



# 机会主义的移动人群计算：任务依赖性 基于工作的盗窃

圣志义，费尔南多，圣罗克，圣罗克，  
阿扎德。

gharineiat@deakin.edu.au

澳大利亚吉隆市迪肯大学信息技术学院

澳大利亚吉隆市迪肯大学pubudu.  
pathirana@deakin.edu.au工程学院

## 摘要

移动设备无处不在、异构和资源受限。在移动设备中执行复杂任务是对资源的需求和耗时的，这迫使开发人员将复杂任务的部分转移到云或边缘计算资源中。由于间歇性的互联网连接、远程资源的不可用性、高成本、延迟和移动设备的能量有限，任务卸载变得越来越具有挑战性。移动设备用户通常被其他移动设备包围，可以利用这些设备来协作计算资源密集型的任务。在工作共享框架的帮助下，设备进行通信和协作是可行的。然而，一些移动设备无法计算任务的复杂部分，有些设备可以在加速模式下进行计算。在这个演示中，我们介绍了一个使用窃取工作算法的协作移动人群计算框架。该算法允许基于设备的计算能力和任务依赖性与协作设备共享工作。实验表明，采用蜜蜂-t框架，与整体执行大型计算密集型任务相比，有相当大的性能提高，并节省能源。

## 1介绍

移动应用程序的开发历来都是考虑到终端用户移动设备的资源有限，特别是对于视频编辑和数据分析等复杂的应用程序。为了克服移动计算固有的资源限制，应用程序开发人员开始集成云或边缘资源来支持复杂的计算[9, 11, 16]。虽然云计算（CC）支持按需访问计算资源[3]，但在延迟和带宽方面存在问题，特别是在有间歇性连接的情况下。移动边缘计算（MEC）的概念作为一种解决方案引入了额外的计算服务器，即在远程云和最终用户设备之间[10, 15]。在MEC中，边缘服务器主要负责资源共享和与远程资源的通信，以便进行任务卸载[8]。在危险的环境中，具有节点的移动性，

为个人或课堂使用制作部分或全部作品的数字或硬拷贝是免费的，但副本不是为利润或商业利益而制作或分发，且副本载有本通知和全部引用。必须尊重本作品的第三方组件的版权。对于所有其他用途，请与所有者/作者联系。

20mobi.com2022年10月17-21日，澳大利亚新南威尔士州悉尼。

版权由所有者/作者所有。

ACM ISBN978-1-4503-9181-8/22/10.....15.00美元

<https://doi.org/10.1145/3495243.3558751>

边缘资源不可用性或网络基础设施的问题，以充分满足服务质量（QoS）要求的方式完成卸载任务可能是一个挑战[1, 5]。

作为一种替代解决方案，为了补充现有的CC和MEC范式，用户周围移动设备数量的增加，促使人们有机会共享彼此的分布式计算资源，而不需要连续连接或来自集中基础设施的监督。本地移动设备资源云的概念被称为“移动人群计算”（MCdC）[2, 6]，其中移动设备的集合及其用户协作地利用这些移动设备的计算资源。蜜蜂[6, 7]、牛至[12, 13]和意外发现[14]是MCdC框架的一些例子。Hoengerbee[6, 7]框架演示了使用蓝牙和Wi-Fi-Direct技术的MCdC的可行性，并采用了一种工作窃取算法来促进任务调度[4]。

这项工作建立在蜜蜂之前的工作的基础上，创建了一个名为蜜蜂-t的新扩展，它处理受工作窃取影响的工作之间的任务依赖关系，这是在以前的工作中没有考虑到的复杂性。在许多真实世界的用例中，可以预期任务之间的依赖关系，即特定任务的执行取决于另一个任务的结果。如果没有得到充分的管理，这种依赖关系可能会导致瓶颈，导致多个任务无法调度，直到处理数据可用。

这个演示展示了HoneybeeT的工作原型，促进了具有各种能力的移动设备来执行具有任务依赖性的协作计算。人类活动识别（HAR）任务是作为一个Android应用程序实现的，使用蜜蜂-t框架，并演示了在移动人群设置中的性能提高和节能。

## 2系统设计

为了形成一个成功的协作移动群体，该系统涉及至少两个移动设备。一种委托器设备，其中存在一个计算密集型任务和一个或多个工作设备，它们愿意贡献计算资源并执行部分计算密集型任务。这是通过将任务分解为委托者上更小的“作业”，这些作业被添加到一个“作业池”中，并允许工人/s从委托者那里“窃取”作业块。委托者还会同时执行作业池中的作业。授权者和工人都在完成时不断返回结果，并不断窃取更多的工作。该系统有两个主要组成部分：（1）作业池和（2）握手。

