质量得分更高。

针对某个工人 w_i ，跨服务任务 τ_m 和 τ_n 的合作质量得分表示为：

$$q_{w_i}(\tau_m, \tau_n) = \alpha \cdot \exp_{mn} + (1 - \alpha) \cdot \text{loc}_{mn} \quad (1)$$

式中： $\exp_{mn}$ 为 τ_m 和 τ_n 间的历史平均合作质量得分； loc_{mn} 为两任务间的位置合作质量得分； α 为控制分数比例的权重系数。合作质量得分 $q_{w_i}(\tau_m, \tau_n)$ 表示两项任务间的协同打包质量，得分越高，说明两项任务越适合合并分配。多项跨服务任务间的总合作质量得分 Q 通过累加计算得到，用公式表示为：

$$Q = \sum q_{v_i} \quad (2)$$

1.2 CGTM 算法

博弈论作为一种数学工具，用于分析智能体之间的冲突与协作关系，广泛应用于经济、政治和计算机科学等多个领域。在策略博弈中，参与者（工人）组成的群体 N 会彼此竞争或合作以优化其效用。每位玩家 $i \in N$ 可以从策略集合 S_i 中选择某一策略 s_i （即执行指定的跨服务任务），并通过效用函数 U_i 来衡量其收益。效用函数 U_i 的输入为联合策略 $S \in \mathbb{S}$ ，该策略集合是所有玩家策略的组合。假设 s_i 表示玩家 i 在联合策略中的选择，而 s_{-i} 代表其他所有玩家的策略集合。当策略 S 满足以下条件时，策略博弈达到纯纳什均衡：

$$U_i(s_i^*, s_{-i}^*) \geq U_i(s_i, s_{-i}^*), \forall s_i \in S_i \quad (3)$$

即当其他玩家的策略保持不变时，没有玩家能通过单方面调整策略来提升效用。

本文将跨服务任务分配问题建模为策略博弈 G ，并设计了一种合作意识驱动的博弈匹配方法（cooperation-aware game theoretical matching, CGTM）以获得纳什均衡解，从而为每位工人分配最优的跨服务任务组，提高整体合作质量。具体来说，工人 w_i 被视为博弈中的玩家，其目标是选择最佳效用的任务组。对每位玩家 w_i 而言，其策略集 S_i 包含所有可能的可执行任务组合。策略博弈 G 的联合策略 S 映射到跨服务任务分配的分配 $\mathcal{M}_b$ 。在联合策略 S 下，玩家 w_i 的效用函数 U_i 表达式为：

$$U_i(S) = \begin{cases} R(w_i, \mathcal{K}_n(w_i)), |\mathcal{K}_n(w_i)| = 1 \\ R(w_i, \mathcal{K}_n(w_i)) + Q(\mathcal{K}_n(w_i)), |\mathcal{K}_n(w_i)| \geq 2 \\ 0, |\mathcal{K}_n(w_i)| = 0 \end{cases} \quad (4)$$

式中： $\mathcal{K}_n(w_i)$ 表示联合策略 S 下分配给 w_i 的任务集。若

$\mathcal{K}_n(w_i)$ 的任务数量为 0，则效用为 0；若为 1，则通过收益函数 $R(w_i, \mathcal{K}_n(w_i))$ 计算效用；若数量大于等于 2，则效用包括收益 $R(w_i, \mathcal{K}_n(w_i))$ 和任务合作质量得分 $Q(\mathcal{K}_n(w_i))$ 的组合。算法 1 详细展示了 CGTM 的核心思想。

算法 1 CGTM 算法
Input: 任务集合 O 包括 P_r 、 D_r ，玩家集合 W
Output: 纳什均衡
1: 随机初始化跨服务任务组合和玩家 - 任务分配;
2: while 不是纳什均衡 do
3: foreach $w_i \in W$ do
4: 为玩家 w_i 选择最佳响应的跨服务任务 $\{p_i, d_i\}$;
5: 将跨服务任务组 $\{p_i, d_i\}$ 分配给玩家 w_i ;
6: end for
7: end while
Return 玩家 - 跨服务任务分配集合

