



(12) 发明专利申请

(10) 申请公布号 CN 113590229 A

(43) 申请公布日 2021.11.02

(21) 申请号 202110923267.8

(22) 申请日 2021.08.12

(71) 申请人 中山大学

地址 510275 广东省广州市海珠区新港西路135号

(72) 发明人 韩瑜 李锦铭 古博 秦臻 张旭 姜善成 唐兆家

(74) 专利代理机构 深圳市创富知识产权代理有限公司 44367

代理人 高冰

(51) Int. Cl.

G06F 9/445 (2018.01)

G06F 9/48 (2006.01)

G06F 9/50 (2006.01)

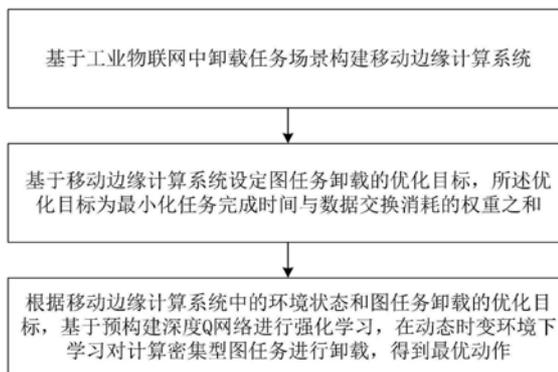
权利要求书2页 说明书8页 附图2页

(54) 发明名称

一种基于深度强化学习的工业物联网图任务卸载方法及系统

(57) 摘要

本发明公开了一种基于深度强化学习的工业物联网图任务卸载方法及系统,该方法包括:基于工业物联网中卸载任务场景构建移动边缘计算系统;基于移动边缘计算系统设定图任务卸载的优化目标,所述优化目标为最小化任务完成时间与数据交换消耗的权重之和;根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作。该系统包括:系统模型构建模块、优化目标设定模块和强化学习模块。通过使用本发明,能够制定在时变条件和有限资源下图形任务的卸载策略。本发明可广泛应用工业物联网领域。



1. 一种基于深度强化学习的工业物联网图任务卸载方法,其特征在于,包括以下步骤:

S1、基于工业物联网中卸载任务场景构建移动边缘计算系统;

S2、基于移动边缘计算系统设定图任务卸载的优化目标,所述优化目标为最小化任务完成时间与数据交换消耗的权重之和;

S3、根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作;

所述预构建深度Q网络包括结构相同的train Q-网络和target Q-网络。

2. 根据权利要求1所述一种基于深度强化学习的工业物联网图任务卸载方法,其特征在于,所述基于工业物联网中卸载任务场景构建移动边缘计算系统这一步骤,其具体包括:

S11、基于工业物联网中卸载任务场景构建移动边缘计算系统,所述卸载任务场景中包括一个任务发起者和多个任务执行者;

S12、将任务之间的依附关系用无向非循环图 $G = \{V, E\}$ 表示,其中包含一组任务 $V = \{V_i | i \in W\}$,以及一组边 $E = \{e_{ij} | (i, j) \in w, i \neq j\}$,其中W表示任务总数,G中的每个边 e_{ij} 用作二进制指示符变量指示 v_i 和 v_j 之间是否存在数据交换;

对移动边缘计算系统中进行图任务卸载,所述图任务卸载存在传输时间消耗、执行时间消耗和数据交换消耗。

3. 根据权利要求2所述一种基于深度强化学习的工业物联网图任务卸载方法,其特征在于,所述基于预构建深度Q网络进行强化学习的具体步骤包括:

状态空间,在t时刻的状态表示为 $s_t = \{h_t^d, f_t, u_t, G_t, d_t\}$,其中 $h_t^d = \{h_{t,i}^d | i \in m\}$ 代表着t时刻任务执行者i的通道收益, $f_t = \{f_{t,i} | i \in m\}$ 代表着t时刻任务执行者i的cpu频率, $u_t = \{u_{t,i} | i \in m\}$ 代表着t时刻任务执行者i的空闲间隙数量, G_t 代表任务图的拓扑关系, $d_t = \{d_{t,i} | i \in m\}$ 代表着t时刻任务执行者i与任务发起者的距离;

动作空间,当前任务 v_i 的动作表示为 $a_i = \{a_{i,1}, a_{i,2}, \dots, a_{i,m} | i \in m\}$,其中, $a_{i,j}$ 被设置为二进制指示符;

