

# 湖南大学

HUNAN UNIVERSITY

## 本科生毕业论文(设计)

论文(设计)题目： 基于移动边缘计算的计算卸

载算法设计

学生姓名： 陆陈定莹

学生学号： 201809020123

专业班级： 通信 1804

学院名称： 信息科学与工程学院

指导老师： 肖竹

2022 年 4 月 28 日

# 湖南大学

## 毕业论文（设计）原创性声明

本人郑重声明：所呈交的论文（设计）是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律后果由本人承担。

学生签名：

日期：20 年 月 日

## 毕业论文（设计）版权使用授权书

本毕业论文（设计）作者完全了解学校有关保留、使用论文（设计）的规定，同意学校保留并向国家有关部门或机构送交论文（设计）的复印件和电子版，允许论文（设计）被查阅和借阅。本人授权湖南大学可以将本论文（设计）的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本论文（设计）。

本论文（设计）属于

1、保 密 ☐，在 \_\_\_\_\_ 年解密后适用本授权书。

2、不保密 ☐。

（请在以上相应方框内打“√”）

学生签名：

日期：20 年 月 日

导师签名：

日期：20 年 月 日

## 基于移动边缘计算的计算卸载算法设计

### 摘 要

边缘计算作为云计算的演化，能够满足来自日新月异的应用程序的不断增多的计算需求。移动设备上的应用把自身需要大量计算的工作，卸载到 MEC 服务器上计算，可以大大改善用户体验，减少时延。但是，由于移动设备上的电池容量是有限的，而且人们很关心电池寿命的问题，反复充电会缩短电池寿命，当电量耗尽时，计算任务会被中断。为了同时能满足良好计算性能和环保的目的，本文通过一种能源获取技术（Energy Harvesting）来捕获可再生能源，例如太阳能、人们运动时产生的动能。本文中，会考虑一个 MEC 系统和带有 EH 技术的移动设备，并且提出一种基于李雅普诺夫优化的计算卸载策略，是一种低复杂度的算法，优化计算卸载的决策、在本地运行的 CPU 频率和卸载到 MEC 服务器上的发送功率。该算法优势体现在，每个时间片内的决策只与系统当前的状态有关联，不需要知道该时间片内任务请求，对于无线信道是否有深度衰落，也无需知道。该算法的最优值，只需要解决目标函数的最值问题，就能在每个时间片内找到最优方案。仿真结果与三种基准算法进行了在运行开销，完成时延和丢失率三个维度的比较，结果表示提出的算法能够显著优于别的基准算法。

**关键词——移动边缘计算，能源获取、频率控制、功率控制、服务质量、李雅普诺夫优化。**

# **The Mobile-edge-computing-based Task Offloading Algorithm Design**

## **Abstract**

As an evolution of cloud computing ,mobile edge computing(MEC) is able to meet the needs of computation-intensive tasks from all walks of applications. Mobile devices(MDs) opt to offload these computational resource hungry tasks to the edge server, which can moderate the experience of user, i.e., latency, considerably. Nevertheless, due to the problem that it is impossible to arm an unlimited battery on the MDs and charging it repeatedly, also a question that human usually concerns much, when the battery runs out, the computation tasks will be interrupted. For the purpose of simultaneously satisfying both pretty computing performance and environment protection, we apply an energy harvesting technique to capture the renewable energy such as solar energy and motion energy. In this paper, we are going to consider a MEC system with MDs armed EH. Besides, we are about to propose a task offloading strategy based on Lyapunov Optimization, which is a low-complexity algorithm jointly optimizing the system actions, CPU frequency and transmitting power. The benefits of this algorithm is that all the actions are i.d.d., which means all of them aren't coupled among different time slot and will not request the CSI and ESI. Also, what we need to obtain derives from solving a deterministic problem. The simulation results compare the LYA. algorithm with the other three baselines in the dimension of the average execution cost, the average complement time and the ratio of drop, which demonstrates the outstanding performance of LYA.

**Key Words :** Mobile-edge Computing ; Energy harvesting ; Frequency control; Power control; QoS; Lyapunov Optimization

## 目录

毕业设计（论文）原创性声明.....	I
毕业设计（论文）版权使用授权书.....	I
摘要.....	II
Abstract.....	III
插图索引.....	V
<b>1 绪论.....</b>	<b>1</b>
1.1 背景.....	1
1.2 国内外研究现状.....	2
1.3 研究方法及其结果概览.....	3
1.4 文章结构.....	3
<b>2 系统模型.....</b>	<b>4</b>
2.1 带有可获得能源设备的边缘计算模型.....	4
2.2 三种计算方式.....	5
2.2.1 本地计算模型.....	5
2.2.2 边缘计算模型.....	6
2.2.3 能量采集模型.....	6
<b>3 问题形成.....</b>	<b>8</b>
3.1 运行开销最小化问题.....	8
3.2 问题分析.....	9
<b>4 动态计算卸载：LODCO 算法.....</b>	<b>11</b>
4.1 LODCO 算法.....	11
4.2 每个时间片内的最优计算卸载.....	13
<b>5 性能分析.....</b>	<b>19</b>
5.1 可行性.....	19
5.2 近似最优性.....	19
<b>6 仿真结果.....</b>	<b>21</b>
6.1 理论结果证实.....	21
6.2 性能评估.....	23
<b>7 总结.....</b>	<b>29</b>

## 插图索引

系统模型图.....	3
算法流程图.....	12
图 6.1 能量上界图.....	22
图 6.2 不同参数下算法的平均运行开销.....	22
图 6.3 平均运行开销和 $V$ 的关系.....	23
图 6.4 需要的电池容量和 $V$ 的关系.....	23
图 6.5 任务平均运行开销和 $\rho$ 的关系.....	24
图 6.6 任务平均完成时间和 $\rho$ 的关系.....	24
图 6.7 任务丢失率和 $\rho$ 的关系.....	25
图 6.8 任务平均运行开销和 $P_H$ 的关系.....	25
图 6.9 任务平均完成时间和 $P_H$ 的关系.....	25
图 6.10 任务丢失率和 $P_H$ 的关系.....	26
图 6.11 任务平均运行开销和 $\tau_d$ 的关系.....	26
图 6.12 任务平均完成时间和 $\tau_d$ 的关系.....	27
图 6.13 任务丢失率和 $\tau_d$ 的关系.....	27
图 6.14 任务平均运行开销和 $d$ 的关系.....	28
图 6.15 任务平均完成时间和 $d$ 的关系.....	28
图 6.16 任务丢失率和 $d$ 的关系.....	28

# 1 绪论

## 1.1 背景

近年来、可穿戴设备，智能手机、平板电脑等移动的设备数量指数型增长，人均拥有可连接设备达到了七八个，这样的趋势加速了物联网的出现的同时，也促进了应用程序的演变和功能的多样化。由于摄像头和传感器在设备上的普及，越来越多的应用程序有了更优越的功能，如 GPS 定位，面部识别和大型多人实时线上游戏等。然而移动设备的资源是有限的，而这些具有新型功能的应用程序需要大量的计算需求，两者之间的矛盾让用户体验，也就是时延变得紧张，这种紧张会导致一个发展良好的应用程序集群出现得越来越晚。

移动边缘计算（MEC）通过宏基站、小型基站等与物联网设备相连，将云计算资源下沉，为移动设备计算这些计算任务密集的应用提供了一种新的途径。以前的云计算方法，是利用谷歌云平台、亚马逊网络等距离可以跨越洲的云平台，这种过于远的距离让数据在广域网中交换带来了很大的时延，进而影响到用户体验。与云计算相比，MEC 能够将本地的计算任务卸到一个 MEC 服务器上，这可以极大地减少时延，避免数据交换的网络拥塞和减少移动设备的能耗并延长电池的使用寿命。

然而，虽然将任务卸载到算力资源丰富的边缘服务器上计算效率是很高的，但是对于现目前的可穿戴设备来说，基本都是靠电池供能的，将任务通过无线信道发送到边缘服务器的过程当中，需要发送功率，这就需要能量消耗，而这个过程就有可能由于能量不够，而带来急剧下降的计算性能。这个问题可以通过给设备配置更加大容量的电池，或者频繁给设备充电来解决。显然大容量的电池会增加硬件开销，这并不有利于我们。另一方面，频繁的充电是让移动设备销量减少的特点之一，并且在有的特殊场景，比如在无线感知网络中用来监控的物联网，也不具备可行性。同时，信息与通信技术带来的快速增长，使得能耗大量增加，这就需要绿色计算来帮忙。可获得能源（EH）技术是解决这个问题一个有前景的办法，该技术能够捕获周围的可再生能源，比如说太阳能、风能和人类活动过程中产生的动能，由于这些能量的易获得性，能够促进可持续性发展和长时间运行。通过把 EH 技术加入移动设备，在使用 MEC 的过程中，可以显著地提高计算性能。然而任何事情都是双刃剑。使用这种能源获取技术

可以给 MEC 系统带来优势，但是伴随着优势也会出现许多问题。尤其是带有电池供能的设备的 MEC 系统，其设计的计算卸载策略不能完全利用可再生能源。所以本文将会提出一种针对带有 EH 设备的 MEC 系统的新的设计方案。

