Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing

Chen X, Jiao L, Li W, et al. Efficient multi-user computation offloading for mobile-edge cloud computing[J]. IEEE/ACM Transactions on Networking, 2016, 24(5): 2795-2808.

移动边缘云计算是一种新的范式，可以在靠近移动用户的无线电接入网络的边缘提供云计算能力。在本文中，我们首先研究了多通道无线干扰环境下移动边缘云计算的多用户计算分流问题。我们表明计算集中式最优解式 NP-hard 的，因此采用博弈论方法，以分布式方式实现有效的计算分流。我们把移动设备用户的计算分流问题看做多用户计算分流的博弈，提出了分布式计算分流决策。我们分析了博弈的结构性质，证明了该博弈具有纳什均衡，且具有有限的改进性质。然后，我们设计一个分布式计算分流算法，可以实现纳什均衡，导出收敛时间的上限，并根据两个重要的性能指标量化其在集中式最优解上的效率比。我们进一步将研究扩展到多通道无线竞争环境中多用户计算分流的场景。数值结果证实，随着用户规模的增加，提出的算法可以实现优异的计算分流性能和规模。

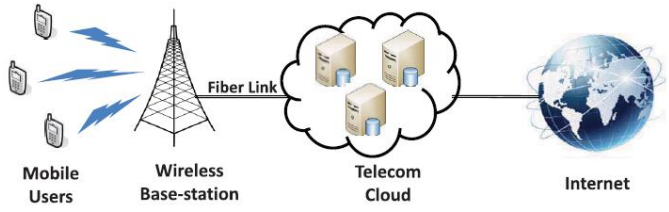


Fig. 1. An illustration of mobile-edge cloud computing.

移动边缘计算：通过无线接入将计算分流到物理位置上相对近的计算服务器或计算机集群上。本文主要解决两个问题：1 移动用户如何选择本地计算和云计算；2 如果用户选择了云计算，如何选择合适的信道实现高效的无线接入。

集合 $N=\{1,2,...,n\}$ 表示 n 个移动设备用户，无线基站 s，集合 $M=\{1,2,...,m\}$ 表示 m 个无线信道，决策方案 $a=\{a_1,a_2,...,a_n\}$，$a_i \in \{0\}\quad M$（用户 i 选择本地 0 或者是 m 信道），用户 n 的任务用 (b_n,d_n) 即（输入数据的大小，完成任务所需的 CPU 周期数）。用户 n 的本地计算的总开销 $K_n^m=\lambda_n^t t_n^m+\lambda_n^e e_n^m=\lambda_n^t\frac{d_n}{f_n^m}+\lambda_n^e\gamma_n d_n$。（$f_n^m$ 是用户 n 的计算能力，即每秒的 CPU 周期数，γ 是每 CPU 周期消耗能量的系数。）用户 n 的云计算总开销 $K_n^c(a)=\lambda_n^t(t_{n,off}^c(a)+t_{n,exe}^c)+\lambda_n^e e_n^c(a)=\lambda_n^t(\frac{b_n}{r_n(a)}+\frac{d_n}{f_n^c})+\lambda_n^e(\frac{q_n b_n}{r_n(a)}+L_n)$。（$\lambda_n^t$、$\lambda_n^e$ 是用户决策的权重，比较在意延迟就置前者为 1，后者为 0；$r_n(a)$ 是做出决策 a 的情况下用户 n 的上传速率，f_n^c 是云分配给用户 n 的计算能力，q_n 是用户 n 的传输功率，L_n 是用户保持信道占有所需的能量）。

考虑整个系统的两个性能指标：使用云分流的用户个数和整个系统的能量消耗。不论是最大化用户个数还是最小化能量消耗，这两个问题的集中式最优解问题都是 NP-难的，所以考虑分布式最优解问题。

定义 $a_{-i}=(a_1,...,a_{i-1},a_{i+1},...,a_n)$ 是除了用户 i 之外的其他所有用户的决策，那么考虑的

最优化问题为：给定其他用户的决策a_{-i}的情况下，用户 i 如何做出决策使得自身的能耗最小，即$\min_{a_i\in\{0,1,\ldots,m\}} Z_i(a_i,a_{-i})=\begin{cases}K_i^m, & if\ a_i=0,\\ K_i^c(a), & if\ a_i=0,\end{cases}$，然后根据博弈论，可以证明该问题存在纳什均衡，并且在两项系统指标上，分布式问题的最优解都可以很好的近似为集中式问题的最优解。

引发思考的问题：

（1）实际中，任务上传的时间不一样，不一定都是同时进行分流

（2）考虑任务的可分性，例如设备本地可以处理任务中的一些简单部分，困难部分例如数据计算超出设备 cpu 能力或者请求访问其他设备数据等，这些再分流到云端进行处理。

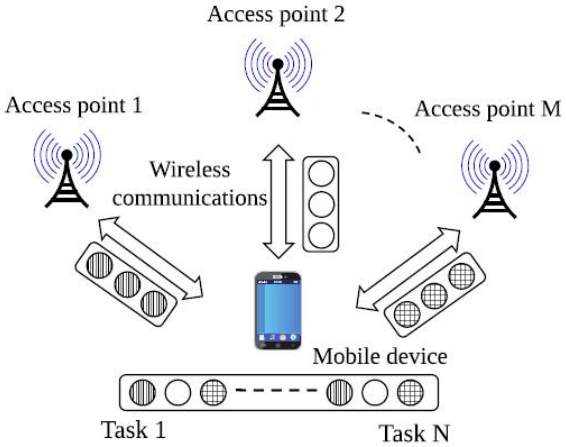
（3）实际中传输延迟大于计算延迟，未考虑距离因素

（4）不同的计算任务可能有不同的分流方式，有的任务可以分流给一个云，有的任务需要分流给多个云

Offloading in Mobile Edge Computing: Task Allocation and Computational Frequency Scaling

Dinh T Q, Tang J, La Q D, et al. Offloading in Mobile Edge Computing: Task Allocation and Computational Frequency Scaling[J]. IEEE Transactions on Communications, 2017.