回报函数,系统的奖励设置为 $r = \frac{1}{\alpha T(u) + (1-\alpha)E(b)}$,其中, $T(u)$ 表示时间消耗, $E(b)$ 表示数据交换消耗, α 和 $(1-\alpha)$ 分别表示时间消耗和数据交换消耗的权重。

4. 根据权利要求3所述一种基于深度强化学习的工业物联网图任务卸载方法,其特征在于,所述根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作这一步骤,其具体包括:

S31、根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,在动态的环境下计算产生的时间消耗与数据交换消耗;

S32、确定动作a对应的回报r,将新的观测的环境状态 s' 输入到预构建深度Q网络中,并利用回报r来计算损失函数loss,进而通过反向梯度传递的方式对train Q-网络的参数进行更新;

S33、重复执行步骤S32,直至判断到回报r收敛并趋近最大化,则当前的动作为最优动作。

5. 根据权利要求4所述一种基于深度强化学习的工业物联网图任务卸载方法,其特征

在于,所述确定动作 a 对应的回报 r ,将新的观测的环境状态 s' 输入到预构建深度Q网络中,并利用回报 r 来计算损失函数 $loss$,进而通过反向梯度传递的方式对train Q-网络的参数进行更新这一步骤,其具体包括:

初始化图任务 $G = \{V, E\}$ 、经验回放池、train-Q网络的参数 θ^{train} 、target-Q网络的参数 θ^{target} 和系统环境,并设置空队列 Q ;

任务发起者随机选择一个任务 v_i ,将其作为队列的头结点并进行入队操作;

任务发起者依次从队列中取出待卸载任务 v_i ,并根据图任务 G 中的边集合 E 依次将与任务 v_i 相关联的所有未被卸载的任务进行入队操作;

将任务发起者当前所观测环境的状态输入到target Q-网络中,并输出 $q\{s, a | \theta\}_{a \in A}$,根据 ϵ -贪婪算法选出动作 a_i ,而后进行卸载任务;

任务发起者从环境中观测状态 s_{i+1} 和奖励回报 r_i 并将 (s_i, a_i, r_i, s_{i+1}) 作为经验值存取到经验回放池中;

判断到经验回放池存满数据,从其中随机选取 K 个经验值;

计算目标值并结合经验值,基于梯度下降方法更新train-Q网络的参数 θ^{train} ;

经过 F 时间步数后,根据当前时刻train Q-网络的参数 θ^{train} 更新target Q-网络的参数 θ^{target} 。

6. 一种基于深度强化学习的工业物联网图任务卸载系统,其特征在于,包括:

系统模型构建模块,用于基于工业物联网中卸载任务场景构建移动边缘计算系统;

优化目标设定模块,基于移动边缘计算系统设定图任务卸载的优化目标,所述优化目标为最小化任务完成时间与数据交换消耗的权重之和;

强化学习模块,用于根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作。

一种基于深度强化学习的工业物联网图任务卸载方法及系统

技术领域

[0001] 本发明涉及工业物联网领域,尤其涉及一种基于深度强化学习的工业物联网图任务卸载方法及系统。

背景技术

[0002] 在中国智能制造的大背景下,工厂的各环节正逐步向智能化发展,无人叉车作为智能厂内物流的主要实现方式,被广泛应用与各类工厂。它们能够自动地在工厂中高效地运送物料,解决了人工搬运劳动强度过高的问题。无人叉车(UF)具有计算处理器和各种传感设备(如无人机摄像头和高质量传感器),可以承载具有创新和计算密集型特性的感知相关应用(如人员识别、障碍识别、异常识别与预警等)。

[0003] 一些感知应用会包含大量的复杂任务,由于单个UF计算能力和能耗的限制,在UF本地进行一些复杂任务的处理是不现实的,因此需要将任务卸载到临近的设备或基站处理。由于复杂任务通常由相互依赖的子任务组成,这将使任务卸载复杂化,并使合理的任务卸载成为挑战。图任务是用来表示各种计算密集型任务之间的依赖关系,其中任务和数据流分别由图的顶点和边表示,因此根据任务图进行任务的卸载是一种有效的方法。

[0004] 对于通信条件复杂的工业物联网来说,由于无线信道条件频繁变化以及每个设备的计算资源存在波动,现有的图形任务卸载方法不能很好地执行。在高度动态的环境下,必须频繁求解优化问题,这可能会导致计算资源的浪费。

发明内容

[0005] 为了解决上述技术问题,本发明的目的是提供一种基于深度强化学习的工业物联网图任务卸载方法及系统,考虑到通信环境的复杂性,最小化任务完成时间与数据交换消耗的权重(WETC)之和,制定在时变条件和有限资源下图形任务的卸载策略。