## 1.2 国内外研究现状

对于移动计算系统的计算卸载在最近受到很多关注。有人提出多编码卸载结构<sup>[1]</sup>，来延长电池寿命和增加计算性能。但是明显计算卸载的效率与无线信道衰落状况相关度很高。于是又有人将计算卸载和无线信道的状况结合在一起<sup>[2]</sup>，提出一种随机控制算法来适应随时间变动的无线信道衰落情况。一种一对一场景下的优化时延任务被提出<sup>[3]</sup>。但是这些工作都有一个假设，假设移动设备上的 CPU 的频率是一成不变的，由于 CPU 的能耗与 CPU 的频率正相关，这种假设从能量效率的角度看并不合理。用一种动态电压和频率调制的技术，通过控制 CPU 频率，带有时延限制的本地运行能耗会被最小化。更有用李雅普诺夫研究能耗和时延权衡的多种类计算任务的移动云计算系统，决定卸载决策，任务分配，CPU 频率和网络接口<sup>[4]</sup>。将可获得能源技术应用到通信系统中是看中了其对于实现自我可持续性和绿色通信的潜力<sup>[5]</sup>。因为一些不具有因果的信息，如信道信息(CSI)和能量信息(ESI)，提出一种 DWF 算法来最大化点对点的 EH 衰落信道的吞吐量<sup>[6]</sup>。后来有人用具有因果的信息，衍生到 EH 网络中<sup>[7]</sup>。对于带有 EH 设备的 MEC 系统的设计原则和那些用电池供电的设备的 MEC 系统是不一样的。前者更侧重于最大化计算性能，后者侧重于最小化能耗。有人研究策略的影响，并且基于实际场景中任务产生的随机性，在此基础上研究了随机任务模型条件研究了混合卸载场景下蜂窝链路干扰和 D2D 传输链路干扰对移动用户的计算卸载决策下的有效计算卸载策略<sup>[9]</sup>。针对具有能量收集能力的移动边缘计算系统的计算资源分配问题，提出一种基于李雅普诺夫贪婪优化 (Lyapunov Greedy Optimization, LGO) 算法。构建在设备电池电量逐渐收敛下，移动设备时延与能耗联合成本的动态最小化优化问题<sup>[10]</sup>。提出一种由能量收集技术获取能源进行供能的边缘计算系统模型，将时延、能耗和任务丢弃率作为指标，建立执行代价模型；然后引入能量收集技术模拟用户能量收集过程；最后针对模型时域耦合问题，利用带扰动 Lyapunov 理论将问题转化为逐时隙定性问题，并基于改进的灰狼算法对 CPU 频率和发射功率进行迭代以获取最小的任务执行代价<sup>[11]</sup>。



### 1.3 研究方法及结果概览

①考虑一个带有 EH 技术设备的，并且该设备能够将任务卸载到边缘服务器的 MEC 系统。将运行开销作为性能评估的一项指标，运行开销包括运行任务的时延和任务丢失的惩罚因子。

②最小运行开销问题是一个复杂的多维马尔可夫问题，该问题的传统解法有许多局限性。因此提出了一个基于李亚普诺夫优化的算法。使用将时间分片的系统，在每个时间片内，系统需要决策任务在本地计算还是边缘计算，本地计算要分配多少资源，边缘计算又要分配多少资源。这些决策都可以通过解决固定的目标函数获得。

③仿真表明大量的可获得能源能够得到更短的运行时延，无论是本地运行还是 MEC 服务器运行。算法的性能分析也有总结。结果表明，通过调整一对二元数组的值可以让算法达到最小开销问题的近似最优解。并且，算法可以不了解是否有任务产生，信道的衰落情况如何，每个时间片收集到了多少能量，实际可利用的有多少能量，这些优点让算法更具有普适性。

④第五部分稍微分析了解决的优化问题和原始最优解的差距。并且，提出的算法与三个基准算法进行了比较。结果表明，在大大降低任务丢失率的同时，与基准算法相比也有较低的运行开销。

### 1.4 文章结构

第二部分是系统模型，将介绍文中使用的模型，也即带有可获得能源设备的边缘计算模型。并且会讨论计算部分的模型和能源获得型模型；

第三部分是问题形成，该部分将介绍性能指标：运行开销。同时会形成运行开销最小化（ECM）问题；

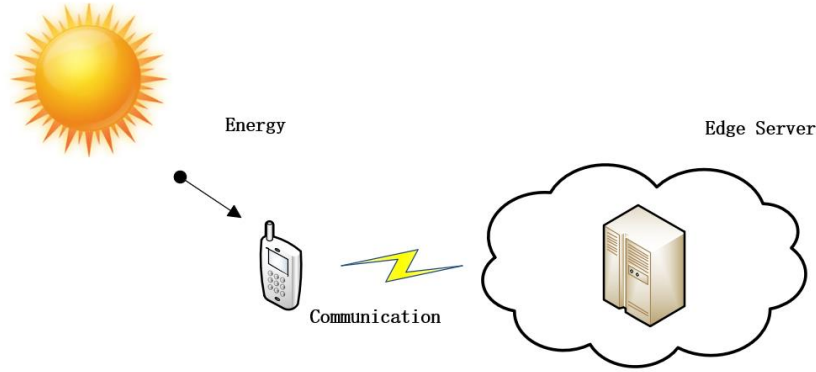
第四部分是算法介绍，该部分，将会提出算法来解决 ECM 问题。会提出一个引理，表面每次处理任务的 CPU 周期频率都是相同的，这能帮助简化决策集。这样就能将原问题稍微修改，得到的问题易于利用 Lyapunov Optimization。同时易于得到解决方案。

第五部分是性能分析，在这部分，将证明算法对于解决修改过的 ECM 问题的可行性，然后分析算法的性能；

第六、七部分则是仿真结果和总结。

## 2 系统模型

### 2.1 带有可获得能源设备的边缘计算模型



(图 1: 系统模型图)

考察带有移动可连接设备和 MEC 服务器的模型。关键在于该可连接设备是带有能够获取能源的设备，并且仅由这个可再生能源供能。对于一个 MEC 服务器来说，是一个小型的数据中心，由电信工作人员管理，将其放置在距离设备  $d_m$  的位置，这样设备能够通过无线信道与其进行通信。设备与 MEC 服务器上通信，该服务器运行得有虚拟机，进而能够执行设备的计算任务。设备通过卸载任务给 MEC 服务器，使得计算性能能够得到显著提升。

本文考虑将时间分片的模型，假设时间被分为很小的时间片，将时间片的长度和时间片的集合分别表示为  $\tau$  和  $\tilde{\mathcal{L}} = \{0, 1, \dots\}$ 。假设无线信道的衰落是独立同分布的，也就是说在每个时间片中，都假设无线信道是静止的。将信道功率增益在  $t$  时刻记作  $h^t$ ，并且  $h^t \sim F_H(x)$ ， $t \in \tilde{\mathcal{L}}$ 。 $F_H(x)$  是  $h^t$  的累计分布函数（CDF）。便于查阅，将系统模型中所用的符号在表格一中呈现。

表格一：文章所用符号和描述

符号	描述
$d$	终端与 MEC 之间的距离长度
$\tilde{\mathcal{L}}$	所有时间片集合
$h^t$	在时间片 $t$ 终端和 MEC 中间的无线信道功率增益
$D_{mobile}^t(D_{server}^t)$	在 $t$ 时刻移动设备（服务器）的运行时
$\phi$	任务丢失开销的权重

$E_H^{max}$	每个时间片能得到的最大能量值
$\{f_w^t\}$	在时间片 $t$ 本地计算对应的时钟周期频率
$B^t$	在 $t$ 时刻初的电池能量
$f_{CPU}^{max}(p_{tx}^{max})$	最高频率（发送功率）门限值
$A(L, \tau_d)$	产生的任务比特流为 $L$ ，任务最长完成时间 $\tau_d$
$E_{mobile}^t(E_{server}^t)$	在 $t$ 时刻移动设备（服务器）的能耗
$e^t(E_H^t)$	在 $t$ 时刻获得（可获得）的能量
$X(W)$	处理一个比特任务输入所需要多少个周期
$p^t$	在时间片 $t$ 边缘计算的设备发射功率
$\{I_j^t\}$	在 $t$ 时刻计算模式指令

## 2.2 三种计算方式

如上表所示， $A(L, \tau_d)$ 代表每个时间片内产生的任务， $L$ （单位 $bit$ ）指的是该任务输入的比特流， $\tau_d$ 指任务运行的最大截止时间，整个运行过程花费的时间不得大于 $\tau_d$ 。这些在应用程序中产生的计算任务假设是服从独立同分布的伯努利分布。用 $\rho$ 来表示在每个时间片中，移动设备产生计算任务的可能性大小。 $\zeta^t = 1$ 表示在该时间片内有计算任务产生， $\zeta^t = 0$ 则表示没有。

对于产生的任意 $A(L, \tau_d)$ ，有三种处理方式：本地计算、边缘计算、任务丢失。考虑到当设备的能量不足时，该计算任务就会被丢弃。将 $I_j^t \in \{0,1\}$ 记作任务计算模式的指令，其中 $j = \{m, s, d\}$ ， $I_m^t$ ， $I_s^t$ ， $I_d^t$ ，分别表示在第 $t$ 个时间片中，计算任务选择在本地执行，卸载到 MEC 服务器执行和丢弃，这三种情况。所以，计算模式选择指令满足下面等式：

$$I_m^t + I_s^t + I_d^t = 1, t \in \tilde{\mathcal{L}} \quad (1)$$

### 2.2.1 本地计算模型

将 $X$ 记作处理一个比特的任务输入所需要的 CPU 周期数目，这个数目对于不同的应用程序而言也是不同的，本文假设 $X$ 的值固定。这也就是说，完全处

理一个任务 $A(L, \tau_d)$ 所需要的 CPU 周期数 $W = LX$ 。对于一个时间片内的 $W$ 个周期，每个周期对应的频率为 $f_\omega^t$ ， $\omega = 1, \dots, W$ 。所以，在第 $t$ 个时间片中，本地运行的时延可以表示为

$$D_{mobile}^t = \sum_{\omega=1}^w (f_\omega^t)^{-1} \quad (2)$$

进而，能耗就可以表示为

$$E_{mobile}^t = \kappa \sum_{\omega=1}^w (f_\omega^t)^2 \quad (3)$$

其中 $\kappa$ 是跟芯片结构有关的电容常数。在此基础上，还假设 CPU 的周期频率有最大值， $f_\omega^t < f_{CPU}^{max}$ ， $\forall \omega = 1, \dots, W$ 。