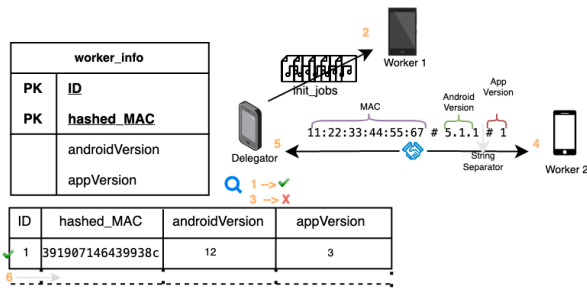


图1：蜜蜂T握手场景

JobPool 在委托器处保存一个作业集合。它使用双端列表进行填充，并且作业根据窃取请求者从列表的任意一端进行排队和退出队列。当进行窃取时，作业将从队列中删除，并维护有关被窃取作业的信息，并且窃取作业将由委托器和工作设备同时完成。我们修改了Job对象，并对蜜蜂的工作窃取算法进行了修改，从而实现了基于依赖关系的顺序任务窃取。在作业对象中，作业数据结构包含“阶段”属性，以表示数据处理的当前阶段。这将被用作窃取条件，这意味着如果一个设备不能执行第一阶段的作业，那么该作业将不被允许窃取，而是一个更有能力的作业将被释放。

握手：帮助根据设备的能力来调节任务窃取策略。参与设备的元数据必须在协作会话开始前收集。设备的元数据存储在SQLLite数据库中，用于查找设备信息，然后再决定是否启动握手。图1展示了握手机制的概述。该图表示：当委托设备与工作1建立套接字连接时，在步骤1中，委托设备将检查其数据库（DB）工作信息表，看看是否有工作1应用程序和Android版本的记录。由于记录存在，委托者没有发起握手，而是在第2步中共享初始作业块。关于工作2与委托者成功的套接字连接，在步骤3中，委托者在共享初始作业块之前无法在表工作信息中找到条目。在步骤4中，委托者发起一个握手请求，并且工作者2将通过共享其元数据来响应该握手，这是步骤5。为了避免将来的握手，在步骤6中，委托者将更新工作信息表。

### 3演示

这个演示模拟了这样一个场景，即移动应用程序需要外部资源来提供准确和及时的结果，而在具有移动性和基础设施连接有限的环境中。让我们假设一名乘客（约翰）在火车上，与朋友一起旅行，由于健康状况，需要持续监测身体运动。约翰热衷于利用他的朋友“移动设备”的集体“移动人群”来获得及时的结果，同时也保存他的手机电池，而不是使用

CC和/或MEC服务。为了模拟这个场景，演示使用了一个由加速度计、陀螺仪和磁力计数据组成的数据集，该数据集需要处理阶段和机器学习（ML）模型来进行活动识别，需要有能力的处理器。图2显示了HAR任务的常规执行流程。第一阶段是数据预处理阶段；数据预处理阶段的目的是消除与活动数据一起记录的任何噪声信号。在我们的实验中，“巴特沃斯低通滤波器”被用于进行滤波。第二阶段是特征提取，其中对预处理后的数据进行“快速傅里叶变换”。在最后阶段，使用“张流”库实现一个LSTM模型来执行识别活动。这三个任务必须按顺序执行，因此具有依赖性。

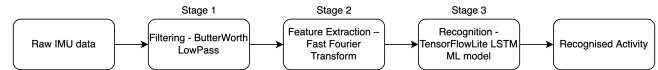


图2：在HAR中涉及的执行步骤的概述

蜜蜂-t目前支持三个安卓版本。1) 版本“1”，没有阶段、逻辑，适用于安卓版本6及更低版本的设备。2) 版本“2”有所有阶段，适用于7到11之间的安卓版本的设备。最后，3) 版本“3”也有所有的阶段，适用于安卓版本12或更大版本的设备。表1中列出的设备

表1：实验中使用的设备及其作用

身份证	设备	应用程序版本	机器人版本	实验角色
D1	OnePlus 5T	2	10	代表/工人
D2	像素6 Pro	3	12	工作者
D3	像素4	2	11	代表/工人
D4	像素3	2	11	工作者
D5	莫托G	1	5.1.1	工作者

在安卓版本中的设备角色以及在可计算性方面都是不同的。除了谷歌Pixel 6 Pro和Pixel 3设备外，所有其他设备都符合第2版的蜜蜂-t应用程序。谷歌Pixel 6 Pro符合App版本3，MotoG设备符合App版本1。在除单片执行外的所有实验中，Pixel 4或一加5T设备都是委托器，其他设备都是工作人员。图3显示了协同计算任务后获得的性能和能量收益。该演示还将显示一个案例，其中系统处理随机离开和连接的协作设备，正如在模拟列车场景中可以预期的那样。