[0006] 本发明所采用的第一技术方案是:一种基于深度强化学习的工业物联网图任务卸载方法,包括以下步骤:

[0007] S1、基于工业物联网中卸载任务场景构建移动边缘计算系统;

[0008] S2、基于移动边缘计算系统设定图任务卸载的优化目标,所述优化目标为最小化任务完成时间与数据交换消耗的权重之和;

[0009] S3、根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作;

[0010] 所述预构建深度Q网络包括结构相同的train Q-网络和target Q-网络。

[0011] 进一步,所述基于工业物联网中一个任务发起者和多个任务执行者卸载任务场景构建移动边缘计算系统这一步骤,其具体包括:

[0012] S11、基于工业物联网中卸载任务场景构建移动边缘计算系统,所述卸载任务场景中包括一个任务发起者和多个任务执行者;

[0013] S12、将任务之间的依附关系用无向非循环图 $G = \{V, E\}$ 表示,其中包含一组任务 $V = \{V_i | i \in W\}$,以及一组边 $E = \{e_{ij} | (i, j) \in w, i \neq j\}$,其中 W 表示任务总数, G 中的每个边 e_{ij} 用作二进制指示符变量指示 v_i 和 v_j 之间是否存在数据交换;

[0014] 对移动边缘计算系统中进行图任务卸载,所述图任务卸载存在传输时间消耗、执行时间消耗和数据交换消耗。

[0015] 进一步,所述预构建深度Q网络的强化学习如下:

[0016] 状态空间,在 t 时刻的状态表示为 $s_t = \{h_t^d, f_t, u_t, G_t, d_t\}$,其中 $h_t^d = \{h_{t,i}^d | i \in m\}$ 代表着 t 时刻任务执行者 i 的通道收益, $f_t = \{f_{t,i} | i \in m\}$ 代表着 t 时刻任务执行者 i 的cpu频率, $u_t = \{u_{t,i} | i \in m\}$ 代表着 t 时刻任务执行者 i 的idle gaps数量, G_t 代表任务图的拓补关系, $d_t = \{d_{t,i} | i \in m\}$ 代表着 t 时刻任务执行者 i 与任务发起者的距离;

[0017] 动作空间,当前任务 v_i 的动作表示为 $a_i \{a_{i,1}, a_{i,2}, \dots, a_{i,m} | i \in m\}$,其中, $a_{i,j}$ 被设置为二进制指示符,即 $a_{i,j} = 1$ 表示为选择设备 n_j 来卸载任务 v_i , $a_{i,j} = 0$ 表示为不选择设备 n_j 来卸载任务 v_i ;

[0018] 回报函数,系统的奖励设置为 $r = \frac{1}{\alpha T(u) + (1-\alpha)E(b)}$,其中, $T(u)$ 表示时间消耗, $E(b)$ 表示数据交换消耗, α 和 $(1-\alpha)$ 分别表示时间消耗和数据交换消耗的权重。

[0019] 进一步,所述根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作这一步骤,其具体包括:

[0020] S31、根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,在动态的环境下计算产生的时间消耗与数据交换消耗;

[0021] S32、确定动作 a 对应的回报 r ,将新的观测的环境状态 s' 输入到预构建深度Q网络中,并利用回报 r 来计算损失函数loss,进而通过反向梯度传递的方式对train Q-网络的参数进行更新;

[0022] S33、重复执行步骤S32,直至判断到回报 r 收敛并趋近最大化,则当前的动作为最优动作。

[0023] 进一步,所述确定动作 a 对应的回报 r ,将新的观测的环境状态 s' 输入到预构建深度Q网络中,并利用回报 r 来计算损失函数loss,进而通过反向梯度传递的方式对train Q-网络的参数进行更新这一步骤,其具体包括:

[0024] 初始化图任务 $G = \{V, E\}$ 、经验回放池、train-Q网络的参数 θ^{train} 、target-Q网络的参数 θ^{target} 和系统环境,并设置空队列 Q ;

[0025] 任务发起者随机选择一个任务 v_i ,将其作为队列的头结点并进行入队操作;

[0026] 任务发起者依次从队列中取出待卸载任务 v_i ,并根据图任务 G 中的边集合 E 依次将与任务 v_i 相关联的所有未被卸载的任务进行入队操作;