### 2.2.2 边缘计算模型

为了卸载计算任务，任务 $A(L, \tau_d)$ 的比特流由移动设备发射到边缘服务器上。可以合理地假设，相对于移动设备，服务器有很强的算力，进而可以不考虑任务在服务器上处理的时延。还可以假设，处理完后的 $A(L, \tau_d)$ 任务的输出相对输入而言很小，我们可以选择忽略结果回送到移动设备的时延。将移动设备的发送功率记作 $p^t$ ，有最大值 $p_{tx}^{max}$ 。根据上表和香农公式，在第 $t$ 个时间片能达到的速率 $r(h^t, p^t) = \omega \log_2 \left(1 + \frac{h^t p^t}{\sigma}\right)$ 。由此，当系统决策是产生的任务进行边缘计算时，整个过程的运行时延就是输入比特流的传输时延

$$D_{server}^t = \frac{L}{r(h^t, p^t)} \quad (4)$$

移动设备的能耗

$$E_{server}^t = p^t \cdot D_{server}^t = p^t \cdot \frac{L}{r(h^t, p^t)} \quad (5)$$

### 2.2.3 能量采集模型

能量采集过程被建模为一个连续不断的能量包，在每个时间片初，会有 $E_H^t$

大小的能量到达设备。假设不同的时间片中， $E_H^t$ 是独立同分布的，并不是设备可直接利用的能量，且有最大值 $E_H^{max}$ 。由于自然界中的可再生能源是非理想化的，也就是会出现消失和变化的情况，因此独立同分布很好地符合了这一特性。到达了设备，并且能够被利用的能源记作 $e^t$

$$0 \leq e^t \leq E_H^{max}, t \in \tilde{\mathcal{L}} \quad (6)$$

该能量也能够被用于下一次计算任务（有剩余的话）。我们先假设电池容量足够大。后续将证实在提出的算法之下，电池能量确实有上界，因此，在实际应用中只需要有一定大小的电池容量即可。将在  $t$  时间片初的电池能量记作 $B^t$ 。为了具有普适性，我们假设 $B^0 = 0$ 、 $B^t < +\infty$ ， $t \in \tilde{\mathcal{L}}$ 。为了简便，本文中除了本地运算和卸载到边缘服务器的发送消耗以外的能耗都忽略不计。将移动设备在  $t$  时刻消耗的能量记作 $E(I^t, f^t, p^t)$ ，依赖于选择的计算模型，CPU 周期频率，分配的发送功率。可以表示为

$$E(I^t, f^t, p^t) = I_m^t E_{mobile}^t + I_s^t E_{server}^t \quad (7)$$

满足不等式

$$E(I^t, f^t, p^t) < B^t < +\infty, t \in \tilde{\mathcal{L}} \quad (8)$$

因此，能量的迭代可以表示为

$$B^{t+1} = B^t - E(I^t, f^t, p^t) + e^t \quad (9)$$

使用 EH 的移动设备，计算卸载策略的设计相比于传统电池供能设备的云计算模型更加复杂。特别是，ESI 和 CSI 都需要考虑到，而且不同时间片的电池能量也让系统决策在不同的时间片内做出的决策都不一样。

### 3 问题形成

#### 3.1 运行开销最小化问题

运行时延是用户服务质量最关键的一个指标，因此可以用来衡量一个算法是否优秀。然而，由于自然界中的能量会有变化甚至收集不到，并且移动设备会有电量不够的原因，产生的一些任务会丢失。为了把这种情况考虑进来，使用一个单位的开销来处罚每个丢失的任务。因此，我们将运行开销定义为运行时延和任务丢失开销的加权和

$$cost^t = D(I^t, f^t, p^t) + \phi \cdot \mathbf{1}(\zeta^t = 1, I_d^t = 1) \quad (10)$$

其中 $\phi$ 是任务丢失的权重， $\mathbf{1}(\cdot)$ 是指令函数，

$$D(I^t, f^t, p^t) = \mathbf{1}(\zeta^t = 1) \cdot (I_s^t D_{server}^t + I_m^t D_{mobile}^t) \quad (11)$$

我们假设执行一个任务成功的可能性比丢失的可能性大，也就是说 $\tau_d \leq \phi$ 。

如果确定一个任务要执行，即是说 $I_m^t = 1$ 或者 $I_s^t = 1$ ，该任务应该在最后期限 $\tau_d$ 之内完成，表达式表示就是

$$D(I^t, f^t, p^t) \leq \tau_d \quad (12)$$

至此，ECM 问题可以表示为

$$P_1: \min_{I^t, f^t, p^t, e^t} \lim_{T \rightarrow +\infty} \frac{1}{T} E \left[ \sum_{t=0}^{T-1} cost^t \right]$$

Subject to (1), (6), (8), (12)

$$I_m^t + I_s^t \leq \zeta^t, t \in \tilde{\mathcal{L}} \quad (13)$$

$$E(I^t, f^t, p^t) \leq E_{\max}, t \in \tilde{\mathcal{L}} \quad (14)$$

$$0 \leq p^t \leq P_{tx}^{max} \cdot \mathbf{1}(I_s^t = 1), t \in \tilde{\mathcal{L}} \quad (15)$$

$$0 \leq f_{\omega}^t \leq f_{CPU}^{max} \cdot \mathbf{1}(I_m^t = 1), \omega = 1, \dots, W, t \in \tilde{\mathcal{L}} \quad (16)$$

$$I_m^t, I_s^t, I_d^t \in \{0,1\}, t \in \tilde{\mathcal{L}} \quad (17)$$

上面约束中的（13）表示，系统决策只能有一个本地计算或者边缘计算，或者两者都不成立，任务丢失。约束（14）防止了电池出现放电很多，超过阈值而造成损害。最大可分配的传输功率和 CPU 周期频率的限制由（15）和（16）给出。（17）表明每个任务要么执行、要么不执行。

### 3.2 问题分析

在本文中的状态，由任务是否产生，可利用的再生能量多少，设备电池能量情况和无线信道增益的大小四个部分组成。系统决策有两个部分，一部分是决定最优的能量获取是多少，另一部分是找到最优的本地计算频率和边缘计算发射功率。如上所说，系统采取的决策是独立同分布的，与过去的决策和过去的系统状态是没有关系的。并且，目标函数是长时间的平均运行开销。因此，此前提到的 $P_1$ 是一个马尔科夫问题（MDP）。原则上， $P_1$ 能够用标准的 MDP 算法（the relative value iteration algorithm and the linear programming reformulation approach）优化。这些方法的缺点是，描述每个时间片内的系统状态是有限的情况总数，并且系统采取的决策种数也是需要离散的。举个例子，如果用 $K = 10$ 来给无线信道的状态分级，用 $M = 20$ 来给电池能量分级， $E = 5$ 来描绘可获得的能源，用 $L_T = 10$ 和 $L_F = 10$ 来分别描述发射功率和本地计算频率的等级高低，总共就有  $2KME=2000$  种可能的系统状态。对于 the relative value iteration algorithm，由于在一些状态中会存在 $L_T + L_F + 1$ 种可行的决策。对于 the linear programming reformulation approach，我们需要用  $2KME*(L_T + L_F + 1)$ 解决 linear programming (LP) 问题，这样的话即使是很小的  $W=1000$ ，也是不可行的。并且，用 MDP 算法基于数值迭代，会很难获得解决方法。更重要的是，将系统状态和可行性分层离散化，会导致严重的性能衰退，而且对最优策略进行存储也是一个大挑战。

在后面的部分，将会提出一个 Lyapunov Optimization-based Dynamic Computation Offloading (LODCO) algorithm 来解决 $P_1$ ，该方法有以下好处：  
①没有必要离散化系统状态和操作集了，每个时间片内 LODCO 算法的复杂度也

很低，也不用存储最优决策。

②该算法是不需要关于 CSI、ESI 等先验知识的。

③能够调节 $\langle V, E_{min} \rangle$ 的大小来得到不同的算法性能。理论上来说，LODCO 能够非常接近原始问题 $P_1$ 。

④能够获得移动设备电池大小的最大值，这将会给实际的 EH 组件和能源存储单元一个参考值。



## 4 动态计算卸载：LODCO 算法

### 4.1 LODCO 算法

首先给出引理 1：如果一个任务在  $t$  时刻被本地执行了，那么最优的 CPU 频率都是相同的，也就是说  $f_{\omega}^t = f^t$ ,  $\omega = 1, \dots, W$ 。

这个引理表明我们可以用一个标量  $f^t$  代替一个  $W$  维的向量  $\mathbf{f}^t$ ，减少了优化变量的数目。然而，由于能耗的限制，系统决策在不同的时间片中是相互影响的，增加了设计挑战。这对于带有 EH 的系统来说是一个普遍的问题。我们发现，通过引入一个不为 0 的下界  $E_{min}$ ，在每个时间片输出的能量。这种相互影响现象就能被消除，系统的操作也能够通过忽略 (8) 来优化。这样，我们先引进一个修正版的  $P_1$

$$P_2: \min_{\mathbf{I}^t, f^t, p^t, e^t} \lim_{T \rightarrow +\infty} \frac{1}{T} E \left[ \sum_{t=0}^{T-1} cost^t \right]$$

Subject to (1), (6), (8), (12)–(17)

$$E(\mathbf{I}^t, f^t, p^t) \in \{0\} \cup [E_{min}, E_{max}], t \in \tilde{\mathcal{L}} \quad (18)$$

其中  $0 < E_{min} \leq E_{max}$ 。跟  $P_1$  相比，对于每次执行，根据引理 1，只需要决定一个标量  $f^t$ ，因此  $D_{mobile}^t = W(f^t)^{-1}$ ,  $E_{mobile}^t = W\kappa(f^t)^2$ 。并且， $P_1$  中所有的限制都保留在  $P_2$  当中，额外的一个限制由 (18) 给出。因此， $P_2$  是  $P_1$  的一个收紧版本。将问题  $P_1$  和  $P_2$  的优化值分别记作  $EC_{P_1}^*$  和  $EC_{P_2}^*$ 。接下来的观点会揭示  $EC_{P_1}^*$  和  $EC_{P_2}^*$  之间的关系，会在后续帮助展示提出的算法的近似最优解。

