**Relate Work**

**1.无序任务分配相关工作**

早期的MEC系统任务分配研究主要集中在为单个用户的服务上。为了确定单个用户提出的任务卸载策略，文[3]给出了一个二进制卸载算法。为了充分利用并行计算的优势，在文献[33]中研究了部分卸载问题，并给出了（1+θ）比界的近似算法。在[34]中，提供了一种可变替代技术，以联合优化负载比、传输功率和循环频率。这些工作对于处理单个用户提出的任务具有很高的性能，但是它们考虑的情况似乎有点简单。同时，他们在确定卸载策略时没有考虑数据分布。

考虑到MEC系统总是为多个用户服务，[5]将多个用户通过一个无线接入点的流量计算作为流量计算博弈，提出了一种实现纳什均衡的分散机制来求解。在文献[35]中，作者试图利用排队论来优化上下行调度。文[36]中给出了一种分解算法，用于迭代控制负载选择、时钟频率控制和传输功率分配的计算。上述算法对于处理多个用户提出的任务是有效的，但是，由于它们是为具有单个基站的MEC系统而设计的，因此不同基站之间的协作没有被充分考虑。此外，这些算法也忽略了任务分配过程中的数据分布。此外，[8]还引入了集中式和分布式贪婪最大调度算法来解决多用户多任务的流量计算问题。然而，这些算法没有考虑任务分配过程中的数据共享，也忽略了每个任务的延迟约束，因此不适合解决本文所讨论的问题。

为了增强不同基站之间的协作，文[37]提出了一种新的基站间协作方案，作者试图通过缓存流行计算任务的结果来减少响应延迟。[38]提出了一种游戏理论算法，使所有应用程序的收益最大化。以上工作充分利用了基站之间的协作。然而，这些算法也忽略了输入数据的分布，不适用于数据共享的MEC系统。[39]提供了基于支付的激励机制。该机制支持基站间的计算资源共享，并建立了一个社会信任网络来管理基站间的安全风险，但仍存在上述问题。而且，由于只考虑任务到达率，没有考虑每个任务所占用的数据和资源，对每个基站的处理能力的约束不够合理。

最近，[9]和[10]对涉及三个层次并为多个用户服务的MEC系统的体系结构进行了充分的研究。在文献[9]中，我们给出了一个针对超密集网络的启发式贪婪加载方案。在[10]中，提出了一种能够实现纳什均衡的分布式流量计算算法。这两项工作对于MEC系统都是有效的，但是它们没有考虑任务分配过程中的数据分布和任务延迟约束，因此也无法在数据共享的MEC系统中分配任务。此外，云的处理能力也没有得到充分利用。

**2．有序任务分配相关工作**

据我们所知，我们是第一个在MEC系统中研究有序任务分配算法的人。因此，在本节的其余部分，我们将总结其他分布式系统中有序任务分配方法的研究成果。

在分布式系统中，有向无环图（DAG）被广泛用于表示任务依赖关系。基于这种结构，人们提出了许多启发式和随机算法，也可以分为四类。

首先，一组算法基于优先级列表，例如HEFT[25]和HCPT[40]。这种算法具有高性能和低复杂度的特点。然而，他们假设在调度之前计算和传输时间可以提前获得，同时，他们没有考虑不同处理器的资源限制，他们的目标也不是最小化能源成本。因此，这些算法不适合在MEC系统中解决OTA问题。

其次，在[27][41]中研究了基于分组的启发式算法，该算法用于调度具有无限数量处理器的分布式系统中的有序任务，这对于MEC系统是不可行的。[42]研究了具有多个异构处理器的系统中，不同处理器的处理速度和带宽不同的任务调度问题。该算法最大限度地减少了所有有序任务的调度长度。但是，它没有考虑每个任务占用的资源和每个进程的限制，同时，它的优化目标也与我们的OTA问题不同。

为了进一步避免任务间的传输，文[43]提出了一种充分利用各处理器空闲时间的启发式方法。它允许任务复制，即一个任务可以由不同的处理器执行多次，如果它们有空闲时间。这种方法对于同构系统是有效的，这与本文讨论的MEC系统有很大的不同。同时，不同处理器的空闲时间对于MEC系统来说是不可用的，因此也不适合解决我们的OTA问题。

此外，研究人员还研究了一组基于随机搜索的有序任务调度技术，如[44]和[45]。它们大多是根据遗传算法设计的，由于会多次迭代，所以复杂度极高。因此，这些方法由于其高度的复杂性以及缺乏对每个处理器资源限制的考虑，不适合我们的OTA问题。

最后，以上讨论的相关工作也不能应用于OTA问题，因为他们所考虑的任务是无序的。

**3.定价博弈**

在移动边缘计算系统中，任务卸载和资源分配是两个主要热门的研究方向，现在已经有很多研究针对如何卸载任务，如何分配资源等问题进行了深入的研究。但是不可忽视的是，服务商提供服务也需要获得一定的收入去支撑服务，给用户提供更好的服务。同时，在ad-hoc移动边缘计算系统中，资源充足的用户给其他用户提供服务的时候也希望可以获得相应的回报。作为用户，希望可以支付尽可能少的成本去降低延迟满足延时的需求。因此，经济成本也是影响任务卸载，资源分配的重要因素。