[0027] 将任务发起者当前所观测环境的的状态输入到target Q-网络中,并输出 $q\{s, a | \theta\}_{a \in A}$,根据 ϵ -贪婪算法选出动作 a_i ,而后进行卸载任务;

[0028] 任务发起者从环境中观测状态 s_{i+1} 和奖励回报 i 并将 (s_i, a_i, r_i, s_{i+1}) 作为经验值存取到经验回放池中;

[0029] 判断到经验回放池存满数据,从其中随机选取 K 个经验值;

- [0030] 计算目标值并结合经验值,基于梯度下降方法更新train-Q网络的参数 θ^{train} ;
- [0031] 经过F时间步数后,根据当前时刻train Q-网络的参数 θ^{train} 更新target Q-网络的参数 θ^{target} 。
- [0032] 本发明所采用的第二技术方案是:一种基于深度强化学习的工业物联网图任务卸载系统,包括:
- [0033] 系统模型构建模块,用于基于工业物联网中一个任务发起者和多个任务执行者卸载任务场景构建移动边缘计算系统;
- [0034] 优化目标设定模块,基于移动边缘计算系统设定图任务卸载的优化目标,所述优化目标为最小化任务完成时间与数据交换消耗的权重之和;
- [0035] 强化学习模块,用于根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作。
- [0036] 本发明方法及系统的有益效果是:本发明考虑到通信环境的复杂性,最小化WETC之和,制定在时变条件和有限资源下图形任务的卸载策略,通过深度Q网络在多次迭代中不断积累经验,并通过反馈信号进行学习,从而能够在动态环境中,制定接近最优的策略来合理地卸载复杂的图形任务。

附图说明

- [0037] 图1是本发明一种基于深度强化学习的工业物联网图任务卸载方法的步骤流程图;
- [0038] 图2是本发明一种基于深度强化学习的工业物联网图任务卸载系统的结构框图;
- [0039] 图3是本发明具体实施例工业物联网场景的示意图;
- [0040] 图4是本发明具体实施例图任务框架的示意图。

具体实施方式

- [0041] 下面结合附图和具体实施例对本发明做进一步的详细说明。对于以下实施例中的步骤编号,其仅为了便于阐述说明而设置,对步骤之间的顺序不做任何限定,实施例中的各步骤的执行顺序均可根据本领域技术人员的理解来进行适应性调整。
- [0042] 参照图1,本发明提供了一种基于深度强化学习的工业物联网图任务卸载方法,该方法包括以下步骤:
- [0043] S1、基于工业物联网中卸载任务场景构建移动边缘计算系统;
- [0044] 具体地,参照图3,由于单个UF计算能力和能耗的限制,该设备需要通过无线通信将本地产生的计算密集型任务卸载到附近的设备和服务器上进行处理,从而降低单台设备的容量需求和能耗,因此,我们将其中的一个任务发起者和多个任务执行者卸载任务的场景构成了一个移动边缘计算(MEC)系统。在本文中,每个MEC系统作为一个并行执行图形任务的单元,在移动网络边缘提供了信息技术服务环境和云计算能力,使得快速处理任务成为可能,并且假设每个MEC系统之间没有干扰。
- [0045] 假设MEC系统中存在一个任务发起者和M个任务执行者,其中 $N = \{N_i | i \in M\}$ 表示任务执行者的编号,每个任务执行者的计算资源被划分为不同数量的空闲间隙,可以用 $z =$

$\{z_i | i \in M\}$ 来描述。我们认为每个空闲间隙可以为图任务中的每个任务提供计算服务。

[0046] S2、基于移动边缘计算系统设定图任务卸载的优化目标,所述优化目标为最小化任务完成时间与数据交换消耗的权重之和;

[0047] S3、根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作;

[0048] 所述预构建深度Q网络包括结构相同的train Q-网络和target Q-网络。

[0049] 具体地,深度Q网络(DQN)是由两个结构相同,但作用不同的神经网络组成,一个是train Q-网络,一个是target Q-网络。两个神经网络各有不同的参数 θ^{train} 和 θ^{target} 。 θ^{train} 用于评估最优动作的Q值(期望回报), θ^{target} 用于选择对应最大Q值的动作(通过 ϵ -贪婪算法)。这两组参数将动作选择和策略评估分离,降低了估计Q值过程中的过拟合风险。我们使用经验池来存储所有智能体产生的经验,并将从经验池中随机采样得到的经验作为trainQ-网络的输入,来对其进行参数更新,这样不仅可以大大减少训练所需要的内存和计算资源,同时降低了数据之间的耦合性。