在本文中，我们提出从单个移动设备分流到多个边缘设备的优化框架。我们的目标是通过优化任务分配决策和移动设备的 CPU 频率来最大限度地减少总任务的执行延迟和移动设备的能耗。本文考虑了移动设备的两种情况，即固定 CPU 频率和弹性 CPU 频率。由于这些问题是 NP 难的，针对固定 CPU 频率的情况，我们提出了一种基于线性松弛的方法和基于半固定松弛（SDR）的方法，以及针对弹性 CPU 频率的情况提出基于穷尽搜索的方法和基于 SDR 的方法。我们的仿真结果表明，基于 SDR 的算法实现了近乎最优的性能。在考虑多边缘设备和弹性 CPU 频率时，可以通过提出的方案在能量消耗和任务的执行延迟方面实现性能的改善。最后，我们证明了移动设备灵活的 CPU 范围可以对任务分配产生影响。



Offloading framework with multiple APs.

NOTATIONS USED THROUGHOUT THE PAPER

Notation	Definition
i	index of a task
k	index of a AP's CPU or MD's CPU
α_i	size of the input data of task i
β_i	size of the out data of task i
w_i	required number of CPU cycles to process task i
λ_t	scalar weight of tasks' execution latency
λ_e	scalar weight of the MD's energy consumption
$C_k^{\mathrm{UL}}, C_k^{\mathrm{DL}}$	uplink and downlink data rate between the MD and AP k
D_{ik}	execution latency when task i is offloaded to CPU k
r_k	service rate of the AP's CPU k
r_0	CPU frequency of the MD
ρ	model dependent constant for the CPU frequency of the computational processing energy consumption
P^{Comp}	computational power of the MD
$P^{\mathrm{Tx}}, P^{\mathrm{Rx}}$	transmitting and receiving power of the MD
E^{Comp}	computational energy consumption of the MD
E^{TR}	transmission energy consumption of the MD
x_{ik}	decision variable assigning task i to CPU k
$\mathbf{X}$	task allocation matrix
$t(\mathbf{X}, r_0)$	tasks' execution latency
$e(\mathbf{X}, r_0)$	total energy consumption of the MD
$r_{\min}, r_{\max}$	minimum and maximum CPU frequency of the MD

AP 指的是 Fog-Radio Access Network (F-RAN) node。不仅具有通信能力，还提供应用服务。占用正交信道。

问题形式化：

一个 MD（移动设备）有 n 个独立的任务，任务集合 N={1,2，……，n}，N 个独立任务被分割乘 M+1 个不相交集合，每个集合看作一个批次，或是一个大任务；m 个 AP 的 CPU 集合 M={1,2，……，m}；任务 i 用元祖$\{\alpha_i, \beta_i, \omega_i\}$表示；

任务分配矩阵 $\mathbf{X} = \{x_{ik}\} \in \{0,1\}^{N\times(M+1)}$，$\mathbf{x} = [\mathbf{x_0^T}, \mathbf{x_1^T}, \quad, \mathbf{x_k^T}, \quad \mathbf{x_m^T}]^T$，其中

$\mathbf{x}_k = [x_{1k}, x_{2k}, \quad x_{Nk}]^T$

假定 AP 收到全部任务后开始处理任务，不考虑排队任务重叠的情况，那么 CPU k 的执行延迟为 $T_k = T_k^{UL} + T_k^{Comp} + T_k^{DL} = \sum_{i\in N} x_{ik}(\frac{\alpha_i}{C_k^{UL}} + \frac{\omega_i}{C_k^{Comp}} + \frac{\beta_i}{C_k^{DL}}) = \sum_{i\in N} x_{ik} D_{ik}$。

APs 并行处理任务，MD 的执行延迟为 $t(\mathbf{X}, r_0) = \max_{\mathrm{k}\in\mathrm{M}} T_k$。

总能耗=MD 的计算能耗+无线传输能耗为

$e(X,r_0)=E^{Comp}+E^{TR}=\rho r_0^{\zeta}\sum_{i\in N}x_{i0}D_{i0}+P^{Tx}\sum_{k\in M\setminus\{0\}}\sum_{i\in N}x_{ik}d_{ik}^{UL}+P^{Rx}\sum_{k\in M\setminus\{0\}}\sum_{i\in N}x_{ik}d_{ik}^{DL}$ 。

目标函数$\psi(X,r_0)=\lambda_t t(X,r_0)+\lambda_e e(X,r_0)$。

优化问题为：$\min_X \psi(X,r_0)$

$$\text{s.t}\ \sum_{k\in M}x_{ik}=1,\forall i\in N,$$
$$r_0\in[r_{\min},r_{\max}],$$
$$x_{ik}\in\{0,1\},$$

