(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2020 年 9 月 17 日 (17.09.2020)

WIPO | PCT

(10) 国际公布号
WO 2020/181896 A1

(51) 国际专利分类号:
G06F 9/50 (2006.01)

(21) 国际申请号: PCT/CN2019/130582

(22) 国际申请日: 2019 年 12 月 31 日 (31.12.2019)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201910193429.X 2019年3月14日 (14.03.2019) CN

(71) 申请人: 深 圳 先 进 技 术 研 究 院 (**SHENZHEN INSTITUTES OF ADVANCED TECHNOLOGY**) [CN/CN]; 中国广东省深圳市南山区西丽大学城学苑大道1068号, Guangdong 518055 (CN)。

(72) 发明人: 任 宏 帅 (**REN, Hongshuai**); 中国广东省深圳市南山区西丽大学城学苑大道1068号, Guangdong 518055 (CN)。 王 洋 (**WANG, Yang**); 中国广东省深圳市南山区西丽大学城学苑大道1068号, Guangdong 518055 (CN)。 须成忠(**XU, Chengzhong**); 中国广东省深圳市南山区西丽大学城学苑大道1068号, Guangdong 518055 (CN)。

(74) 代理人: 深圳市科进知识产权代理事务所（普通合伙） (**SHENZHEN KEJIN INTELLECTUAL PROPERTY OFFICE**); 中国广东省深圳市南山区粤兴三道二号虚拟大学园产业化基地A701室, Guangdong 518055 (CN)。

(81) 指定国(除另有指明，要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK,

(54) **Title:** MULTI-AGENT REINFORCEMENT LEARNING SCHEDULING METHOD AND SYSTEM AND ELECTRONIC DEVICE

(54) 发明名称: 一种多智能体强化学习调度方法、系统及电子设备

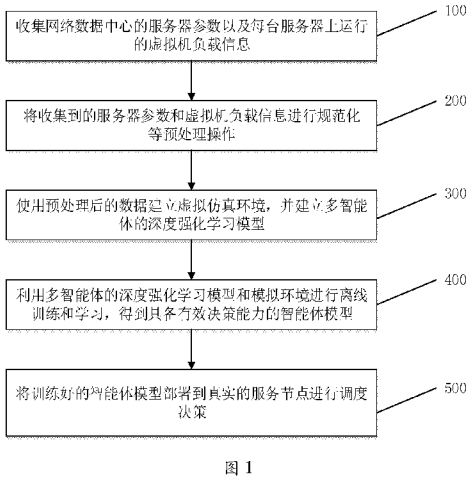

图 1

100 Collect server parameters of a network data center and load information of virtual machines running on each server
200 Perform normalization and other pretreatment operations on the collected server parameters and the load information of the virtual machines
300 Establish a virtual simulation environment by using the pretreated data and build a multi-agent deep reinforcement learning model
400 Perform offline training and learning by using the multi-agent deep reinforcement learning model and a simulation environment to obtain an agent model with effective decision capability
500 Deploy the trained agent model to a real service node for scheduling decision

(57) **Abstract:** Disclosed are a multi-agent reinforcement learning scheduling method and system and an electronic device. The method comprises: step a: collecting server parameters of a network data center and load information of virtual machines running on each server (100); step b: establishing a virtual simulation environment by using the server parameters and the load information of the virtual machines, and building a multi-agent deep reinforcement learning model; step c: performing offline training and learning by using the multi-agent deep reinforcement learning model, and training an agent model for each server; and step d: deploying the agent model to a real service node, and scheduling according to the load condition of each service node. The virtualization technology is used for virtualizing the services running on the server and the virtual machines are scheduled for load balancing, thereby achieving more macroscopic resource allocation and realizing the collaboration strategy of multi-agents in a complex dynamic environment.