[0050] 进一步作为本方法的优选实施例,所述基于工业物联网中一个任务发起者和多个任务执行者卸载任务场景构建移动边缘计算系统这一步骤,其具体包括:

[0051] S11、基于工业物联网中卸载任务场景构建移动边缘计算系统,所述卸载任务场景中包括一个任务发起者和多个任务执行者;

[0052] S12、将任务之间的依附关系用无向非循环图 $G = \{V, E\}$ 表示,其中包含一组任务 $V = \{V_i | i \in W\}$,以及一组边 $E = \{e_{ij} | (i, j) \in w, i \neq j\}$,其中W表示任务总数,G中的每个边 e_{ij} 用作二进制指示符变量指示 v_i 和 v_j 之间是否存在数据交换;

[0053] 具体地,为了方便地反映图任务的不同拓扑关系,我们将任务之间的依附关系用无向非循环图 $G = \{V, E\}$ 表示,其中包含一组任务 $V = \{V_i | i \in W\}$,以及一组边 $E = \{e_{ij} | (i, j) \in w, i \neq j\}$,其中W表示任务总数,G中的每个边 e_{ij} 用作二进制指示符变量指示 v_i 和 v_j 之间是否存在数据交换。此外,参数 $L = \{L_i | i \in W\}$ 表示不同任务的数据大小,计算工作量由参数 $K = \{K_i | i \in W\}$ 表示,其中 K_i 表示执行任务 v_i 所需的CPU周期量。图任务框架参照图4。

[0054] 另外,我们假设在传输过程中为每个任务执行者分配一个专用的频谱资源块,以支持任务卸载和下载的并发传输。考虑到上行链路传输时间远长于下行链路传输时间,本文只讨论了上行链路传输所带来的消耗,并且假设由于任务发起者长期受到资源不足的困扰,因此会倾向于将计算密集型任务全部卸载到任务执行者。

[0055] 对移动边缘计算系统中进行图任务卸载,所述图任务卸载存在传输时间消耗、执行时间消耗和数据交换消耗。

[0056] 传输时间消耗 t_{ij}^{trans} ($1 \leq i \leq W, 1 \leq j \leq M$)表示任务发起者将任务i并行卸载到边缘设备j,其数值与信道条件、传输功率、带宽等条件有关。我们用 P_{TI} 表示从任务发起者上行传输的固定功率, h_{ij}^d 表示卸载任务i到设备j的通道收益。并且,我们假设所有任务的接收端均存在均值为零、方差 σ^2 相等的加性高斯白噪声。根据香农定理,从任务发起者将任务i卸载到设备j的上行传输率:

$$[0057] \quad R_{ij}^d = B_j \log_2 \left(1 + \frac{P_{T1} h_j^d}{\sigma^2} \right)$$

[0058] 其中, B 表示为分配给边缘设备的正交信道的固定带宽。所以, 任务 i 的上行传输时间为: $t_{ij}^d = \frac{L_i}{R_{ij}^d}$ 。

[0059] 我们用二进制指示变量 a_{ij} 表示任务 i 是否卸载到设备 j

$$[0060] \quad a_{ij} = \begin{cases} 1, & \text{任务 } i \text{ 卸载到设备 } j \\ 0, & \text{任务 } i \text{ 没有卸载到设备 } j \end{cases}$$

[0061] 因此, 任务传输消耗总时间为:

$$[0062] \quad T^{\text{trans}}(a) = \max \left(\sum_{i=1}^w \sum_{j=1}^m a_{ij} t_{ij}^d \right)$$

[0063] 执行时间消耗: 当任务传输到各个空闲间隙的之后, 边缘设备开始并行执行各子任务。我们用 $f = \{f_i | i \leq M\}$ 表示为边缘设备上执行任务所用的 CPU 频率, 并认为承担计算任务的空闲间隙采用同样的 f_i 。因此, 任务执行消耗总时间为:

$$[0064] \quad T^{\text{exec}}(a) = \max \left(\sum_{i=1}^w \sum_{j=1}^m \frac{a_{ij} K_i}{f_j} \right)$$

[0065] 所以, 任务传输和执行所消耗的总时间为:

$$[0066] \quad T(a) = T^{\text{trans}}(a) + T^{\text{exec}}(a)$$

[0067] 数据交换成本:

[0068] 我们假设当具有连接关系的子任务卸载到不同任务执行者的时候, 其会产生数据交换成本为 $c_{jj'}$ ($j \in m, j' \in m, i \neq j$), 若卸载到同一任务执行者中, 则认为不产生数据交换成本, 这就代表了在 MEC 系统中不同任务执行者之间流量交换而产生的成本。我们用二进制指示变量 $b_{jj'}$ 来表示不同任务执行者之间是否存在数据交换, 则

$$[0069] \quad b_{jj'} = \begin{cases} 1, & \text{当 } j \neq j', a_{ij} = 1, a_{ij'} = 1, e_{ii} = 1 \\ 0, & \text{otherwise} \end{cases}$$

[0070] 因此, 数据交换总消耗为:

$$[0071] \quad E(b) = \sum_{j=1}^m \sum_{j'=j+1}^m b_{jj'} c_{jj'}$$

[0072] 优化目标建模: 为了获得所考虑的 MEC 中对于计算密集型图任务的卸载策略, 我们以最小化任务完成时间与数据交换消耗的权重之和 (WETC) 为主要性能来制定下述的优化问题。

$$[0073] \quad \gamma = \alpha T(u) + (1 - \alpha) E(b)$$

[0074] 从而, 构建如下的优化模型:

$$[0075] \quad \min_{z, f, d} \tau$$

[0076] s. t. (a) $u_{ij} \in \{0, 1\} \quad \forall i \in w, \forall j \in m$

[0077] (b) $\sum_{i \in w} \sum_{j \in m} u_{ij} = w$

[0078] (c) $\sum_{i \in w} u_{ij} \leq z_j \quad \forall j \in m$

[0079] (d) $f_{min} \leq f_j \leq f_{max} \quad \forall j \in m$

[0080] (e) $d_{min} \leq d_j \leq d_{max} \quad \forall j \in m$

[0081] (f) $z_{min} \leq z_j \leq z_{max} \quad \forall j \in m$

[0082] 同时要满足以下六个限制条件：

[0083] (1) $u_{ij} \in \{0, 1\}, \forall i \in W, j \in M$, 表示任务 v_i 是否分配到设备 n_j ；

[0084] (2) $\sum_{i=1}^w \sum_{j=1}^m u_{ij} = W$, 表示任务图中所有的任务都要分配到相关任务执行者的空闲间隙上进行执行；

[0085] (3) $\sum_{i=1}^w u_{ij} \leq z_j, \forall j \in M$, 确保分配给同一任务执行者的任务不能超过最大计算资源

[0086] (4) $f_{min} \leq f_j \leq f_{max}, \forall j \in m$, 表示每个任务执行者的CPU频率大小要限制在一定区间；

[0087] (5) $d_{min} \leq d_j \leq d_{max}, \forall j \in m$, 表示对任务发起者和任务执行者之间的距离加以限制；

[0088] (6) $z_{min} \leq z_j \leq z_{max}, \forall j \in m$, 表示每个任务执行者的计算资源都在一定区间内随机波动。

[0089] 进一步作为本方法的优选实施例,我们采用了一种基于深度Q网络(DQN)和广度优先遍历(BFS)相结合的算法在动态时变的环境下学习对计算密集型图任务进行卸载。所述预构建深度Q网络的强化学习如下：

[0090] 状态空间,在我们所提出的DRL框架中,agent会在时间间隙对环境进行监控并记录系统状态。在 t 时刻,我们将此时的状态表示为 $s_t = \{h_t^d, f_t, u_t, G_t, d_t\}$, 其中

$h_t^d = \{h_{t,i}^d \mid i \in m\}$ 代表着 t 时刻任务执行者 i 的通道收益, $f_t = \{f_{t,i} \mid i \in m\}$ 代表着 t 时刻任务执行者 i 的cpu频率, $u_t = \{u_{t,i} \mid i \in m\}$ 代表着 t 时刻任务执行者 i 的idle gaps数量, G_t 代表任务图的拓补关系, $d_t = \{d_{t,i} \mid i \in m\}$ 代表着 t 时刻任务执行者 i 与任务发起者的距离；

[0091] 动作空间,在获取环境反馈状态时,代理将根据所观察的情况以及当前任务 v_i 与图任务中其他任务的连接关系,并根据Q网络的输出选择最适合的卸载策略。因此,当前任务 v_i 的动作可以表示为 $a_i = \{a_{i,1}, a_{i,2}, \dots, a_{i,m} \mid i \in m\}$, 其中, $a_{i,j}$ 被设置为二进制指示符,即 $a_{i,j} = 1$ 表示为选择设备 n_j 来卸载任务 v_i , $a_{i,j} = 0$ 表示为不选择设备 n_j 来卸载任务 v_i ；