一个任务只分给 1 个 AP

固定 CPU 频率情况下优化任务分流：r_0 D_{i0} $P^{Comp}=\rho r_0^{\zeta}$ 都为常数，引入变量 $t\geq\max_{k\in M}(\sum_{i\in N}x_{ik}D_{ik})$。优化问题变为一个混合整数线性规划问题，可用分支限界法，但复杂度大，故本文提出线性规划松弛的方法，关键是将$x_{ik}\in\{0,1\}$放大到$x_{ik}\in[0,1]$，然后就可以用内点法求解。

弹性 CPU 频率的情况下，首先用穷举搜索的方法找到最优解，作为算法比较的标准，又提出基于半固定松弛（SDR）的方法，求得近似解。

本文的问题：

（1）未考虑不同的任务之间存在处理的时间限制问题；

（2）用 CPU 周期数来定义任务的计算量是否标准

（3）传输能量损耗的计算应该与距离有关，且 CPU 计算的能耗实际原大于传输能耗

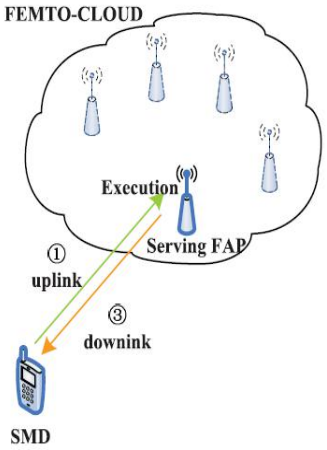
（4）任务之间存在排队问题，应该将排队的延迟也计算到任务延迟中

（5）未考虑任务分配的实际问题，可以将相似的任务分配给同一个 AP，AP 的计算任务均衡问题，比如每个 AP 最多处理总任务数/总 AP 数。

Mobile-Edge Computing: Partial Computation Offloading Using Dynamic Voltage Scaling

Wang Y, Sheng M, Wang X, et al. Mobile-edge computing: Partial computation offloading using dynamic voltage scaling[J]. IEEE Transactions on Communications, 2016, 64(10): 4268-4282.

将动态电压缩放技术结合到计算分流中为移动边缘计算提供了更多的灵活性。在本文中，我们通过共同优化智能移动设备（SMD）的计算速度、SMD 的传输功率、以及以 SMD 能量消耗最小化（ECM）和应用程序执行延迟最小化（LM）为系统设计目标的分流率来研究部分计算分流。考虑到 SMD 是由单一的云服务器提供的，我们将 ECM 问题和 LM 问题都定义为非凸问题。为了解决 ECM 问题，我们用可变替代技术将其重新形成为凸的，并获得其最优解。 为了解决非凸和非光滑 LM 问题，我们提出了采用单变量搜索技术的局部最优算法。此外，我们将场景扩展到多个云服务器系统，在此系统中，SMD 可以将其计算分流到一组云服务器上。在这种情况下，我们以封闭形式为 ECM 和 LM 问题获得云服务器之间的最优计算分布。最后，大量的仿真表明，我们提出的算法可以显著降低能耗，缩短现有分流方案的延迟时间。



W_U/W_D	uplink/downlink channel bandwidth
d	distance from the SMD to its serving FAP
h_1/h_2	uplink/downlink channel fading coefficients
N_0	white Guassian noise power
P_t	transmit power of SMD
P_0	static power consumption of SMD
k_t	efficient factor of power amplifier of SMD
P_r	receive power consumption of SMD
P_F	transmit power of the serving FAP
$P_{t_{\max}}$	maximum transmit power of SMD
f_l	computational speed of SMD
f_c	computational speed of cloud
$f_{l_{\max}}$	maximum computational speed of SMD
k	a coefficient depending on chip architecture (using for modeling computation energy consumption)
I	amount of computation input data bits
$L_{\max}$	application-dependent latency requirement
$E_{\max}$	maximum energy supplied by SMD
C	number of cycles needed for the application
β_1	a coefficient accounting for the overhead in uplink transmission

β_2	a coefficient jointly accounting for the overhead in downlink transmission and the ratio of output to input bits offloaded to the cloud
λ	ratio of locally executed amount of bits to the total input data bits
R_U/R_D	uplink/downlink rate
t_U/t_D	uplink/downlink transmit delay
τ_c	execution time in the cloud
t_l/E_l	time/energy consumption for local execution part
t_c/E_c	time/energy consumption for offloading part
L/E	total time/energy consumption of SMD
N	number of FAPs in femto-cloud
L_c	cloud latency in multi-FAP scenario
t_{cm}	time for offloading part in multi-FAP scenario
f_n	computational speed of FAP n
w_n	allocated computation bits to FAP n
$\delta_{Tx,bh}(n)$	one way communication latency from the associated FAP to FAP n
$\delta'_{Tx,bh}(n)$	one way communication latency from FAP n to the associated FAP

新兴的移动分流结构：femto-cloud，由一组云增强的 FAPs（femto access points）协作形成一个 femto cloud，并使得 SMD 能够近距离访问云计算服务。

主要研究面向数据分区型的应用程序任务，用 $(I, L_{\max})$ 表示，I 表示需要计算的输入数据的 bits，$L_{\max}$ 表示依赖于程序的延迟要求。任务所需要的 CPU 周期数 $C=\alpha I$，α 取决于任务的性质，比如计算复杂度。