图 1 为 CGTM 算法的流程示例，其中包含两位玩家（ w_1 和 w_2 ）、两位乘客请求（ p_1 和 p_2 ）以及两个包裹请求（ d_1 和 d_2 ）。图 1（a）展示了两位玩家的观察范围，即他们可以服务的区域。图 1（b）和图 1（c）分别展示了玩家 w_1 和 w_2 的视角下，不同任务组的合作质量得分情况。

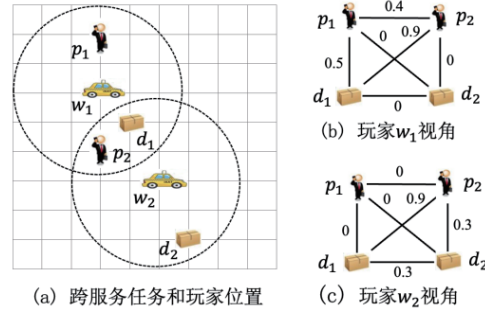


图 1 CGTM 算法运行实例图

从图 1（b）和图 1（c）可以看出， $\{p_1, p_2, d_1\}$ 打包分配给 w_1 而 $\{d_1\}$ 分配给 w_2 是一种较优选择，因为 $\{p_1, p_2, d_1\}$ 的合作质量得分为 1.8（0.9+0.4+0.5），而 $\{p_2, d_1, d_2\}$ 的得分仅为 1.5（0.9+0.3+0.3）。这种分配基于贪婪策略，虽然计算效率高，但这种方式无法保证玩家间的公平。

CGTM 算法通过结合跨服务任务间的合作质量得分，以最佳响应框架不断调整玩家的任务组合，以接近纳什均衡状态。尽管在纳什均衡下的方案可能并非为最高分，但确保了相对的公平性。最终，当跨服务任务完成合理打包时，分配过程也自动完成。如图 1 所示，基于公平原则， w_1 的分配方案为 $\{p_1, d_1\}$ ，而 w_2 则分配到 $\{p_2, d_2\}$ 。

2 CGTM 算法优化

为提升算法 1 中所示的 CGTM 算法在大规模跨服务任务分配问题上的求解效率，本文提出了两项优化策略，以加速收敛至纳什均衡，同时保持合理的效用水平。

2.1 提前终止准则

CGTM 算法属于可中断型算法,意味着在任何阶段中断时,均能输出有效解。随着算法的迭代进行,联合策略的总效用(包括收益和合作得分)逐渐提高。然而,实验中观察到每轮总效用的提升幅度在接近收敛时逐渐减小(算法1中的第3~6行)。因此,在实际应用中,可以引入一个终止条件,即当每轮总效用的增加小于设定的阈值 $\varepsilon \cdot U(S)$ 时停止迭代。其中, $U(S)$ 表示当前总效用值, ε 为超参数。此方法可显著减少CGTM算法的运行时间,而仅对总效用造成轻微影响。

2.2 玩家最佳响应策略的延迟更新

在CGTM算法中,每一轮都需要计算每位玩家的最佳响应策略(算法1中的第4行),但并非所有玩家的策略都需要频繁更新。因此,可采用延迟更新策略,仅在玩家策略可能发生变化时,重新计算其最佳响应。因此,在必要时再更新玩家 w_i 的最佳响应策略是有效的。这一优化减少了不必要的计算,从而加快算法收敛。

3 实验评估

3.1 实验设置

为评估本文提出的合作感知的博弈协作匹配(CGTM)算法的效率和质量,本节将在总收益和响应时间这两个指标上进行实验对比。实验使用的真实出租车数据集包括来自滴滴公司成都市和NYCTaxi公司纽约市的数据,其中成都数据集含有209 423个订单、36 630个节点和50 786条边,纽约数据集则包含10 906 858个订单、264 346个节点和366 923条边。对比算法包括基于博弈论的任务分配算法CTMG^[6-7]、基于Dijkstra的路径规划算法HBSP^[8]以及基于强化学习的自适应路径规划算法ASP^[9-10]。评价指标为总收益和响应时间。所有算法均在Python环境下实现,使用TensorFlow 1.15.0框架,并在配置为Intel i7-9700K @ 3.6 GHz CPU和16 GB RAM的计算机上运行。