[0092] 回报函数,上述考虑的优化目标是在面对随机变换的大规模场景下以实现图任务

卸载产生WETC的最小化,因此我们将系统的奖励设置为 $r = \frac{1}{\alpha T(u) + (1-\alpha)E(b)}$,其中, $T(u)$ 表示时间消耗, $E(b)$ 表示数据交换消耗, α 和 $(1-\alpha)$ 分别表示时间消耗和数据交换消耗的权重。通过最大化 r 来实现最小WETC的目的,最终使任务发起者任务发起者找到最佳的卸载方案。

[0093] 进一步作为本方法的优选实施例,所述根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作这一步骤,其具体包括:

[0094] S31、根据移动边缘计算系统中的环境状态和图任务卸载的优化目标,在动态的环境下计算产生的时间消耗与数据交换消耗;

[0095] S32、确定动作 a 对应的回报 r ,将新的观测的环境状态 s' 输入到预构建深度Q网络中,并利用回报 r 来计算损失函数 $loss$,进而通过反向梯度传递的方式对train Q-网络的参数进行更新;

[0096] S33、重复执行步骤S32,直至判断到回报 r 收敛并趋近最大化,则当前的动作为最优动作。

[0097] 进一步作为本方法优选实施例,所述确定动作 a 对应的回报 r ,将新的观测的环境状态 s' 输入到预构建深度Q网络中,并利用回报 r 来计算损失函数 $loss$,进而通过反向梯度传递的方式对train Q-网络的参数进行更新这一步骤,其具体包括:

[0098] 初始化图任务 $G = \{V, E\}$ 、经验回放池、train-Q网络的参数 θ^{train} 、target-Q网络的参数 θ^{target} 和系统环境,并设置空队列 Q ;

[0099] 任务发起者随机选择一个任务 v_i ,将其作为队列的头结点并进行入队操作;

[0100] 任务发起者依次从队列中取出待卸载任务 v_i ,并根据图任务 G 中的边集合 E 依次将与任务 v_i 相关联的所有未被卸载的任务进行入队操作;

[0101] 将任务发起者当前所观测环境的的状态输入到target Q-网络中,并输出 $q\{s, a | \theta\}_{a \in A}$,根据 ϵ -贪婪算法选出动作 a_i ,而后进行卸载任务;

[0102] 具体地, ϵ -贪婪算法:

[0103]
$$a = \begin{cases} \text{使得} Q \text{值最大的动作,} & 1-\epsilon \\ \text{随机动作,} & \epsilon \end{cases}$$

[0104] 任务发起者从环境中观测状态 s_{i+1} 和奖励回报 r_i 并将 (s_i, a_i, r_i, s_{i+1}) 作为经验值存取到经验回放池中;

[0105] 判断到经验回放池存满数据,从其中随机选取 K 个经验值;

[0106] 计算目标值并结合经验值,基于梯度下降方法更新train-Q网络的参数 θ^{train} ;

[0107] 经过 F 时间步数后,根据当前时刻train Q-网络的参数 θ^{train} 更新target Q-网络的参数 θ^{target} 。

[0108] 如图2所示,一种基于深度强化学习的工业物联网图任务卸载系统,包括:

[0109] 系统模型构建模块,用于基于工业物联网中一个任务发起者和多个任务执行者卸载任务场景构建移动边缘计算系统;

[0110] 优化目标设定模块,基于移动边缘计算系统设定图任务卸载的优化目标,所述优化目标为最小化任务完成时间与数据交换消耗的权重之和;

[0111] 强化学习模块,用于根据移动边缘计算系统中的环境状态和图任务卸载的优化目

标,基于预构建深度Q网络进行强化学习,在动态时变环境下学习对计算密集型图任务进行卸载,得到最优动作。

[0112] 上述方法实施例中的内容均适用于本系统实施例中,本系统实施例所具体实现的功能与上述方法实施例相同,并且达到的有益效果与上述方法实施例所达到的有益效果也相同。

[0113] 以上是对本发明的较佳实施进行了具体说明,但本发明创造并不限于所述实施例,熟悉本领域的技术人员在不违背本发明精神的前提下还可做作出种种的等同变形或替换,这些等同的变形或替换均包含在本申请权利要求所限定的范围内。