能耗于延迟模型：

SMD 的执行时间 $t_l=\dfrac{\alpha\lambda I}{f_l}$，计算能耗 $E_l=\alpha\lambda I k f_l^2$，传输能耗 $E_c=(P_0+k_tP_t)t_U+P_rt_D$，分流时间 $t_c=t_U+\tau_c+t_D$。考虑并行执行，所以总延迟为 $L(f_l,P_t,\lambda)=\max\{t_l,t_c\}$，总能耗为 $E(f_l,P_t,\lambda)=E_l+E_c=\alpha Ik\lambda f_l^2+(P_0+k_tP_t)\dfrac{\beta_1(1-\lambda)I}{W_u\log_2(1+P_ta)}+P_r\dfrac{\beta_2(1-\lambda)I}{R_D}$。

优化问题：$\min\limits_{f_l,P_t,\lambda} E(f_l,P_t,\lambda)$

$$
\begin{aligned}
s.t.\quad & C1: L(f_l,P_t,\lambda)\le L_{\max},\\
& C2: 0\le\lambda\le 1,\\
& C3: 0\le P_t\le P_{t_{\max}},\\
& C4: 0\le f_l\le f_{l_{\max}}.
\end{aligned}
$$

又 f_l 的最优解可以表示为 $f_l^*(\lambda)=\dfrac{\alpha\lambda I}{L_{\max}}$，以及设 $f(P_t)=\dfrac{P_0+k_tP_t}{W_U\log_2(1+P_ta)}$，可以证得

是个单峰函数，P_t 的取值范围与 λ 有关，但是该函数的极点与 λ 无关，于是 P_t 的最优解可以表示为 $P_t^*(\lambda)$。

原优化问题可化为：$\min E(\lambda)$

$$s.t.\ C8:\lambda_{\min} \le \lambda \le \lambda_{\max}$$。

这样就化成一个一维的问题，可以用二分法求得 λ 的最优解，从而求出 P_t、f_l 的最优解。

扩展到多个 FAPs，即 SMD 将数据发送给最合适的 FAP，然后这个 FAP 将任务再分发给 femto cloud 中的其他 FAPs 进行协作计算。将问题分成两个子问题：给定一个假设为最合适的 FAPs，找到最优的计算分布；在所有的 FAPs 中寻找最合适的 FAPs。找最优的计算分布的优化问题其实最小化云内延迟，从而可以求得最优的分布；由于 femto cloud 内的 FAPs 数量一般不会很多，可以用穷举法，找到最优的 FAP。

本文的另一个优化问题，即将能量损耗作为约束来最小化延迟，由于出现 4 个变量，观察发现固定其中两个变量，问题变为凸规划，固定其他两个变量，问题变成线性规划。采用单变量搜索技术的局部最优算法获得一组单调不增序列来逼近最优解。

本文存在的问题：

（1）处理的任务只有一个

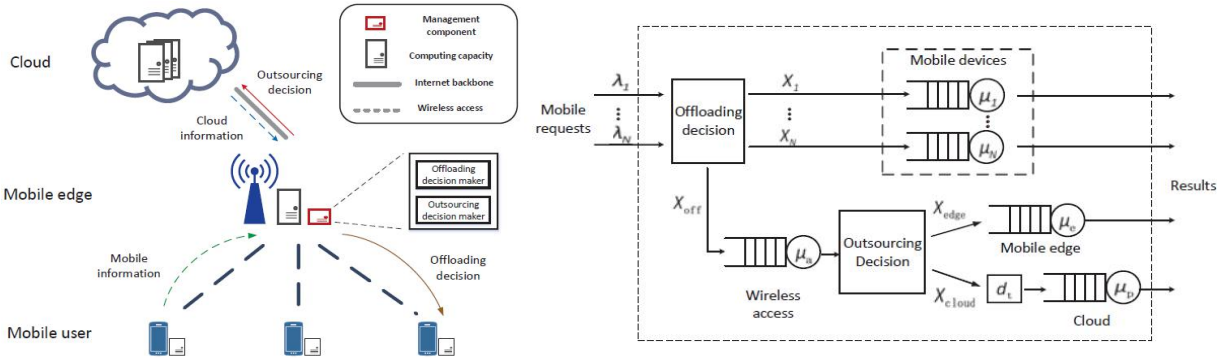
（2）任务可能跟 CPU 周期数不是线性关系。即 $C = \alpha I$ 的定义存疑。

（3）只考虑了单个移动设备。

Cost-Efficient Workload Scheduling in Cloud Assisted Mobile Edge Computing

Ma X, Zhang S, Li W, et al. Cost-efficient workload scheduling in Cloud Assisted Mobile Edge Computing[C]//Quality of Service (IWQoS), 2017 IEEE/ACM 25th International Symposium on. IEEE, 2017: 1-10.