观点 1:  $EC_{P_2}^*$  比  $EC_{P_1}^*$  大一点，但会小于  $EC_{P_1}^*$  加上一个正的常量  $V(E_{min})$ 。

$$EC_{P_1}^* \leq EC_{P_2}^* \leq EC_{P_1}^* + V(E_{min}) \quad (19)$$

通常来说，不等式的右边并不唯一。然而，随着  $E_{min}$  趋近于 0， $V(E_{min})$  会按照如下的推论衰减。

推论 1:

$$\lim_{E_{min} \rightarrow 0} v(E_{min}) = 0$$

上式说明。 $P_2$ 和 $P_1$ 非常接近。因为 $E_{min} \rightarrow 0$ ， $P_2$ 退化为 $P_1$ ，我们也会自然而然的这样认为。然而，由于暂时相关的电量，系统决策在时间上是不独立的。由此，Lyapunov 优化就不能直接应用了。所以借用[8]中提到的一种名为权重扰动的方式来解决这个决策不独立的问题。接下来会详细说明。

定义 1：扰动参数 $\theta$ 满足下面等式

$$\theta \geq \tilde{E}_{max} + V\phi \cdot E_{min}^{-1} \quad (20)$$

其中 $\tilde{E}_{max} = \min \{\max \{\kappa W(f_{CPU}^{max})^2, P_{tx}^{max}\tau\}, V(J \cdot second^{-1})\}$ 可以对算法性能进行调控。

定义 2：另一个变量 $\tilde{B}^t$ 表示成 $\tilde{B}^t = B^t - \theta$ ，看作一个在移动设备上的实际电量的一个转换的版本。

使用虚拟电量队列 $\tilde{B}^t$ 和控制参数 $V$ 能够最小化每个时间片内的运行时延，进而让 $B^t$ 稳定在 $\theta$ 附近。

算法大概过程如下

1、每一个时间片 $t$ 开始时，获得任务要求的指令 $\zeta^t$ ，实际能量队列 $\tilde{B}^t$ ，可获得能量 $E_H^t$ ，和信道功率增益 $h^t$ 。

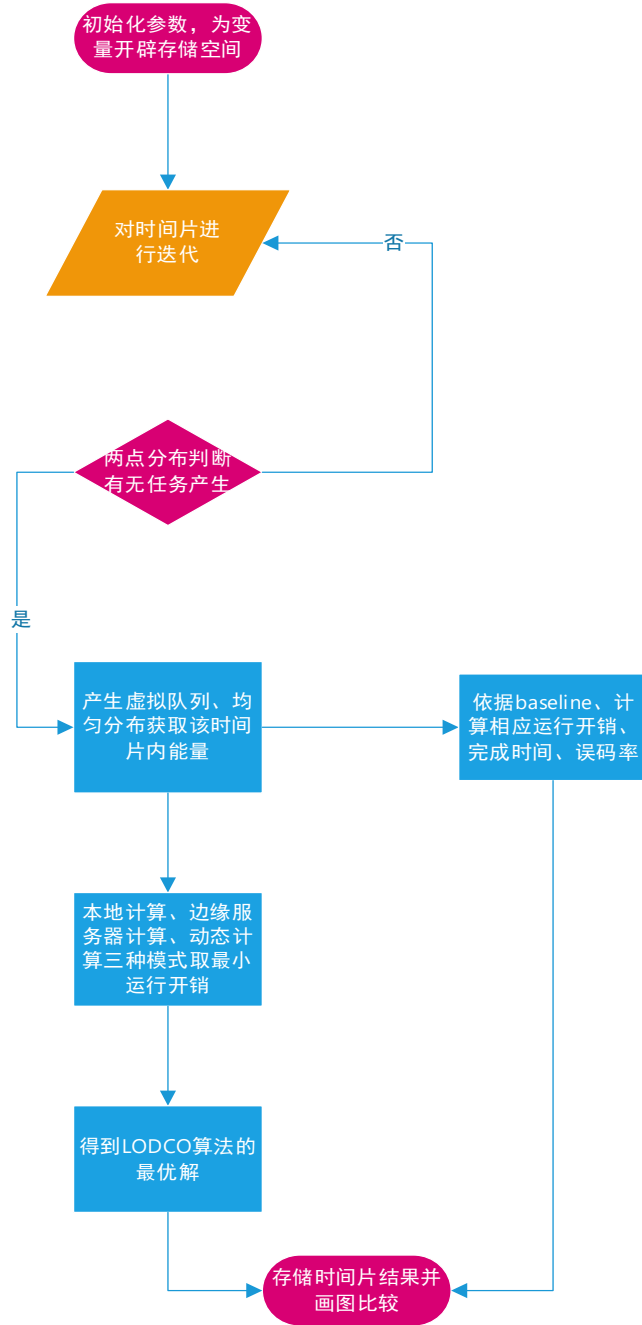
2、通过解决下面的确定性问题，得出 $e^t$ 、 $I^t$ 、 $f^t$ 和 $p^t$

$$\min \tilde{B}^t[e^t - E(I^t, f^t, p^t)] + V[D(I^t, f^t, p^t) + \phi \cdot 1(\zeta^t = 1, I_d^t = 1)]$$

Subject to (1), (6), (12)–(18)

3、根据（9）和定义 2 更新实际队列。

4、时间片迭代。



(算法流程图)

## 4.2 每个时间片内的最优计算卸载

在这个部分，我们会提出每个时间片的最优解决方案，由两部分组成，一部分是最优的能量获取，也就是说确定 $e^t$ ；和确定决策，也就是说确定 $I^t$ ， $f^t$ ， $p^t$ 。该部分得到的结果对于可行性的验证和 LODCO 算法的性能分析是很重要的。

1、最优能量获取：解决如下的线性规划问题就能很直接的得到最优的获取的能量：

$$\min_{0 \leq e^t \leq E_H^t} \tilde{B}^t e^t \quad (21)$$

最优解可由下式给出

$$e^{t*} = E_H^t \cdot \mathbf{1}\{\tilde{B}^t \leq 0\} \quad (22)$$

2、最优计算决策：将 $e^t$ 去耦合后，每个时间片优化问题转化为下面这个优化问题 $P_{CO}$ ：

$$P_{CO}: \min_{I^t, f^t, p^t} -\tilde{B}^t \cdot E(I^t, f^t, p^t) + V[D(I^t, f^t, p^t) + \phi \cdot \mathbf{1}(\zeta^t = 1, I_d^t = 1)]$$

Subject to (1), (12)–(18)

将 $P_{CO}$ 的操作集和目标函数记为 $F_{CO}^t$ 和 $J_{CO}^t(I^t, f^t, p^t)$ 。如果一个时间片内没有任务，也就是 $\zeta^t = 0$ ，由于约束条件，对转换后的优化问题，只能是 $I_m^t = 0, I_d^t = 1, p^t = 0, f^t = 0$ 。

所以，我们更关注每个时间片内有任务产生。先关注本地计算取得最优的计算频率的问题，

$$P_{ME}: \min_{f^t} -\tilde{B}^t \cdot \kappa W(f^t)^2 + V \cdot \frac{w}{f^t}$$

Subject to

$$0 \leq f^t \leq f_{CPU}^{max} \quad (23)$$

$$\frac{w}{f^t} \leq \tau_d \quad (24)$$

$$\kappa W(f^t)^2 \in [E_{min}, E_{max}] \quad (25)$$

通过将 $I_m^t = 1, I_s^t = I_d^t = 0$ 和 $p^t = 0$ 放到 $P_{CO}$ 中可以得到这个问题，对于实际情况来说，本地执行的 $f^t > 0$ 。（24）和（25）分别是时延限制和能量限制。将 $P_{ME}$ 的目标函数记作 $J_m^t(f^t)$ 。从（23）可以看出，手机的算力有限，所以对于 $P_{ME}$ 来说，可行性会有一些限制。这个带有限制的最优解会由观点 2 提出。

观点 2：计算 $f_L = \max\left\{\sqrt{\frac{E_{min}}{\kappa W}}, \frac{w}{\tau_d}\right\}$ 、 $f_U = \min\left\{\sqrt{\frac{E_{max}}{\kappa W}}, f_{CPU}^{max}\right\}$ 。当且仅当 $f_L \leq f_U$

时， $P_{ME}$ 是可行的，最优解由下面的式子给出。

$$f^{t*} \triangleq \begin{cases} f_U, & \tilde{B}^t \geq 0 \text{ or } \tilde{B}^t < 0, f_0^t > f_U \\ f_0^t, & \tilde{B}^t < 0, f_L \leq f_0^t \leq f_U \\ f_L, & \tilde{B}^t < 0, f_0^t < f_L \end{cases} \quad (26)$$

其中  $f_0^t = \left(\frac{V}{-2\tilde{B}^t\kappa}\right)^{\frac{1}{3}}$ 。

证明：先来看一下可行性条件。由于 (24)，为了满足时延的要求， $f^t$  不应该小于  $\frac{w}{\tau_d}$ 。并且，由于 CPU 能耗与  $f^t$  是正相关，电池输出能量限制可以等价表示为

$$\sqrt{\frac{E_{min}}{\kappa W}} \leq f^t \leq \sqrt{\frac{E_{max}}{\kappa W}}, \text{ 通过合并，重写了 CPU 频率如上所述。}$$

接下来，会揭示当  $P_{ME}$  可行时，(26) 的最优性。当  $\tilde{B}^t \geq 0$ ， $J_m^t(f^t)$  与  $f^t$  呈负相关，也就是说当  $f^t = f_U$  时， $J_m^t(f^t)$  达到最小值；当  $\tilde{B}^t < 0$ ， $J_m^t(f^t)$ 、 $-\tilde{B}^t \cdot \kappa W (f^t)^2$  和  $V \cdot \frac{w}{f^t}$  都是  $f^t$  的凸函数。通过将  $J_m^t(f^t)$  置零，我们得到唯一解  $f_0^t = \left(\frac{V}{-2\tilde{B}^t\kappa}\right)^{\frac{1}{3}} > 0$ 。如果  $f_0^t < f_L$ ， $J_m^t(f^t)$  是  $[f_L, f_U]$  的增函数，因此最优解  $f^{t*} = f_L$ 。如果  $f_0^t > f_U$ ， $J_m^t(f^t)$  就是  $[f_L, f_U]$  的减函数，因此最优值  $f^{t*} = f_U$ 。否则，如果  $f_L \leq f_0^t \leq f_U$ ， $f_0^t$  明显是极值点，故最优解  $f^{t*} = f_0^t$ 。