在针对成本的方面的研究主要分为从用户角度出发和从服务提供商角度出发两个方向。

从用户角度出发，更多的计算资源服务用户的任务可以有效的减少延迟，但是这也意味着需要付出更多的成本。因此，对于给定的定价方案，用户需要选择合适的卸载任务策略，使其开销最小化。[46]将资源分配问题定义为广义纳什均衡问题，并提出了一个分布式的游戏公式算法，用户之间竞争资源且无需透露自己的私有信息，可以有效的保护用户隐私。[47]是针对ad-hoc的移动边缘计算系统，它将任务调度问题描述为一个分布式多设备任务调度博弈，提出了一种开销优化的多设备任务调度策略，该策略考虑了机会消耗，延迟，能耗，和货币支出，旨在最小化每个移动设备的开销。

从服务提供商的角度出发。定价越高，使用资源的用户就会越少。而定价太低，利润就会减少。因此，确定合适的定价方案是服务商考虑的主要问题。

现有的研究提出的定价方案主要分为统一化定价，差异化定价和拍卖式定价。

统一定价方案的研究中，文献[48]建立了用户和边缘卖家之间的Stackelberg博弈模型，并确定了一个使卖家收益最大化的最优单价。[49]中作者重点讨论了卖家向用户提供基于边缘云的缓存服务的特定情况。当提供商以固定价格购买缓存空间时，考虑了卖家和用户之间的Stackelberg均衡。在差异化定价的研究中，[50]提出了一个差异化定价方案，最大限度地提高了用户和卖家的整体幸福感。[51]在设备用户和边缘服务器联合的系统模型中考虑了差异化定价。[52] 将边缘云和用户之间的交互被建模为一个Stackelberg博弈，边缘云设置价格使其收益最大化，用户设计卸载决策，使其延迟加支付的成本最小化。并提出了两种算法针对统一定价和差异化定价的策略使边缘云收益最大化。在拍卖式定价方案中，[53]引入了一种基于出价的拍卖式定价机制，该机制考虑了通信和计算资源的联合分配。[54] 基于市场的定价和拍卖模型，提出一种激励机制，使资源提供者在移动边缘计算中获得最大的收益。在非竞争环境下，将激励机制转化为基于市场定价模型的利润最大化问题。利用凸优化方法得到最优解。在竞争环境下，提出了一个利润最大化的多轮拍卖机(PMMRA）机制。[55]还讨论了用于边缘计算的双拍卖系统。作者提出了基于盈亏平衡的双重拍卖和基于动态定价的双重拍卖两种不同的系统，并分析了它们的有效性。

[3] W. Zhang, Y. Wen, K. Guan, K. Dan, H. Luo, and D. O. Wu, “Energy-optimal mobile cloud computing under stochastic wireless channel,” IEEE TWC, vol. 12, no. 9, pp. 4569–4581, 2013.

[33] Y. H. Kao, B. Krishnamachari, M. R. Ra, and B. Fan, “Hermes: Latency optimal task assignment for resource-constrained mobile computing,”inIEEEConferenceonComputerCommunications,2015, pp. 1894–1902.

[34] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, “Mobile-edge computing:Partialcomputationoffloadingusingdynamicvoltage scaling,” IEEE TCOM, vol. 64, no. 10, pp. 4268–4282, 2016.

[5] X. Chen, “Decentralized computation offloading game for mobile cloud computing,” Parallel & Distributed Systems IEEE Transactions on, vol. 26, no. 4, pp. 974–983, 2014.

[35] M. Molina, O. Munoz, A. Pascual-Iserte, and J. Vidal, “Joint scheduling of communication and computation resources in multiuser wireless application offloading,” in IEEE International Symposium on Personal, Indoor, and Mobile Radio Communication, 2015, pp. 1093–1098. [36] S. Guo, B. Xiao, Y. Yang, and Y. Yang, “Energy-efficient dynamic offloading and resource scheduling in mobile cloud computing,” in INFOCOM, 2016, pp. 1–9.

[8] W. Chen, D. Wang, and K. Li, “Multi-user multi-task computation offloading in green mobile edge cloud computing,” IEEE Transactions on Services Computing, 2018.

[37] M. S. Elbamby, M. Bennis, and W. Saad, “Proactive edge computing in latency-constrained fog networks,” in European Conference on Networks and Communications, 2017, pp. 1–6.

[38] R. Yu, J. Ding, S. Maharjan, S. Gjessing, Y. Zhang, and D. H. K. Tsang, “Decentralized and optimal resource cooperation in geodistributed mobile cloud computing,” IEEE Transactions on Emerging Topics in Computing, vol. PP, no. 99, pp. 1–1, 2018.