移动边缘计算是一种具有低延迟的优势的有前途的计算范例。然而，与传统的移动云计算相比，移动边缘计算在计算能力方面受到限制，特别是在人口密集的情况下。在本文中，我们提出了一个云辅助移动边缘计算框架，利用云资源来提高系统的计算能力。为了平衡系统延迟与成本，进一步设计了移动工作负载调度和云外包。具体地说，就是通过将云计算辅助移动边缘计算系统建模为一个排队网络，对系统延迟进行了分析。此外，还提出了一个优化问题，以最小化系统的延迟和成本。这个问题被证明是凸的，可以通过使用 karush - kuhn - tucker(KKT)条件来解决。通过利用约束的线性特性，提出了具有线性复杂度的算法，而不是直接求解具有指数复杂度的 KKT 条件。对所提出的算法进行了大量的仿真。与公平比算法和贪心算法相比，该算法在相同的外包成本下，可以分别降低系统延迟到 33%和 46%。仿真结果表明，该算法能够有效地应对异构移动用户的挑战，平衡计算延迟与传输开销之间的权衡。



云辅助移动边缘计算框架（CAME），本文通过将 CAME 系统建模为一个队列网络来分析延迟，移动计算请求到达系统时，移动边缘的管理组件收集相关信息（移动设备的任务需求以及计算能力、云的租用价格和计算能力），分流决策模块根据每个移动设备的情况来做分流决策，当大量工作分流到移动边缘时，通过外包决策模块，决定是否需要租用云来增强计算能力。

系统模型：移动用户集合 $N=\{1,2,...,n\}$，用户 i 的计算请求的到达服从参数为 λ_i 的泊松分布。μ_i 表示用户 i 的服务率，x_i 为执行用户 i 的计算请求的平均速率，$0 \le x_i \le \lambda_i$（1），所以用户 i 的平均计算延迟 $D_i = \dfrac{1}{\mu_i - x_i}$，$x_i \le \mu_i$（2）。

从移动用户分流的计算请求到达率 $x_{off} = \sum_i^n \lambda_i - \sum_i^n x_i$（3），当分流计算请求时就会有额外的通信量产生，假设一单位的计算请求有 c 单位的传输请求，那么传输请求的到达率 $\lambda_a = cx_{off}$，μ_a 表示无线网的服务率，平均的传输延迟 $D_a = \dfrac{1}{\mu_a - cx_{off}}$，$0 \le x_{off} \le \dfrac{\mu_a}{c}$（4）。

在外包决策模块处理完之后，x_{edge} 为计算请求到达移动边缘的到达率，μ_e 表示移动边缘的服务率，移动边缘的计算延迟 $D_{edge} = \dfrac{1}{\mu_e - x_{edge}}$，$0 \le x_{edge} \le \mu_e$（5）。外包给云的计算请求的到达率 $x_{cloud} = x_{off} - x_{edge}$（6）。

在云中，计算资源可以看做是一个封装的 instance，用 k 表示租用的云的 instance 数量。$0 \le k \le M$（7），μ_{ins} 表示单个 instance 的服务率，那么云的服务率 $\mu_p = k\mu_{ins}$，云的平

均计算延迟 $D_{cloud}=\frac{1}{k\mu_{ins}-x_{cloud}}$，$0\le x_{cloud}\le k\mu_{ins}$（8）。

系统延迟认为是系统各部分延迟的加权总和，权重有计算请求的比值表示。系统延迟

$$D=\sum_{i=1}^{n}\frac{x_i}{\lambda}D_i+\frac{x_{off}}{\lambda}D_a+\frac{x_{edge}}{\lambda}D_{edge}+\frac{x_{cloud}}{\lambda}(D_{cloud}+d_t)$$
$$=\frac{1}{\lambda}(\sum_{i=1}^{n}\frac{x_i}{\mu_i-x_i}+\frac{x_{off}}{\mu_a-x_{off}}+\frac{x_{edge}}{\mu_e-x_{edge}}+\frac{x_{cloud}}{k\mu_{ins}-x_{cloud}}+x_{cloud}d_t)$$

，其中 $\lambda=\sum_{i=1}^{n}\lambda_i$，$d_t$ 表示移动边缘与云之间的传输。

系统成本分为两类，一是内部成本，即移动边缘基础设施的成本，假设移动边缘提供恒定的计算能力，用常数 c_m 表示，二是外包成本，用 c_p 表示单个 instance 按小时收费的价格，所以系统总成本 $C=kc_p+c_m$。

问题形式化：决策变量 $\mathbf{a}=(x_1,x_2,...,x_n,x_{off},x_{edge},x_{cloud},k)$ 优化问题为 $\min f(\mathbf{a})=C(\mathbf{a})+VD(\mathbf{a})$，s.t. (1)—（8）。

对于该问题可以分两步解决，首先对与每个给定的 $k\in\{0,1,2,...,\mathrm{M}\}$，找出优化问题的 $\min f(\mathbf{x})=C(\mathbf{x})+VD(\mathbf{x})$，s.t. (1)—(6)、(8)的最优解，其中 $\mathbf{x}=(x_1,x_2,...,x_n,x_{off},x_{edge},x_{cloud})$；然后选择最优的 $k^*=\arg\min_{k\in\{0,1,2,...M\}}\{f(\mathbf{x}^*(k)),k)\}$。

这个问题可以证明是凸的，可以通过使用 karush - kuhn - tucker(KKT)条件来解决，但是计算复杂度是指数级的，所以提出了一个线性复杂度的算法，即忽略不等式约束条件，最后证明由此得到的最优解也仍然是原问题的最优解。

本文的问题：若站在移动用户的角度思考，是否付费取决于移动边缘的决策，这不太合理，若移动边缘为恶意节点，或者移动边缘提供商于云服务提供商互相勾结，可能增大用户的开销成本。一个合理的想法是站在移动边缘提供商的角度，考虑如何在保证服务质量的情况下最小外租用云资源的费用成本。

Fair Caching Algorithms for Peer Data Sharing in Pervasive Edge Computing Environments

Huang Y, Song X, Ye F, et al. Fair Caching Algorithms for Peer Data Sharing in Pervasive Edge Computing Environments[C]//Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on. IEEE, 2017: 605-614.