从观点 2 可以看出，最优的 CPU 频率是通过平衡运行时延和能耗得到的。我们发现一个高的 CPU 频率，意味着低的运行时延，可以由大量的可获得能源支撑，这是因为可再生能源其他方面的开销减少，有足够多的能量能够被用来保证用户的体验，下面推论 2 中将会提到的。

推论 2：本地执行的最优 CPU 频率与信道功率增益  $h^t$  是独立的，也不会随着虚拟电量队列的长度而减少。

证明：因为  $P_{ME}$  不依赖于  $h^t$ ，由于  $f_L$  和  $f_U$  都不依赖于  $h^t$ ，并且  $\tilde{B}^t < 0$  时， $f^t$  与  $\tilde{B}^t$  正相关，可以基于 (26) 得到结论， $f^t$  是  $\tilde{B}^t$  的不减函数。

接下来考虑任务在 MEC 服务器执行的情况，最优发送功率可以通过解决下面的  $P_{SE}$  问题得到

$$P_{SE}: \min_{p^t} -\tilde{B}^t \cdot \frac{p^t L}{r(h^t, p^t)} + V \cdot \frac{L}{r(h^t, p^t)}$$

Subject to

$$0 \leq p^t \leq p_{tx}^{max} \quad (27)$$

$$\frac{L}{r(h^t, p^t)} \leq \tau_d \quad (28)$$

$$\frac{p^t L}{r(h^t, p^t)} \in [E_{min}, E_{max}] \quad (29)$$

这个问题就是将 $P_{CO}$ 中 $I_s^t = 1$ ,  $I_m^t = I_d^t = 0$ 和 $f^t = 0$ 得到的。(28)和(29)就是时延限制和电池输出能量限制。将 $P_{SE}$ 的目标函数记作 $J_s^t(p^t)$ 。由于无线信道的衰落，可能会有卸载不可行的时候。为了得到可行性条件和最优解，先提出一个引理帮助分析。

引理 2: 对于 $h > 0$ 时,  $g_1(h, p) \triangleq \frac{p}{r(h, p)}$ , 是 $p$ 的一个增函数 ( $p > 0$ ) 取值范围是 $(\sigma \ln 2(\omega h)^{-1}, +\infty)$ 。

基于引理 2, 结合(27) - (29)的限制条件, 变成一个不等式, 得到一个可行性条件, 如接下来的引理 3 所示。

引理 3: 当且仅当 $p_L^t \leq p_U^t$ 时,  $P_{SE}$ 可行。 $p_L^t$ 和 $p_U^t$ 的定义如下

$$p_L^t \triangleq \begin{cases} p_{L, \tau_d}^t, \frac{\sigma L \ln 2}{\omega h} \geq E_{min} \\ \max\{p_{L, \tau_d}^t, p_{E_{min}}^t\}, \frac{\sigma L \ln 2}{\omega h} < E_{min} \end{cases}$$

$$p_U^t \triangleq \begin{cases} \min\{p_{tx}^{max}, p_{E_{max}}^t\}, \frac{\sigma L \ln 2}{\omega h} < E_{max} \\ 0, \frac{\sigma L \ln 2}{\omega h} \geq E_{max} \end{cases} \quad (30)$$

在(30)中,  $p_{L, \tau_d}^t \triangleq \left(2^{\frac{L}{\omega \tau_d}} - 1\right) \sigma / h^t$ ,  $p_{E_{min}}^t$ 是在 $\sigma \ln 2(\omega h)^{-1} < E_{min}$ 的情况下, 等式 $pL = r(h^t, p)E_{min}$ 的唯一解, 而 $p_{E_{max}}^t$ 是在 $\sigma \ln 2(\omega h)^{-1} < E_{max}$ 的情况下, 等式 $pL = r(h^t, p)E_{max}$ 的唯一解。

现在给出 $P_{SE}$ 的最优解。

主张 3: 如果 $p_L^t \leq p_U^t$ ,  $P_{SE}$ 可行, 最优解由下式给出

$$p^{t*} \triangleq \begin{cases} p_U^t, \tilde{B}^t \geq 0 \text{ or } \tilde{B}^t < 0, p_U^t < p_0^t \\ p_L^t, \tilde{B}^t < 0, p_L^t > p_0^t \\ p_0^t, \tilde{B}^t < 0, p_L^t \leq p_0^t \leq p_U^t \end{cases} \quad (31)$$

其中的 $p_0^t$ 是等式 $\psi(h^t, p, \tilde{B}^t) = 0$ 的唯一解,  $(h^t, p, \tilde{B}^t) \triangleq -\tilde{B} \log_2 \left(1 + \frac{hp}{\sigma}\right) - \frac{h}{(\sigma + hp) \ln 2} (V - \tilde{B}p)$ 。

证明: 当 $\tilde{B}^t \geq 0$ 时, 因为 $J_s^t(p^t)$ 的两部分,  $-\tilde{B}^t \cdot \frac{p^t L}{r(h^t, p^t)}$ 和 $V \cdot \frac{L}{r(h^t, p^t)}$ 都是 $p^t$ 的不增函数, 所以有 $p^{t*} = p_U^t$ 。当 $\tilde{B}^t < 0$ 时, 定义一个 $g_2(h, p, \tilde{B}) \triangleq -\frac{\tilde{B}p}{r(h, p)} + \frac{V}{r(h, p)}$ , 因此, 对 $p$ 求偏导可得

$$\begin{aligned} \frac{dg_2(h, p, \tilde{B})}{dp} &= \frac{-\tilde{B}^t \log_2 \left(1 + \frac{h^t p^t}{\sigma}\right) - \frac{h^t}{(h^t p^t + \sigma) \ln 2} (-\tilde{B}^t p + V)}{\omega \log_2^2 \left(1 + \frac{h^t p^t}{\sigma}\right)} \\ &\triangleq \frac{\psi(h^t, p, \tilde{B}^t)}{\omega \log_2^2 \left(1 + \frac{h^t p^t}{\sigma}\right)} \end{aligned} \quad (32)$$

由于 $\frac{d\psi(h^t, p, \tilde{B}^t)}{dp} > 0$ ,  $\psi(h^t, p, \tilde{B}^t)$ 是 $p$ 的增函数。并且, 因为 $\psi(h^t, 0, \tilde{B}^t) = -\frac{h^t V}{\sigma \ln 2} < 0$ , 并且 $\lim_{p \rightarrow +\infty} \psi(h^t, p, \tilde{B}^t) = +\infty$ , 由解的唯一性可知, 存在一个唯一解 $p_0^t \in (0, +\infty)$ , 满足 $\psi(h^t, p_0^t, \tilde{B}^t) = 0, \forall h^t > 0$ 。当 $h^t$ 和 $p^t$ 都大于 0 时, (32) 的分母是正的, 当 $p \in (0, p_0^t), \frac{dg_2(h, p, \tilde{B})}{dp} < 0$ , 也就是说 $g_2(h, p, \tilde{B})$ 是减函数; 当 $p \in (p_0^t, +\infty), \frac{dg_2(h, p, \tilde{B})}{dp} > 0$ , 也就是说 $g_2(h, p, \tilde{B})$ 是增函数。综上所述, 当 $\tilde{B} < 0$ 且 $p_L^t \leq p_0^t \leq p_U^t$ 的时候,  $J_s^t(p^t)$ 在 $[p_L^t, p_0^t)$ 是不增函数, 在 $(p_0^t, p_U^t]$ 是不减函数, 因此最优解是 $p^{t*} = p_0^t$ ; 当 $\tilde{B} < 0$ 且 $p_L^t > p_0^t$ 时,  $J_s^t(p^t)$ 在可行域内是一个不减函数, 因此 $p^{t*} = p_L^t$ ; 当 $\tilde{B} < 0$ 且 $p_U^t < p_0^t$ 时,  $J_s^t(p^t)$ 可行域内是一个不增函数, 所以最优解 $p^{t*} = p_U^t$ 。

与本地运行的情况类似, 最优发送功率也有一个性质如推论 3 所述。

推论 3: 对于一个给定的 $h^t$ 让 $P_{SE}$ 可行, 最优的发射功率关于虚拟队列 $\tilde{B}$ 是一

个不减函数。

基于观点 2 和 3，最优的计算卸载决策可以通过估计  $P_{CO}$  对三种模式的最优值估计来得到，最优值可以用下式表示

$$\langle I^{t*}, f^{t*}, p^{t*} \rangle = \arg \min_{\langle I^t, f^t, p^t \rangle \in F_{CO}^t} J_{CO}^t(I^t, f^t, p^t)$$

$$J_{CO}^t(I^t, f^t, p^t) = I_m^t \cdot J_m^t(f^t) + I_s^t \cdot J_s^t(p^t) + 1(I_d^t = 1, \zeta^t = 1) \cdot V\phi$$

其中  $V\phi$  代表任务丢失时的惩罚值。



## 5 性能分析

### 5.1 可行性

通过展示每个时间片内的最优解能满足（8）中的能量消耗限制，来证明算法可行。

观点 4：在每个时间片的最优解下，当  $B^t < \tilde{E}_{max}$  时， $I_d^t = 1, I_m^t = I_s^t = 0, f^t = 0, p^t = 0$ ，将不会超出（8）中的约束，也就是说 LODCO 算法对于  $P_2$  是可行的。

基于能量获取决策和主张 4，可以看到电池能量在如下的推论当中是一个范围量。

推论 4：在 LODCO 算法下，移动设备的电池能量在  $[0, \theta + E_H^{max}]$ ， $t \in \tilde{\mathcal{L}}$ 。

证明：根据观点 4，电池能量输出限制不被破坏，所以  $B^t$  的下界很明显。基于（22）中的最优能量获取， $B^t$  的上界可以得到：假设  $\theta < B^t \leq \theta + E_H^{max}$ ，由于  $e^{t*} = 0$ ，有  $B^{t+1} \leq B^t \leq \theta + E_H^{max}$ ；另一种情况，如果  $B^t \leq \theta$ ，由于  $e^{t*} = E_H^{max}$ ，有  $B^{t+1} \leq B^t + e^{t*} \leq \theta + e^{t*} \leq \theta + E_H^{max}$ 。无论哪种，都可以得到  $B^t \in [0, \theta + E_H^{max}]$ ， $\forall t \in \tilde{\mathcal{L}}$ 。