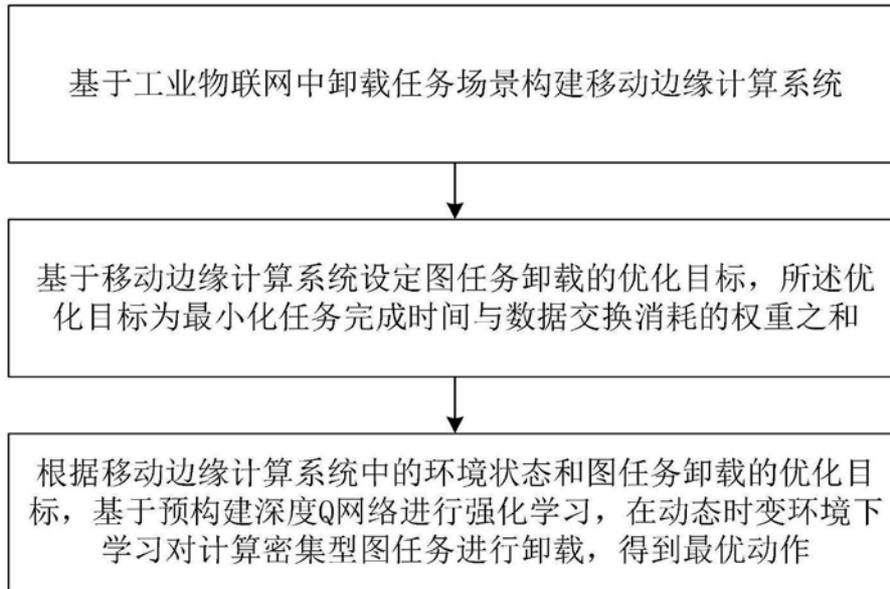


图1

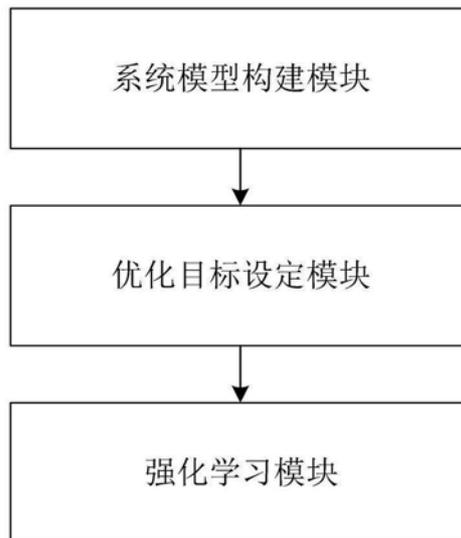


图2

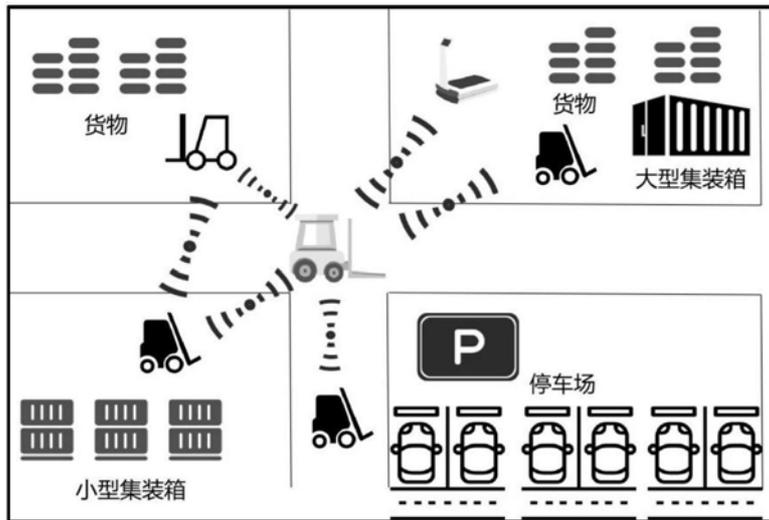


图3

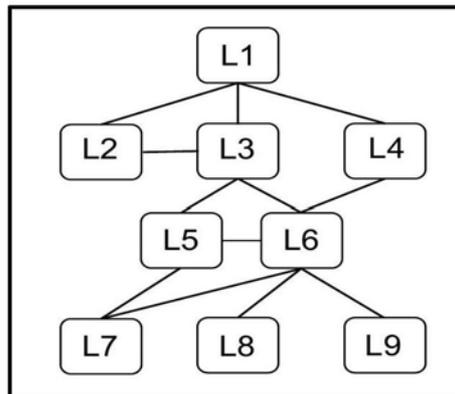


图4