具有传感，存储和通信资源的边缘设备（例如，智能手机，平板电脑，联网车辆，IoT节点）越来越多地渗透到我们的环境中。当附近的边缘设备共享数据时，可以创建许多新颖的应用程序。缓存可以大大提高数据可用性,鲁棒性和延迟。本文研究了边缘环境下缓存公平性的独特问题。由于对对等设备的不同所有权，缓存负载平衡非常重要。我们考虑了公平性指标，并提出了一个整数线性规划问题，即多个连接设施位置的总和(ConFL)问题。我们提出了一种利用现有的连接设施位置近似算法的近似算法，并证明它保留了 6.55 的近似比。我们进一步开发了一种分布式算法，在该算法中，设备可以交换数据，并将受欢迎的候选对象识别为缓存节点。大量的评估表明，与现有的无线网络缓存算法相比，我们的算法显著提高了数据缓存的公平性，同时保持了与现有算法相似的争用延迟。

Computation Offloading and Resource Allocationin Wireless Cellular Networks With Mobile Edge Computing

Wang C, Liang C, Yu F R, et al. Computation Offloading and Resource Allocation in Wireless Cellular Networks with Mobile Edge Computing[J]. IEEE Transactions on Wireless Communications, 2017.

移动边缘计算已经成为提高移动设备计算能力的一项很有前途的技术。与此同时，网络缓存已经成为处理成倍增长的互联网流量的一种自然趋势。这两种网络模式中的重要问题分别是计算分流和内容缓存策略。为了在移动边缘计算的无线蜂窝网络中共同解决这些问题，我们考虑到网络的总收益，将计算分流决策、资源分配和内容缓存策略作为优化问题。此外，我们将原始问题转化为凸问题，然后分解，以一种分布式高效的方法求解。最后，随着分布式凸优化的最新进展，我们提出了一种基于交替方向乘子（ADMM）算法的方法来解决优化问题。通过对不同系统参数的仿真，验证了该方案的有效性。

Joint Subcarrier and CPU Time Allocation for Mobile Edge Computing

Yu Y, Zhang J, Letaief K B. Joint Subcarrier and CPU Time Allocation for Mobile Edge Computing[C]//GLOBECOM. 2016: 1-6.

在移动边缘计算系统中，移动设备可以将计算密集型任务分流到附近的 cloudlet，从而节省能源并延长电池寿命。与完全成熟的云不同，cloudlet 是部署在无线接入点的小型数据中心，因此受到无线电和计算资源的高度限制。我们在本文中表明，单独优化计算或无线电资源的分配（如大多数现有工作所做的）是非常不理想的。因为计算资源的拥塞会导致无线电资源的浪费，而反之亦然。为了解决这个问题，我们提出了一种协调分配无线电和计算资源的联合调度算法。具体来说，我们考虑了一个具有多个移动设备的正交频分复用接入（OFDMA）系统中的 cloudlet，在这个 cloudlet 上研究用于任务分流的子载波分配和用于任务执行的 CPU 时间分配。仿真结果表明，提出的算法明显优于每个资源优化，可以实现更多的分流请求，同时实现显著的节能效果。

A Load Balancing Scheme for Sensing and Analytics on a Mobile Edge Computing Network

Tham C K, Chattopadhyay R. A load balancing scheme for sensing and analytics on a mobile edge computing network[C]//A World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2017 IEEE 18th International Symposium on. IEEE, 2017: 1-9.

实时感知，数据分析和其他应用的处理时间是一个重要的性能指标。当应用程序在分布式系统上执行时，处理节点之间的负载平衡方案会显著影响应用程序的总处理时间。我们考虑在网络边缘的分布式计算的负载平衡方案。在考虑的边缘模型中，具有处理和感测能力的一组移动或静态节点通过无线自组织网络相互连接。边缘节点之间的负载平衡是一个 min-max 优化问题，目标是最小化应用程序的整体处理时间，同时仍然满足无线信道容量和

链路争用约束。我们用聚合效用法将 min-max 问题转换成一个凸优化问题。然后用拉格朗日对偶分解对得到的约束凸优化问题进行了松弛处理，并以梯度下降法求解。这种形式可以在边缘节点之间以完全分布式的方式实现，这与边缘网络的分散性相一致。然而，这个方案的收敛速度可能很慢。进一步提出了一种快速收敛的启发式算法。仿真结果表明，该方法在大多数情况下都具有近似最优性能。

Optimal trade-off between accuracy and network cost of distributed learning in Mobile EdgeComputing: An analytical approach

Valerio L, Passarella A, Conti M. Optimal trade-off between accuracy and network cost of distributed learning in Mobile Edge Computing: An analytical approach[C]//A World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2017 IEEE 18th International Symposium on. IEEE, 2017: 1-9.