[39] L. Chen and J. Xu, “Socially trusted collaborative edge computing in ultra dense networks,” in Proceedings of the Second ACM/IEEE Symposium on Edge Computing, 2017, p. 9.

[9] H. Guo, J. Liu, and J. Zhang, “Computation offloading for multiaccess mobile edge computing in ultra-dense networks,” IEEE Communications Magazine, vol. 56, no. 8, pp. 14–19, 2018.

[10] L. Tang and S. He, “Multi-user computation offloading in mobile edge computing: A behavioral perspective,”IEEENetwork,vol.32, no. 1, pp. 48–53, 2018.

[25] H.Topcuoglu,S.Hariri,andM.-Y.Wu,“Performance-effectiveand low-complexity task scheduling for heterogeneous computing,” IEEE Transactions on Parallel & Distributed Systems, vol. 13, no. 3, pp. 260–274, 2002.

[26] G. Xie, R. Li, and K. Li, “Heterogeneity-driven end-to-end synchronized scheduling for precedence constrained tasks and messages on networked embedded systems,” Journal of Parallel and Distributed Computing, vol. 83, pp. 1–12, 2015.

[27] T. Yang and A. Gerasoulis, “Dsc: scheduling parallel tasks on an unbounded number of processors,” IEEE Transactions on Parallel & Distributed Systems, vol. 5, no. 9, pp. 951–967, 2002.

[40] A. A. Nasr, N. A. Elbahnasawy, A. Elsayed, A. A. Nasr, N. A. Elbahnasawy, and A. Elsayed, “Task scheduling optimization in heterogeneous distributed systems,” vol. 107, no. 4, pp. 05–12, 2014.

[41] Y. K. Kwok and I. Ahmad, “Dynamic critical-path scheduling: an effective technique for allocating task graphs to multiprocessors,” IEEE Transactions on Parallel & Distributed Systems, vol. 7, no. 5, pp. 506–521.

[42] H. Kanemitsu, M. Hanada, and H. Nakazato, “Clustering-based task scheduling in a large number of heterogeneous processors,” IEEE Transactions on Parallel & Distributed Systems, vol. 27, no. 11, pp. 3144–3157, 2016.

[43] I. Ahmad and Y. kwong Kwok, “On exploiting task duplication in parallel program scheduling,” IEEE Transactions on Parallel & Distributed Systems, vol. 9, no. 9, pp. 872–892. [44] E. S. H. Hou, N. Ansari, and H. Ren, “Genetic algorithm for multiprocessor scheduling,” IEEE Trans Parallel & Distributed Systems, vol. 5, no. 2, pp. 113–120, 1994.

[45] Y. Xu, K. Li, J. Hu, and K. Li, “A genetic algorithm for task scheduling on heterogeneous computing systems using multiple priority queues,” Information Sciences, vol. 270, pp. 255–287, 2014.

[46] Zaw C W, Ei N N, Im H Y R, et al. Cost and latency tradeoff in mobile edge computing: A distributed game approach[C]//2019 IEEE International Conference on Big Data and Smart Computing (BigComp). IEEE, 2019: 1-7.

[47] Tianze L, Muqing W, Min Z, et al. An overhead-optimizing task scheduling strategy for ad-hoc based mobile edge computing[J]. IEEE Access, 2017, 5: 5609-5622.

[48] Kim S H, Park S, Chen M, et al. An optimal pricing scheme for the energy-efficient mobile edge computation offloading with OFDMA[J]. IEEE Communications Letters, 2018, 22(9): 1922-1925.

[49] De Pellegrini F, Massaro A, Goratti L, et al. A pricing scheme for content caching in 5G mobile edge clouds[C]//2016 International Conference on Wireless Networks and Mobile Communications (WINCOM). IEEE, 2016: 193-198.

[50] Yi C, Cai J, Su Z. A multi-user mobile computation offloading and transmission scheduling mechanism for delay-sensitive applications[J]. IEEE Transactions on Mobile Computing, 2019, 19(1): 29-43.

[51] Zhang T. Data offloading in mobile edge computing: A coalition and pricing based approach[J]. IEEE Access, 2017, 6: 2760-2767.

[52] Liu M, Liu Y. Price-based distributed offloading for mobile-edge computing with computation capacity constraints[J]. IEEE Wireless Communications Letters, 2017, 7(3): 420-423.

[53] Chen X, Li W, Lu S, et al. Efficient resource allocation for on-demand mobile-edge cloud computing[J]. IEEE Transactions on Vehicular Technology, 2018, 67(9): 8769-8780.

[54] Wang Q, Guo S, Wang Y, et al. Incentive Mechanism for Edge Cloud Profit Maximization in Mobile Edge Computing[C]//ICC 2019-2019 IEEE International Conference on Communications (ICC). IEEE, 2019: 1-6.

[55] Sun W, Liu J, Yue Y, et al. Double auction-based resource allocation for mobile edge computing in industrial internet of things[J]. IEEE Transactions on Industrial Informatics, 2018, 14(10): 4692-4701.