电池能量的有界，让该算法得实施有了保障。

### 5.2 近似最优性

在这一部分，将分析算法的性能，将会提出一个辅助优化问题  $P_3$  建立起算法能达到的性能和  $P_2$  的最优性能之间的桥梁。

首先会定义李雅普诺夫函数

$$L(\tilde{B}^t) = \frac{1}{2}(\tilde{B}^t)^2 = \frac{1}{2}(B^t - \theta)^2 \quad (34)$$

接下来是李雅普诺夫的漂移函数和漂移加惩罚函数的定义

$$\Delta(\tilde{B}^t) = E[L(\tilde{B}^{t+1}) - L(\tilde{B}^t) | \tilde{B}^t] \quad (35)$$

$$\Delta_v(\tilde{B}^t) = \Delta(\tilde{B}^t) + VE[D(I^t, f^t, p^t) + \phi \cdot 1(\zeta^t = 1, I_d^t = 1) | \tilde{B}^t] \quad (36)$$

在下面的引理，我们可以得到一个 $\Delta_V(\tilde{B}^t)$ 的上界，这在分析 LODCO 性能当中非常重要。

引理 4: 对于 $P_2$ 任意的可行决策 $e^t, I^t, f^t, p^t$ 来说， $\Delta_V(\tilde{B}^t)$ 的上界由下式给出

$$\begin{aligned} \Delta_V(\tilde{B}^t) \leq & C + E[\tilde{B}^t[e^t - E(I^t, f^t, p^t)] + V[D(I^t, f^t, p^t) + \phi \cdot 1(\zeta^t = 1, I_d^t \\ & = 1)]|\tilde{B}^t] \end{aligned} \quad (37)$$

其中  $C = \frac{1}{2}((E_H^{\max})^2 + (\tilde{E}_{\max})^2)$ 。

为了帮助分析，定义了一个如下的辅助问题 $P_3$

$$P_3: \min_{I^t, f^t, p^t, e^t} \lim_{T \rightarrow +\infty} \frac{1}{T} E \left[ \sum_{t=0}^{T-1} cost^t \right]$$

Subject to (1), (6), (12) – (18)

$$\lim_{T \rightarrow +\infty} \frac{1}{T} E[E(I^t, f^t, p^t) - e^t] = 0 \quad (38)$$

在 $P_3$ 当中，平均消耗的能量等于平均获得的能量，这也就是说 $P_2$ 当中的能量限制被（38）所替代。将 $P_3$ 中的最优值记作 $EC_{P_3}^*$ 。下面的引理将会说明 $P_3$ 是 $P_2$ 的一种释放。

引理 5:  $P_3$ 是 $P_2$ 的一种释放，意思是 $EC_{P_3}^* \leq EC_{P_2}^*$ 。

因为对于 $P_2$ 的任意一个可行解都可以用于 $P_3$ 。

## 6 仿真结果

该部分将会证实仿真结果，通过仿真估计性能。本次仿真中，参数设定如下， $E_H^t \sim U(0, E_H^{max})$ ，EH 功率  $P_H = E_H^{max}(2\tau)^{-1}$ ，信道功率增益  $h^t \sim E\left(g_0 \left(\frac{d_0}{d}\right)^4\right)$ ，其中  $g_0 = -40dB$  是路损常量， $d_0 = 1m$  是参考距离。 $\kappa = 10^{-28}$ ， $\tau = \phi = 2ms$ ， $\omega = 1MHz$ ， $\sigma = 10^{-13}W$ ， $P_{tx}^{max} = 1W$ ， $f_{CPU}^{max} = 1.5GHz$ ， $E_{max} = 2mJ$ ， $L = 1000bits$ 。每个字节需要的 CPU 周期数为 5900。 $P_H = 12mW$ ， $d = 50m$ ， $\tau_d = 2ms$ 。为了对比，引入三个准线方案，分别是带有贪婪能量分配的本地计算，贪婪能量分配的边缘计算，贪婪能量分配的动态卸载，这些都将最小化每个时间片内的运行开销。下面分别介绍：

**本地计算：**在有任务产生，也就是  $\zeta^t = 1$  时，最大的可行 CPU 频率

$$f_U^t = \min \{f_{CPU}^{max}, \sqrt{\frac{\min \{B^t, E_{max}\}}{\kappa W}}\}$$

如果满足  $\frac{W}{f_U^t} \leq \tau_d$ ，计算任务就会以  $f_U^t$  的频率在本地执行。否则任务就会丢失。

注意这种方案不能执行计算卸载。

**边缘计算：**在有任务产生，也就是  $\zeta^t = 1$  时，在满足  $\sigma L \ln 2(\omega h^t)^{-1} < \min \{B^t, E_{max}\}$  计算最大可行的发送功率

$$p_U^t = \min \{p_{tx}^{max}, p_{\min \{B^t, E_{max}\}}^t\}$$

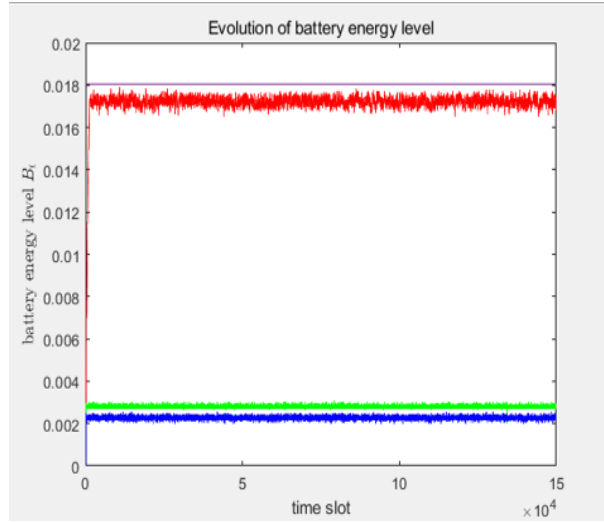
其中  $p_{\min \{B^t, E_{max}\}}^t$  是等式  $pL = r(h^t, p) \min \{B^t, E_{max}\}$  的唯一解。如果满足  $L/r(h^t, p_U^t) \leq \tau_d$ ，移动设备就将以  $p_U^t$  为发送功率将任务卸载到 MEC 服务器上，注意这种方法不会在本地执行。

**动态计算：**在有任务产生，也就是  $\zeta^t = 1$  时，如本地计算和边缘计算中那样计算  $f_U^t$  和  $p_U^t$ 。并检查他们能否满足时延，可行的方案将会是时延较小的那一个，如果两种都不满足时延约束，则任务丢失。

### 6.1 理论结果证实

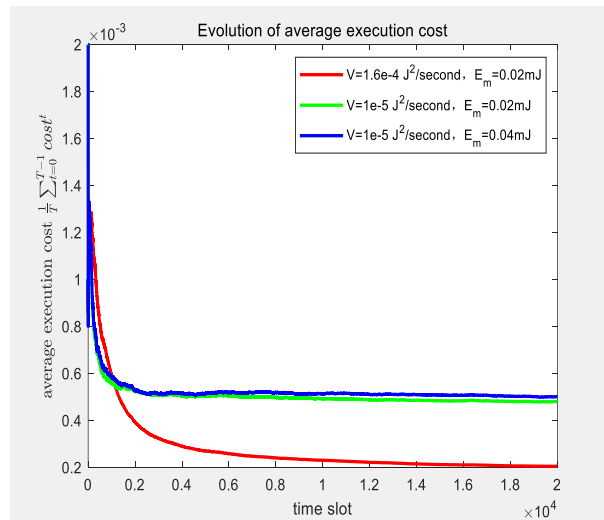
在该部分，我们将会证实 LODCO 算法的可行性和在观点 4、引理 4 和推论 1 中的近似最优性。 $\theta$  的值取 (20) 的右值。在图 6.1 中，电量的多少用来描绘针对  $P_2(P_1)$  的 LODCO 算法的可行性。首先，我们观察能量在一开始是一直积累的，

后来稳定在一定范围内。这是由于提出的算法中，李雅普诺夫的漂移加惩罚函数的上界在每个时间片内都被最小化了。从曲线当中可以看出，将 $V$ 的值设置的更大或者将 $E_{min}$ 设置得更小，稳定的电量变得更高，这与（20）当中的扰动参数的定义所一致。并且，我们可以发现电量的范围在 $[0, \theta + E_H^{max}]$ ，这证实了引理 4，并且证实能量限制没有被破坏，也就是观点 4 成立。平均运行时延关于时间的变化显示在图 6.2 中。从中可以看出将 $V$ 的值设置的更大或者将 $E_{min}$ 设置得更小会得到一个长期性的更小的平均运行开销。但是，这个算法收敛得更慢了。除此之外，如果 $\langle V, E_{min} \rangle$ 能够合适地选择的话，提出的算法与三个基准算法相比能得到可观的性能提升。