最广泛采用的从互联网边缘生成的原始数据（比如由 IoT 或个人移动设备生成）获取知识的方法是通过全球云平台从设备中收集数据并进行分析。然而，随着越来越多的设备在物理环境中传播，这种方法产生了几个问题。数据引力概念是雾和移动边缘计算的基础之一，它指出了数据分析的分散性，即数据分析在数据生成时更接近于数据生成的位置，这既包括可伸缩性，也包括隐私方面的原因。因此,设备产生的数据可以根据以下的一种方法进行处理:(i)直接在收集数据的设备上，(ii)在云中,(iii)通过雾/移动边缘计算技术,即在网络的中间节点，在收集数据子集后运行分布式分析。显然，(i)和(ii)是(iii)的两个极端案例，值得注意的是，在网络不同的收集点上执行的相同的分析任务，在网络生成的流量方面有不同的成本。准确地说，这些成本指的是将数据移动到选定的集合点(例如边缘或云)和由分布式分析过程引起的数据的流量。直到现在，决定是否使用中间集合点，以及它们应该在哪个方面获得目标的准确性和最小化网络流量，这是一个悬而未决的问题。在本文中，我们提出了一个能够解决这个问题的分析框架。准确地说，我们考虑学习任务，并定义一个模型，将学习任务的准确性与特定的集合点和相应的网络流量联系起来。该模型可用于识别学习问题的规格(如二元分类、回归等)，以及其目标精度，即收集数据的最佳水平，以最大限度地降低网络总成本。通过仿真验证我们的模型，以表明在仿真中，我们的模型所显示的中间集合的水平，可以达到目标精度的最小成本。

Radio Environment Aware Computation Offloading with Multiple Mobile Edge Computing Servers

Sato K, Fujii T. Radio Environment Aware Computation Offloading with Multiple Mobile Edge Computing Servers[C]//Wireless Communications and Networking Conference Workshops (WCNCW), 2017 IEEE. IEEE, 2017: 1-5.

在本文中，我们讨论了移动边缘计算（MEC）系统中的计算分流。我们特别关注移动节点向具有计算能力的周边接入点（AP）发送计算请求的情况。与有线分布式计算系统不同，MEC 系统由于无线信道波动（如多径衰落和阴影效应）而受到不稳定的连接，计算结果有可能由于无线连接丢失而消失。为了最大限度地提高无线环境中的分流效果，我们提出了无线环境感知任务分配方法，预测移动节点与 AP 之间的无线连接。特别地，我们讨论了两种方法：基于无线电环境映射（REM）的连通性预测和基于距离的连接。数值计算结果表明，与忽略无线连接的传统方法相比，所提出的方法可以有效地分流计算任务。

Joint Task Offloading Scheduling and Transmit Power Allocation for Mobile-Edge Computing Systems

Mao Y, Zhang J, Letaief K B. Joint Task Offloading Scheduling and Transmit Power Allocation for Mobile-Edge Computing Systems[C]//Wireless Communications and Networking Conference (WCNC), 2017 IEEE. IEEE, 2017: 1-6.

移动边缘计算(MEC)作为一种突出的技术，通过将移动设备的计算密集型任务从移动设备迁移到附近的 MEC 服务器，从而提供高计算需求的移动服务。为了减少执行时延和设备能耗，本文对具有多个独立任务的 MEC 系统的任务分流调度和传输功率分配进行了优化。具体地说，在传输功率分配的基础上，借助于流水作业调度理论，获得了最优任务分流调度，

即确定分流顺序。此外，利用凸优化技术确定了给定任务分流调度决策的最优传输功率分配。仿真结果表明，当 MEC 系统中可用的无线电和计算资源相对均衡时，任务分流调度更为关键。此外，本文还表明，该算法实现了接近最优的执行延迟，并实现了大量的设备节能。

Adaptive Computation Scaling and Task Offloading in Mobile Edge Computing

Dinh T Q, Tang J, La Q D, et al. Adaptive Computation Scaling and Task Offloading in Mobile Edge Computing[C]//Wireless Communications and Networking Conference (WCNC), 2017 IEEE. IEEE, 2017: 1-6.

移动设备(MDs)的能量消耗和应用程序的执行延时可以通过将应用程序任务迁移到附近的边缘设备来提高。在本文中，我们提出了一种优化框架，用于研究 MD 能够将任务分流到多个访问点(APs)并将其中央处理器频率扩展的情况。首先从基于穷举的搜索方法推导出最优解;然后，提出了一种基于半定松弛（SDR）的方法以有效地解决问题。仿真结果表明，基于 SDR 算法的算法能够实现最优性能。我们还表明，我们提出的方案可以通过利用多个 AP 和灵活的 CPU 频率来减少 MD 的能耗和任务的执行延迟。

Multi-objective Optimization for Computation Offloading in Mobile-edge Computing

Liu L, Chang Z, Guo X, et al. Multi-objective optimization for computation offloading in mobile-edge computing[C]//Computers and Communications (ISCC), 2017 IEEE Symposium on. IEEE, 2017: 832-837.

移动边缘云计算是一种新的云平台，它可以随时随地提供普及和敏捷的计算扩展服务，从而为移动设备提供更多的服务。尽管将计算分流到云上可以降低移动设备的能耗，但也可能导致更大的执行延迟。通常移动设备必须支付他们使用的云资源。本文利用排队论的方法，对移动边缘云系统中分流过程的能耗、执行延迟和价格成本进行了深入的研究。具体来说，在对能耗和延迟性能进行建模时，无线传输和计算能力都将被明确和共同考虑。在理论分析的基础上，通过寻找每个移动设备的最优分流概率和最优传输功率的联合目标来制定多目标优化问题，以最小化能耗，执行延迟和价格成本。应用标量化方法和内点法来解决制定的问题。 通过大量的模拟，可以证明所提出的方案的有效性。

Proximity-aware IaaS for Edge Computing Environment

Yamanaka H, Kawai E, Teranishi Y, et al. Proximity-Aware IaaS for Edge Computing Environment[C]//Computer Communication and Networks (ICCCN), 2017 26th International Conference on. IEEE, 2017: 1-10.