3.2 实验分析

(1) 成都数据集

如图2所示,当工人数量从20增加至300时,所有算法的总收益均呈现上升趋势。特别是在工人数量从20提升到100的阶段,各算法的总收益增长显著;而当工人数量进一步增加至300时,总收益的增长幅度则减缓。这一现象可归因于在前者阶段,跨服务任务的数量较为充足,工人数量的增加使得更多的任务得以完成,从而提升了总收益。在工人数量从100到300的过程中,剩余的跨服务任务数量已然有限,因此尽管工人数量继续增加,任务的完成和总收益的提升效果不再显著。图3显示,随着工人数量的增加,四种算法的响应时间也持续上升。这是因为工人数量的增加导致节点数量的增加,使得资源分配过程变得更加复杂,从而增加了计算成本。

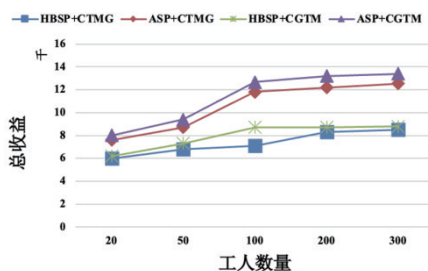


图2 工人数量与总收益的关系

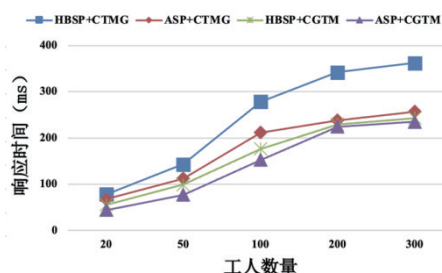


图3 工人数量与响应时间的关系

综合图2和图3的结果可知,ASP+CGTM算法在总收益和响应时间方面表现最佳,而ASP+CTMG的总收益位列第二。HBSP+CGTM则在响应时间上表现良好,HBSP+CTMG的各项指标则相对较低。这种差异源于CGTM算法结合了合作博弈论的思想,将打包和博弈过程紧密融合,相较于CTMG在这两个过程上的独立处理,能够显著减少计算时间,提高响应效率。

在图4和图5中,分别展示了总收益和响应时间在乘客请求数量从200增加到3000过程中的变化。

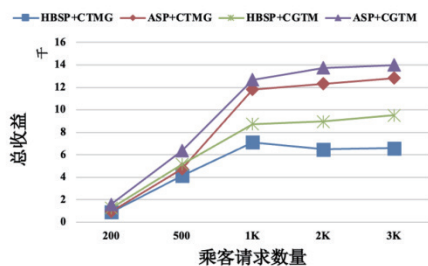


图4 乘客请求数量与总收益的关系

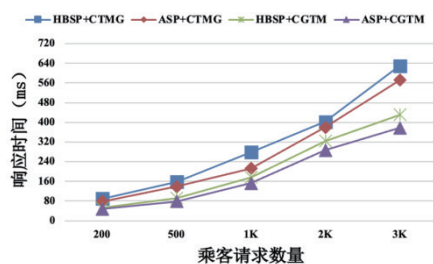


图5 乘客请求数量与响应时间的关系

图4显示,当乘客请求数量从200增加至1000时,四种算法的总收益持续上升,且增长幅度明显。然而,乘客请求数量从1000增加至3000时,各算法的收益增长趋势显著减缓。这主要是因为在前期阶段(200~1000)工人数量充足,

随着乘客任务数量的增加,工人能完成的任务增多,从而推动了总收益的提升;而在后期阶段(1000~3000),由于工人数量有限且已有任务的分配接近饱和,乘客请求的增加并未进一步推动总收益的增长。图5显示,随着乘客请求数量的上升,各算法的响应时间也呈现出逐渐增加的趋势。这一现象与图3的观察结果相似,乘客请求的增加同样导致了分配过程中节点数量的增多,从而提高了计算所需的时间成本。

(2) 纽约数据集

图6表明,当工人数量从200增加至1000时,四种算法的总收益持续增长;然而,当工人数量从1000增加至3000时,总收益的增长趋势明显减缓。这一现象表明,随着工人数量的增加,初期能够显著提高任务的完成度,从而推动总收益的上升,而在后期,由于工人数量有限,任务的增加对总收益的促进效果减弱。随着工人数量的增加,各算法的响应时间也呈持续上升的趋势。这一趋势的形成原因与图2和图3的分析结果相似,不再详细阐述。