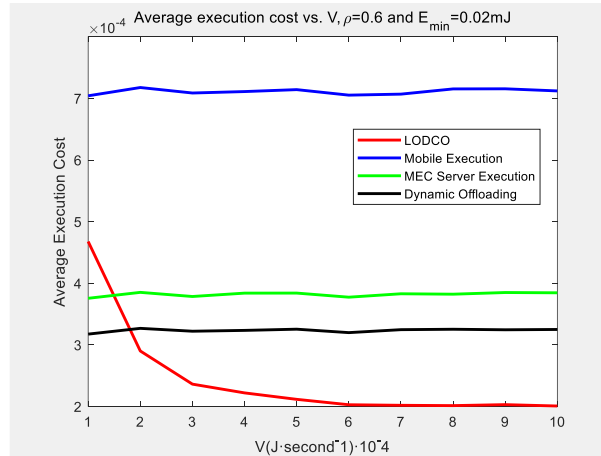
（图 6.1 能量上界图）

其中黑线表示电池能量上界

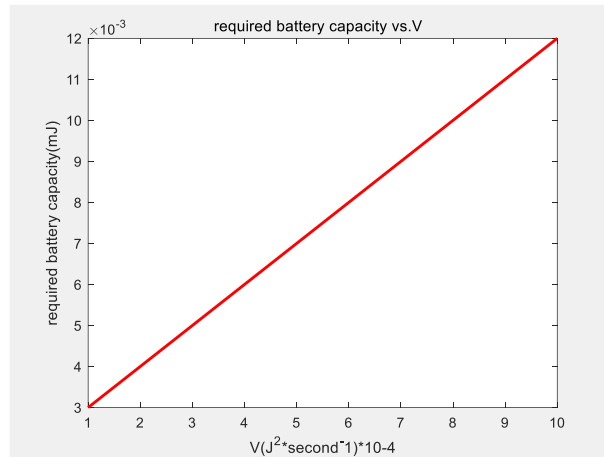


（图 6.2 不同参数下算法的平均运行开销）

平均运行开销和 $V$ 的关系如图 6.3 所示，可以看到本文提出的算法的运行开销与 $V$ 是负相关的，最终会收敛到 $P_2$ 的最优值，这证实了，定理 1 中的近似最优解。然而，从图 6.4 中可以看出需要的电池容量关于 $V$ 是线性增长的趋势，因为 $\theta$ 关于 $V$ 是线性正相关的。综上两点， $V$ 应该合适选择以平衡可达到的性能，收敛时间和需要的电池容量。比方说，如果电池可以容纳 18mJ，则可以将取  $V = 1.6 \times 10^{-4} J^2 \cdot second^{-1}$ ，相比于动态计算、边缘计算和本地计算有依次增长的性能提升。



（图 6.3 平均运行开销和 $V$ 的关系）

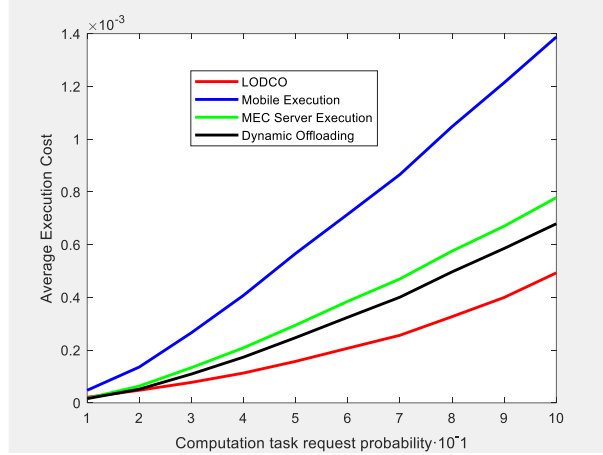


（图 6.4 需要的电池容量和 $V$ 的关系）

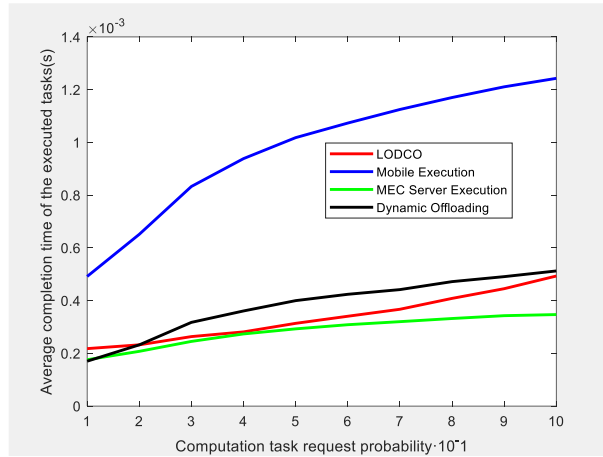
## 6.2 性能评估

这个部分将会展示提出算法的有效性和展示许多系统参数对于算法的影响。首先，是任务产生的可能性 $\rho$ 对于系统性能的影响，包括运行开销，运行任务的平均完成时间和丢失率分别在图 6.5、图 6.6、图 6.7 中展示。在图 6 中，可以看到运行开销随着 $\rho$ 的增长而增长，这符合我们的直觉，并且 LODCO 算法比起

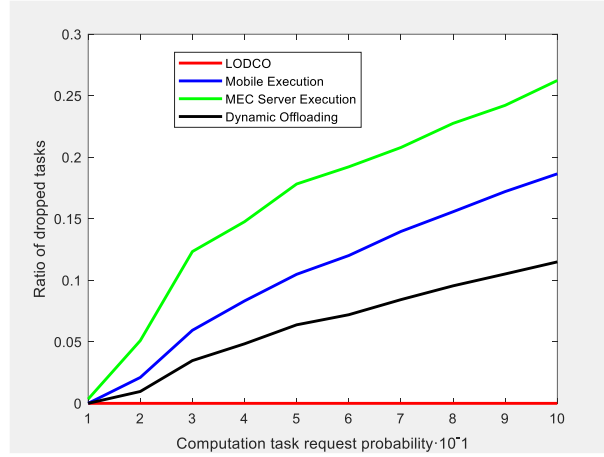
其余三个基线有很好的提升。图 7 和图 8 中展示了运行任务的平均执行时间和丢失率。从中可以看出 LODCO 算法能够达到一个近乎接近零的丢失率，而其他三个基线丢失率随着 $\rho$ 的上升而上升。从执行的任务的平均完成时间来看，当 $\rho$ 值比较小的时候，LODCO 算法明显优于其他三个基线。但是当 $\rho$ 比较大时，这项指标要比边缘计算稍微大一些。这是因为，为了最小化运行开销，LODCO 算法以一个小的运行时延为代价来减少任务丢失率。



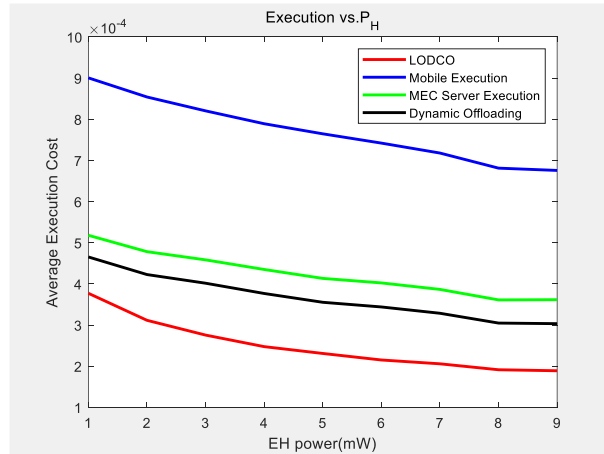
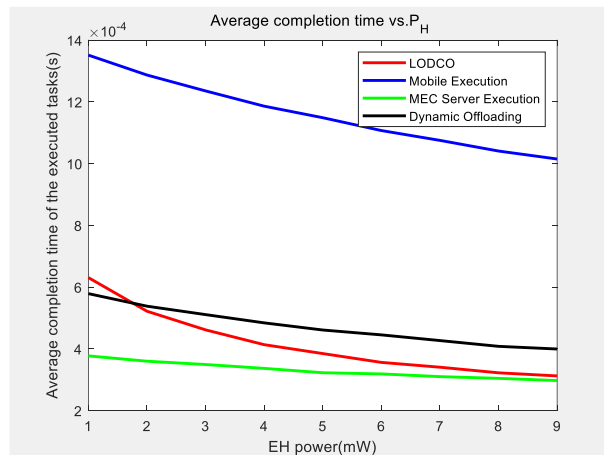
(图 6.5 任务平均运行开销和 $\rho$ 的关系)

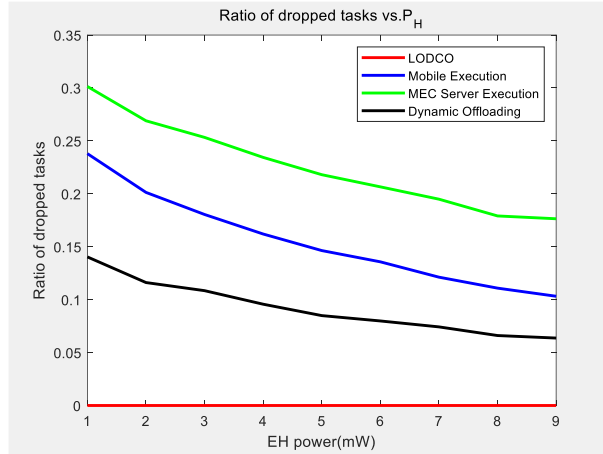


(图 6.6 任务平均完成时间和 $\rho$ 的关系)

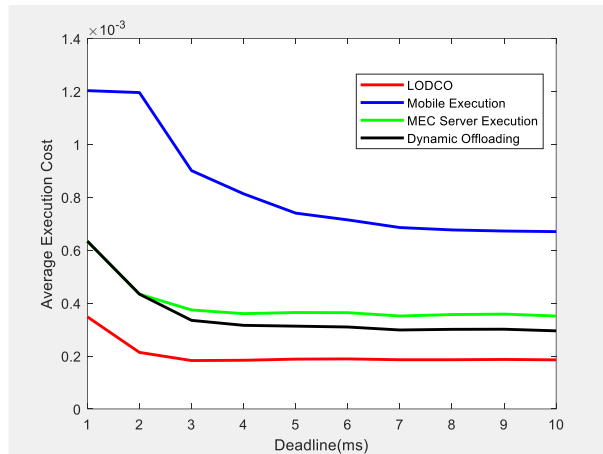

 (图 6.7 任务丢失率和 $\rho$ 的关系)

系统性能和 EH 速率，也就是 $P_H$ 的关系如图 6.8 所示，再次看到 LODCO 算法的有效性。并且，我们可以看到运行开销随着 EH 速率的增长而减少，因为消耗可再生能源不会有任何开销。跟运行开销相似，任务丢失率与 EH 速率也是负相关。有趣的是，在 LODCO 算法下，增加 EH 速率并不能显著减少平均完成时间。比方说，在 $\rho = 0.6$ ， $P_H$ 从 6 到 7mW 时，算法只减少了 0.07ms 的平均完成时间。