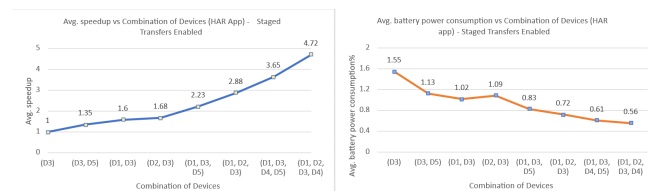


图3：HAR应用设备组合的平均加速和功耗的性能结果

## 参考文献

- [1] 2019. 版权所有。在数字双驱动智能制造中, 飞涛、张孟和A. Y. C. Nee (Eds)。学术出版社, iv。https://doi.org/10.1016/B978-0-12-817630-6.00014-X
- [2] 肯尼斯·李明昂, 茉莉花王, 和杜鹃花。2022. 面向众包物联网(人群物联网): 架构、安全和应用程序。未来的互联网公司14日、2日(2022年)、49年。
- [3] 的迈克尔·阿姆布鲁斯特、阿曼多·福克斯、雷恩·格里菲斯、安东尼·约瑟夫、兰迪·卡茨、安迪·康温斯基、古纳霍·李、大卫·帕特森、阿里尔·拉金、离子·斯托伊卡等, 2010年。云计算的一个视图。通勤。ACM 53, 4 (2010), 50-58。
- [4] 罗伯特·布鲁莫夫和查尔斯·雷瑟森。1999. 通过工作窃取来调度多线程计算。ACM学报(JACM) 46, 5 (1999), 720-748。
- [5] Flavio博诺米, 鲁道夫米利托, 姜朱, 和总督。2012. 雾化计算及其在物联网中的作用。在MCC移动云计算研讨会第一版的论文集中。13-16。
- [6] 尼罗希尼费尔南多, 森洛克和温尼拉哈尤。2012. 蜜蜂: 一个用于移动人群计算的编程框架。在移动和无处不在的系统国际会议上: 计算、网络和服务。施普林格, 224-236。
- [7] 尼罗希尼费尔南多, 森洛克和温尼拉哈尤。2016. 使用附近的移动设备进行计算: 一种用于移动边缘云的工作共享算法。IEEE《云计算交易》7, 2 (2016), 329-343。
- [8] 云超涛, 米兰帕特尔, 达里奥萨贝拉, 努里特普雷切, 和瓦莱丽杨。2015. 移动边缘计算——面向5G的关键技术。ETSI白皮书11、11 (2015), 1-16。
- 李[9]熙英, 安熙中, 阮俊, 崔淑淑, 金大进。2017. 比较大学生的自我报告和测量的智能手机使用情况: 一项初步研究。精神病学调查14, 2 (2017), 198。
- 刘方明、鹏树、海都、丁林杰、于洁、地牛、波波。2013. 将资源贫乏的移动设备与强大的云计算相结合: 架构、挑战 and 应用程序。IEEE无线通信公司20, 3 (2013), 14-22。
- [11] 克里斯蒂安·蒙塔格、康拉德·Błaszkiwicz、雷纳·萨里斯卡、伯恩德·拉克曼、伊奥努特·安多恩、鲍里斯·特伦达菲洛夫、马克·埃比斯和亚历山大·马尔科维茨。2015. 21世纪智能手机使用: 谁在WhatsApp上活跃? BMC研究注释8, 1 (2015), 1-6。
- [12] 佩德罗·桑奇斯, 乔·席尔瓦, 安东尼奥·特奥菲洛和赫维·保利诺。2020. 移动设备网络上的数据中心型分布式计算。在欧洲并行处理会议上。施普林格, 296-311。
- [13] 佩德罗·米格尔·卡斯坦海拉·桑切斯。2017. 在移动电话云中的分布式计算。博士论文。
- [14] 聪石, 拉卡法病, 阿马尔和埃伦·泽古拉。2012. 意外发现: 允许在间歇性连接的移动设备之间进行远程计算。在第十三届ACM移动特别网络和计算国际研讨会的论文集上。145-154。
- [15] 魏松、曹洁、张泉、李友子、徐兰宇。2016. 边缘计算: 视觉和挑战。IEEE物联网杂志3, 5 (2016), 637-646。
- [16] 托马斯DW威尔克森, 大卫埃利斯, 和希瑟肖。2018. 确定典型的智能手机使用情况: 我们需要什么数据? 网络心理学、行为和社交网络21, 6 (2018), 395-398。