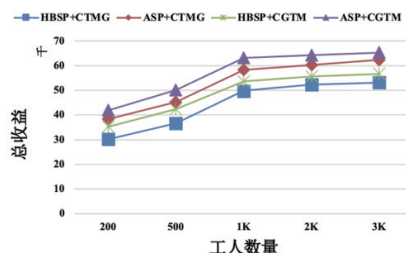


图6 工人数量与总收益的关系

从图7可以看出,当乘客请求数量增加至20000时,4种算法的总收益持续上升;然而,随着请求数量增加至30000,总收益的上升趋势显著减缓。这表明,在初期,乘客请求的增加能够有效推动总收益的增长,但在请求数量达到一定水平后,增长速度减缓。

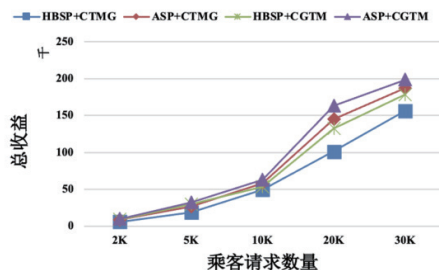


图7 乘客请求数量与总收益的关系

随着乘客请求数量增加,各算法的响应时间也持续上升。这种现象与图4和图5中的结果相似,因此不再详述。

4 总结

本文提出了一种基于博弈论与合作机制的跨服务协同分配方法,以应对现代智能城市中多服务资源高效配置的挑战。通过构建跨服务任务的合作质量得分,并首次量化了任务间的协同打包效果,通过引入了合作意识驱动的博弈协作匹配

算法(CGTM),以实现纳什均衡解并优化任务分配。实验结果表明,与传统单一业务任务分配策略相比,CGTM在总收益和响应时间方面均表现出显著优势,能够有效提升资源利用效率及公平性。此外,通过引入提前终止准则和玩家最佳响应策略的延迟更新策略,进一步加快了算法的收敛速度,确保了在动态环境下的灵活性和高效性。

参考文献:

- [1] ARSLAN A M, AGATZ N, KROON L, et al. Crowdsourced delivery: a dynamic pickup and delivery problem with ad hoc drivers[J]. Transportation science, 2019, 53(1): 1-318.
- [2] CHEN C, ZHANG D Q, MA X J, et al. Crowddeliver: olanning city-wide package delivery paths leveraging the crowd of taxis[J]. IEEE transactions on intelligent transportation systems, 2017, 18(6): 1478-1496.
- [3] CHEN L, ZHONG Q L, XIAO X K, et al. Price and time aware dynamic ridesharing[C/OL]//2018 IEEE 34th International Conference on Data Engineering (ICDE). Piscataway:IEEE,2018[2024-06-09].<https://ieeexplore.ieee.org/document/8509320>.
- [4] GONG W, ZHANG B X, LI C. Task assignment in mobile crowdsensing: Present and future directions[J]. IEEE network, 2018, 32(4): 100-107.
- [5] GAO J H, WANG Q R, ZHANG X, et al. Cooperative air-round instant delivery by UAVs and crowdsourced taxis[C/OL]//2024 IEEE 40th International Conference on Data Engineering (ICDE).Piscataway:IEEE,2024[2024-09-11].<https://ieeexplore.ieee.org/document/10598004>.
- [6] 周婷婷, 隋东, 杨振宇, 等. 基于博弈的恶劣天气下航班路径分配研究[J]. 航空计算技术, 2023, 53(6): 30-34.
- [7] 王嵩. 基于博弈论的混合车辆交互变道决策模型研究[D]. 长春: 长春工业大学, 2024.
- [8] 靳康杰. 基于Dijkstra算法的机器人最短路径规划设计研究[J]. 信息与电脑(理论版), 2024, 36(8): 47-49.
- [9] HAN H Y, WANG J Q, KUANG L Q, et al. Improved robot path planning method based on deep reinforcement learning[J]. Sensors. 2023, 23(12): 5622.
- [10] 黄昱洲, 王立松, 秦小麟. 一种基于深度强化学习的无人小车双层路径规划方法[J]. 计算机科学, 2023, 50(1): 194-204.

【作者简介】

张笛(1995—),女,河南郑州人,硕士,助教,研究方向:大数据技术、调度优化, email: se_diz@126.com.

(收稿日期: 2024-10-29)