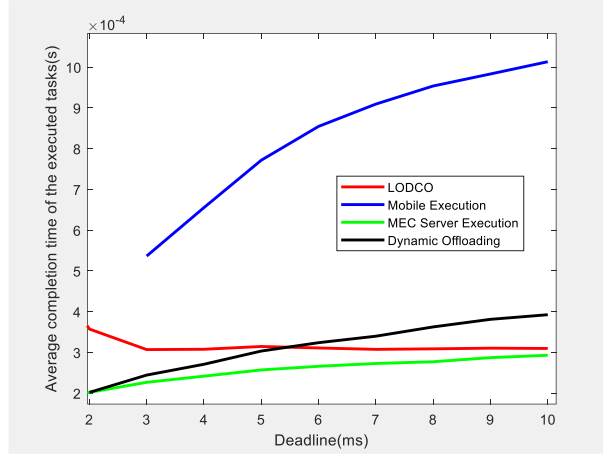
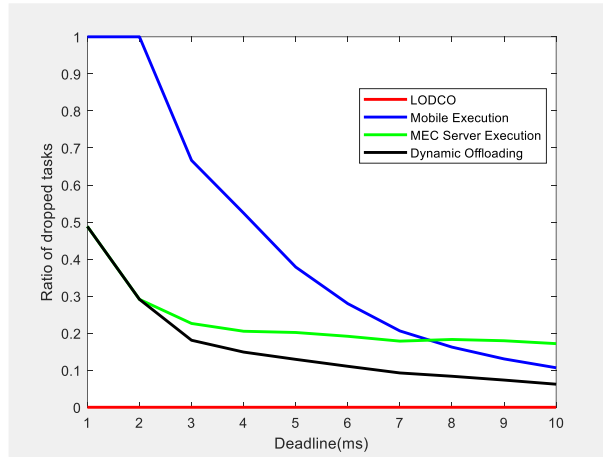

 (图 6.8 任务平均运行开销和 $P_H$ 的关系)


(图 6.9 任务平均完成时间和 $P_H$ 的关系)(图 6.10 任务丢失率和 $P_H$ 的关系)

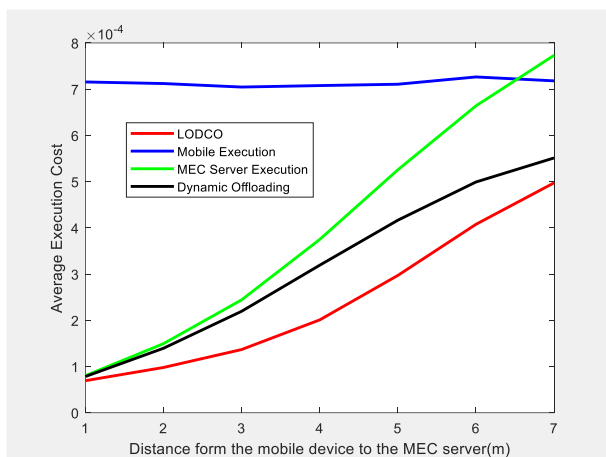
在图 6.11、图 6.12、图 6.13 中，我们展示了运行最大时延 $\tau_d$ 和系统性能之间的关系。当 $\tau_d$ 减小（也就是说计算需求变得更加紧迫），四个策略的运行开销和任务丢失率都提升了，但是平均运行时间是减少的。图中可以看出，当 $\tau_d \leq 0.4ms$ ，本地计算得到的平均运行开销是一个常量 $\rho\phi$ ，而且任务丢失率是100%。同时边缘计算和动态计算的图线重叠。在这样的条件下，移动设备将由于硬件限制不能够执行任何计算任务，所有的任务都需要卸载到边缘服务器上。图 14 证实了边缘计算在 $\tau_d = 0.2ms$ 的时候，任然有50%的成功率。注意到当 $\tau_d \leq 0.8ms$ ，LODCO 达到的平均完成时间比其他两个有卸载任务的算法还要稍微高一些，但是任务丢失率少了20%。这种现象代表 LODCO 算法倾向于通过延长平均完成时间来避免任务的丢失为了达到一个最小的运行开销。

(图 6.11 任务平均运行开销和 $\tau_d$ 的关系)

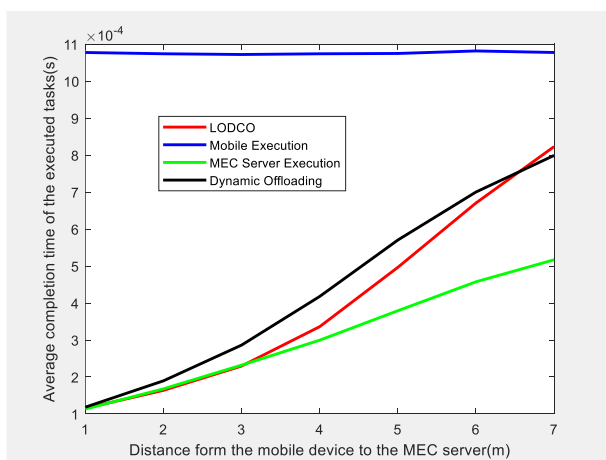


(图 6.12 任务平均完成时间和 $\tau_d$ 的关系)(图 6.13 任务丢失率和 $\tau_d$ 的关系)

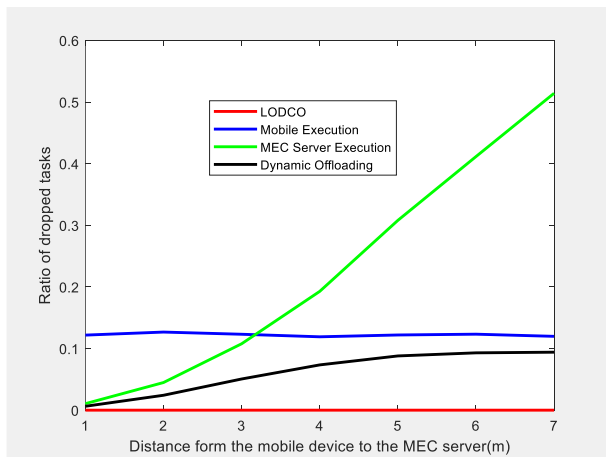
最后，我们将在图 6.14、图 6.15、图 6.16 展示系统性能和 $d$ （也就是从移动设备到 MEC 服务器的距离）的关系。可以看到所有算法的性能随着 $d$ 的增长而恶化，图 15 中看到，当移动设备接近与 MEC 服务器时，三个算法的性能都很接近，并且远远好过本地计算。在这样的场景下，由于很小的路损，移动设备只需要一点获取的能量来将任务卸载到 MEC 服务器。当 $d = 80m$ 时，卸载到边缘服务器计算已经不比本地计算性能优越了，但是 LODCO 算法仍然有较好的性能。在图 16 中可以看到，虽然边缘计算的平均运行时间低于其他三个算法，但是任务丢失率性能大大降低了。作为对比，LODCO 算法还是以接近于 0 的任务丢失率一骑绝尘，同时平均完成时间也有所降低。



(图 6.14 任务平均运行开销和 $d$ 的关系)



(图 6.15 任务平均完成时间和 $d$ 的关系)



(图 6.16 任务丢失率和 $d$ 的关系)

## 7 总结

本次我们研究了一种带有 EH 设备的边缘计算系统。运行开销，代表运行时延和任务失败开销，作为性能指标。提出了一个基于李雅普诺夫的动态计算卸载决策算法。这是一个低复杂度的算法。我们发现 CPU 周期频率（发送功率）关于电池能量的单调性，揭示了 EH 对系统操作的影响。性能分析表明了算法的近似最优性。仿真结果表明了提出的 LODCO 算法不仅在运行开销方面优于其他三个算法，并且以一点运行时延为代价换取了优越的任务丢失性能。我们的研究为将来带有 EH 设备的边缘计算系统提供了可行的设计方案。可以考虑该算法扩展到多个移动设备的场景，或者考虑资源受限的 MEC 服务器。

## 参考文献

- [1] E. Cuervo et al., “MAUI: Making smartphones last longer with code offload,” in Proc. ACM Int. Conf. Mobile Syst. Appl. Serv. (MobiSys), San Francisco, CA, USA, Jun. 2010, pp. 49 – 62.
- [2] D. Huang, P. Wang, and D. Niyato, “A dynamic offloading algorithm for mobile computing,” IEEE Trans. Wireless Commun., vol. 11, no. 6, pp. 1991 – 1995, Jun. 2012.
- [3] O. Munoz, A. Pascual-Iserte, and J. Vidal, “Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading,” IEEE Trans. Veh. Technol., vol. 64, no. 10, pp. 4738 – 4755, Oct. 2015.
- [4] J. Kwak, Y. Kim, J. Lee, and S. Chong, “DREAM: Dynamic resource and task allocation for energy minimization in mobile cloud systems,” IEEE J. Sel. Areas Commun., vol. 33, no. 12, pp. 2510 – 2523, Dec. 2015.
- [5] G. Piro et al., “HetNets powered by renewable energy sources: Sustainable next-generation cellular networks,” IEEE Internet Comput., vol. 17, no. 1, pp. 32 – 39, Jan./Feb. 2013.
- [6] O. Ozel, K. Tutuncuoglu, J. Yang, S. Ulukus, and A. Yener, “Transmission with energy harvesting nodes in fading wireless channels: Optimal policies,” IEEE J. Sel. Areas Commun., vol. 29, no. 8, pp. 1732 – 1743, Sep. 2011.
- [7] L. Huang and M. J. Neely, “Utility optimal scheduling in energy harvesting networks,” IEEE/ACM Trans. Netw., vol. 21, no. 4, pp. 1117 – 1130, Aug. 2013.
- [8] Y. Y. Mao and J. Zhang, “Dynamic Computation Offloading for Mobile-Edge Computing With Energy Harvesting Devices,” IEEE Journal On Selected Areas In Communication, Vol. 34, No. 12, December 2016.

- [9] 胡桂生, 贾云健. 面向移动边缘计算网络的计算卸载策略研究. 重庆大学, 2019.
- [10] 李长云, 黎建波, 徐曦, 李亭立. 具有能量收集设备的移动边缘计算资源分配研究. 《系统仿真学报》, 2021.
- [11] 蒋欣秀, 杨俊东, 杨志军, 李波, 丁洪伟. 移动边缘计算中支持能量收集的卸载策略, 云南大学, 2022.