边缘计算允许应用程序服务能够利用低延迟响应，例如增强现实。边缘计算的基础设施即服务（EC-IaaS）允许多个应用服务提供商（ASP）将虚拟化资源（即虚拟机和虚拟网络）用于公共边缘计算基础设施上的各种应用程序服务。对于应用服务提供商，虚拟机到终端用户设备的接近度（即主机边缘服务器的位置）是保证低延迟响应时间的重要因素。现有方法允许应用服务提供商选择边缘服务器位置，以使虚拟机满足接近度要求。然而，从边缘计算的基础设施提供商发送到应用服务提供商的信息量变大，因为它必须包括所有边缘服务器位置和所有无线基站之间的延迟。我们提出一种减少信息量的方法，同时确保应用提供商获得满足接近度要求的虚拟机。一个边缘计算基础设施提供商表明无线基站组在适当的距离内拥有虚拟机，有助于应用提供商估计虚拟机的必要数量。与应用提供商相比，边缘计算的基础设施提供商则决定哪些边缘服务器应该包含虚拟机。我们可以确定所提出的方法可以减少传输信息。边缘计算的基础设施提供商也有一个好处，可以将详细的延迟信息保留在应用服务提供商中，而与现有方法相比，功耗是不变的。

Online Learning for Offloading and Autoscaling in Renewable-Powered MobileEdge Computing

Xu J, Ren S. Online learning for offloading and autoscaling in renewable-powered mobile edge computing[C]//Global Communications Conference (GLOBECOM), 2016 IEEE. IEEE, 2016: 1-6.

最近出现的移动边缘计算(又名雾计算)可以在移动网络的边缘对延迟敏感的应用程序

进行现场处理。然而，提供支持移动边缘计算的电力电源成本高昂，甚至在某些崎岖的或欠发达地区是不可行的，因此，在越来越多的情况下，要求将现场可再生能源作为一个主要的甚至是唯一的电力供应。尽管如此，可再生能源的高间歇性和不可预测性使得在可再生的移动边缘计算系统中向用户提供高质量的服务变得非常具有挑战性。本文中，我们解决了将可再生能源纳入移动边缘计算的挑战，并提出了一种高效的强化学习型资源管理算法，它可以动态地学习动态工作负载分流(集中云)和边缘服务器供应的最优策略，以最小化长期系统成本(包括服务延迟和运营成本)。我们的在线学习算法使用了(脱机)值迭代和(在线)强化学习的分解，从而实现了与标准强化学习算法(如 q - learning)相比，学习速度和运行时性能的显著提高。

Cooperative Load Balancing Scheme for Edge Computing Resources

Beraldi R, Mtibaa A, Alnuweiri H. Cooperative load balancing scheme for edge computing resources[C]//Fog and Mobile Edge Computing (FMEC), 2017 Second International Conference on. IEEE, 2017: 94-100.

边缘计算，是一种正在兴起的利用网络边缘的计算能力的解决方案。边缘计算的一个关键挑战是提供低服务阻塞和低延迟的计算服务，否则就会导致边缘计算系统的低效部署。与云计算不同，边缘计算中的计算资源是有限的。通过实现数据中心之间的合作可以处理这个限制。

在本文中，我们提出 CooLoad，一个安装在网络边缘的两个数据中心之间的协作方案，当其中一个数据中心临时超负荷的时候，它可以与另一个数据中心交换通用的计算请求。如果服务请求到达一个数据中心，其请求缓冲区已满，则请求将被转发到另一个协作数据中心，并由该数据中心服务，消耗其 CPU 周期。

我们定义了一个数学模型，其中描述了合作策略的主要方面，并评估了该方案的有效性，并说明了在原则上，服务阻塞概率和服务延迟是如何同时降低的。我们也提出了可能实施的合作方案。

Cognitive Edge Computing based Resource Allocation Framework for Internet of Things

Amjad A, Rabby F, Sadia S, et al. Cognitive Edge Computing based resource allocation framework for Internet of Things[C]//Fog and Mobile Edge Computing (FMEC), 2017 Second International Conference on. IEEE, 2017: 194-200.

不论是主动或是被动的移动设备，都属于物联网的一部分，由于它们的固有属性都是处理资源请求，所以处理能力以及延迟成为主要的优化标准。要实现对云资源的整体优化使用——动态跟踪、监控以及编制框架是需要克服的主要挑战之一。在同样的背景下，在分布式位置上加强计算设备的使用，可以促进物联网的成功;随后，也会促进第五代(5G)无线技术的成功。这为传统的 cloudlet 或云联合的分布式计算设备的未使用资源提供了巨大的潜力。然而,这需要一个高效的微观层面的分布式计算资源跟踪、监测和编制;地理位置上的资源分布以及未使用资源的可用性本质上是时变的。在本文中，我们提出了一种基于认知边缘计算的框架解决方案，以实现对分布式资源的有效利用。这为终端用户提供了 cloudlet 和云联合的计算设备的动态软扩展，以及终端用户的创收途径。