

An Integrated Top-down and Bottom-up Task Allocation Approach in Social Sensing based Edge Computing Systems

Daniel (Yue) Zhang, Dong Wang

问题的出现:

边缘计算中的现有任务分配方案可分为两大类: 自上而下(集中式)方法和自下而上(分散式)方法。**基于自上而下的方法:** 中央决策者具有系统中任务和计算资源的完全控制和信息。它根据运行时状态将任务分配给边缘设备, 以满足应用程序的性能要求。**在基于自底向上的方法:** 私有边缘设备以分布式方式做出自己的任务分配决策, 而无需中央决策者。

现无法解决 SSEC 中的任务分配问题:

非对称信息挑战: 边缘设备或应用程序都没有关于整个 SSEC 系统的完整信息

竞争目标挑战: SSEC 应用程序和边缘设备可能具有不一致甚至竞争的目标

不同的所有权挑战: 计算和通信资源通常由不同的实体拥有

解决方案:

本文开发了一种自顶向下和自下而上相结合(TDBU)的方法。为了满足边缘设备的目标, 在 TDBU 中开发了一种自下而上的博弈论方法, 该方法允许边缘设备以最大化其收益的方式指定其首选任务。为了满足该应用程序的目标, 我们使用随机控制理论和在线学习在 TDBU 中开发了一种自上而下的控制机制, 以调节任务分配过程, 从而确保应用程序的 QoS 性能。

模型:

A. System Model



Figure 1. Social Sensing based Edge Computing System

B. Task Model

任务是周期性初始化的, 并且具有相同的周期和截止时间 (Ω)。

边缘设备的计算任务将原始传感器的测量结果转换为所需的中间输出。任务的计算结果将发送回边缘服务器, 以进行进一步的数据处理和分析。

$\text{Task} = \{\tau_1, \tau_2, \dots, \tau_Z\}$, $\tau_z = \{VI_z, VO_z, c_{z,x}\}$ 。 VI_z 是任务 τ_z 要处理的数据量, VO_z 是输出的大小。如果将 τ_z 分配给边缘设备 E_x , 则 $c_{z,x}$ 是估计的最坏情况执行时间 (WCET)。

C. Cost and Incentive Model

Edge Cost (π_x): 边缘设备 ED_x 完成计算任务所需支付的能源成本

在边缘设备 E_x 上执行 τ_z 所消耗的能量: $e_{z,x} = Power_x \times c_{z,x}$,

$Power_x$ 是边缘设备 E_x 的平均功耗 $Power_x = Power_{comp,x} + Power_{trans,x}$

Task Reward r_z : 应用程序提供的用于补偿边缘设备完成每个社交感知任务的奖励措施
固定预算表示为 ηC

边缘设备 $ED = \{E_1, E_2, \dots, E_X\}$: 在靠近边缘的位置执行感知和计算任务

应用程序服务器 A: 为社交感知应用程序感兴趣的所有用户提供了全局服务接口

本地边缘服务器 $ES = \{ES_1, ES_2, \dots, ES_Y\}$: 在靠近边缘的位置提供了额外的数据存储和计算

AS 和 ES 均由应用程序管理器拥有

Edge Payoff $u_{\{z, x\}}$: 边缘设备 E_x 通过执行任务 τ_z 所获得的总体收益(执行任务的成本和报酬)

TDBU 的目标:

服务器目标: 从服务器的角度来看, 目标是满足预定的 QoS 要求

$$\text{minimize: } \sum_{z=1}^Z \delta_z, \quad \text{s.t. } \sum_{z=1}^Z r_z \leq \eta_C$$

边缘目标: 从边缘设备的角度来看, 目标是最大程度地提高收益。

$$\begin{aligned} \text{maximize: } & \sum_{z \in S_x} u_{z,x}, \forall 1 \leq x \leq X \\ \text{s.t. } & \text{dist}_{x,z} \leq \eta_{D_x}, \forall 1 \leq x \leq X, \forall 1 \leq z \leq Z \end{aligned}$$

算法:

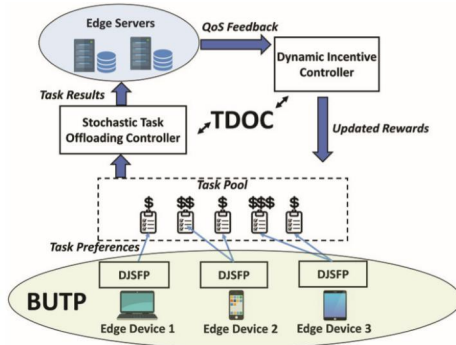


Figure 2. Overview of TDBU

TDBU 方案包含两个主要组件:

- 1) 博弈论的自下而上任务偏好 (BUTP) 模块, 该模块允许边缘设备根据自己的收益选择自己的首选任务。
- 2) 自上而下的最佳控制 (TDOC) 模块, 可确保满足服务器和边缘的目标。

A. Game-Theoretic Bottom-Up Task Preference Module

主要思想: 允许边缘设备在分配过程中表达其任务偏好

- 1) 允许边缘设备正确识别最大化其自身收益的策略
- 2) 从服务器隐藏设备状态的详细信息, 并有效地保护最终用户的隐私

具体实现:

- 1) 在检测周期的开始, 应用程序定义 Z 个任务和每个任务 τ_z 的奖励 r_z 。
- 2) 每个边缘设备都会自动选择一项收益最高的任务。我们将此任务分配过程的一轮称为“迭代”。
- 3) 在每次迭代中, 我们假设每个边缘设备都是近视设备, 并且一次仅选择一个任务。
- 4) 保持迭代, 直到所有边缘设备达成共识。
- 5) 每个设备将执行的任务的输出发送到边缘服务器以请求奖励。

收益函数取决于三个因素: 1) 处理选定任务的成本 2) 与任务完成相关的奖励 3) 服务器应用的惩罚函数以惩罚延迟的计算结果。

$$\pi_{z,x} = \begin{cases} e_{z,x}, & \text{dist}_{x,z} \leq \eta_{D_x} \\ \infty, & \text{dist}_{x,z} > \eta_{D_x} \end{cases}$$

奖励函数 (以激励最终用户在边缘设备上贡献其资源):

$$l_{z,x,y} = \begin{cases} \frac{\lambda \times \Omega}{\Omega - (td(x,y) + c_{z,x})}, & td(x,y) + c_{z,x} < \Omega \\ \infty, & td(x,y) + c_{z,x} \geq \Omega \end{cases}$$

基于报酬, 惩罚和成本, 我们将完成任务 τ_z 的边缘设备 E_x 的支付函数定义为:

$$u_{z,x} = \begin{cases} \text{Exp}\left(\frac{r_z}{\pi_x \times l_{z,x,y'}}\right) = \frac{r_z \times (\Omega - c_{z,x} - td(x, y'))}{\lambda \times \Omega \times N(z) \times e_{z,x}} \\ 0, & \pi_{z,x} = \infty \end{cases}$$

B. Top-Down Optimal Control Module

自上而下控制的原因是服务器无法依靠最新的边缘设备来执行最佳任务分配以满足应用程序的 QoS 要求。另一方面，边缘设备的收益功能可能与服务器目标不太吻合。

包含两种控制机制：

- 1) 基于 Lyapunov 优化的起泡沫控制器的随机任务，以在最大化 QoS 的同时明确考虑边缘设备的收益；
- 2) 基于在线学习的动态激励控制器，可使用反馈控制来调整每个边缘设备的收益，以解决系统动态问题（例如，截止日期未命中率，设备状态）。

第一种控制机制—通过 Lyapunov 优化进行随机任务填充控制器：

一旦边缘设备做出了任务分配决策并执行了选定的任务，任务的结果就会实时发送到本地边缘服务器。

$$\begin{aligned} \text{Minimize : } & \lim_{t \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T E(p_{edge}(t)); \\ \text{Subject to : } & \lim_{t \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T E(d_{z,y}(t)) \leq \Omega; \end{aligned}$$

第二种控制机制—通过在线学习的动态激励控制器：

通过在线学习框架动态更新奖励惩罚功能，从而满足应用程序的 QoS 目标。将控制损失函数 ϕ 设置为满足截止日期的任务的平均复杂度

$$\phi = \frac{\sum_{z=1}^N VI_z \times (1 - \delta_z)}{\sum_{z=1}^Z (1 - \delta_z)} - \frac{\sum_{z=1}^Z VI_z \times \delta_z}{\sum_{z=1}^Z \delta_z}, \sum_{z=1}^Z \delta_z \neq 0 \text{ or } Z \quad (14)$$

实验：

A. Hardware Platform

具有 Intel E5-2600 V4 处理器和 16GB DDR4 内存的三台 PC 工作站作为本地边缘服务器。

B. Energy Profiling

使用 FLUKE AC / DC 电流钳来监视每个边缘设备的实时电流信号，并使用 National Instruments USB-6216 数据采集 (DAQ) 系统捕获电流值。

- 1) Quality of Service (Application (Server) Side)
- 2) Payoff (Edge Device Side)
- 3). Allocation Overhead and Convergence