WO 2020/181896 A1

[见续页]

LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(84) 指定国(除另有指明，要求每一种可提供的地区保护)：ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第21条(3))。

(57) 摘要：一种多智能体强化学习调度方法、系统及电子设备，所述方法包括：步骤a：收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息（100）；步骤b：使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建立多智能体的深度强化学习模型；步骤c：利用所述多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型；步骤d：将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。通过虚拟化技术将服务器上运行的服务虚拟化，通过调度虚拟机的方式来进行负载均衡，资源分配更加宏观，可以实现多智能体在复杂的动态环境下产生协作的策略。

一种多智能体强化学习调度方法、系统及电子设备

技术领域

本申请属于多智能体系统技术领域，特别涉及一种多智能体强化学习调度方法、系统及电子设备。

背景技术

在云计算环境下传统的服务部署方式很难应对多变的访问方式，资源的固定分配虽然能够稳定的提供服务，但同时这其中也存在的大量的资源浪费，例如在同一个网络拓扑结构下，有的服务器可能经常处于满负载运行，而有些服务器却只部署了几个服务仍然存在许多没有被使用的存储空间和运算能力，可见传统的部署服务难以应对这种资源的浪费，而且难以实现高效的调度，使得无法高效的利用资源。因此需要一种能够自适应动态环境的调度算法来平衡网络中个服务器的负载。

随着虚拟化技术的发展，虚拟机容器等技术的出现也将资源调度问题由静态分配推进到了动态分配的局面，近年来针对资源自适应调度的方案层出不穷，大多数都采用了启发式算法，通过调节参数的方式进行动态调度，并根据阈值调整运行环境的可用资源的充裕或紧张的情况，使用启发式算法迭代计算合适的阈值。但是这种调度方式只是在海量的数据组合上去寻求最优解，并且求解的最优决策只是针对当前特定时间节点，没有充分的利用时序信息，难以解决大型复杂的动态环境下的资源分配问题。

随着人工智能的兴起，深度强化学习技术的发展使得智能体在大状态空间上的决策成为了可能。在多智能体强化学习领域中，如果使用传统的

Q-learning、PG(Policy Gradient Method，策略梯度算法)等强化学习算法进行分布式学习仍然无法取得预期的效果，因为在每个步骤中每个智能体都尝试学习预测其他智能体的行动，而在动态环境下其他智能体总是在变化的，因此环境会变得不稳定难以学习到知识，无法实现最优化的资源分配。另外从强化学习的方法上来看，目前的调度手段大多都是单智能体强化学习与分布式强化学习，如果只用一个智能体集中式训练，会因为网络拓扑结构下复杂的状态变化与排列组合的大量动作空间使得算法难以训练，不易收敛。而使用分布式强化学习的办法也面临着另外一种问题，通常的分布式强化学习是通过多个智能体共同训练来加快收敛速度，但是事实上这些智能体的调度策略都是相同的，只是在训练的过程中用多个分身加快训练的速度，所以最后得到的都是同质化的智能体不具备协作能力。传统的多智能体方法中每个智能体会在每一步决策的时候去预测其他智能体的决策，但是因为在动态环境下其他智能体的决策也是不稳定的，训练十分困难而且每个智能体能做的事情几乎一样没有协作的策略。

发明内容

本申请提供了一种多智能体强化学习调度方法、系统及电子设备，旨在至少在一定程度上解决现有技术中的上述技术问题之一。

为了解决上述问题，本申请提供了如下技术方案：

一种多智能体强化学习调度方法，包括以下步骤：

步骤a：收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息；

步骤b：使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建

立多智能体的深度强化学习模型；

步骤 c：利用所述多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型；

步骤 d：将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

本申请实施例采取的技术方案还包括：所述步骤 a 还包括：将收集到的服务器参数和虚拟机负载信息进行规范化预处理操作；所述规范化预处理操作包括：定义每个服务节点虚拟机信息为一个多元组，所述多元组包括虚拟机的数量与其各自的配置，每个虚拟机包括两个调度状态，分别为待调度状态和运行状态，每个服务节点包括两个状态，分别为饱和状态和饥饿状态，各个虚拟机占用的资源比之和少于所在服务器配置的上限。

本申请实施例采取的技术方案还包括：在所述步骤 b 中，所述多智能体的深度强化学习模型具体包括预测模块和调度模块，所述预测模块通过各个服务节点输入的信息对当前状态下需要调度出去的资源进行预测，根据当前服务节点的配置信息将动作空间映射到当前服务节点的总容量之内；所述调度模块根据标记出来的待调度状态的虚拟机，进行重新调度分配产生调度策略，各个服务节点上的智能体根据产生的调度动作计算回报函数；所述预测模块度量调度策略的好坏，使整个网络中各个服务节点负载均衡。

本申请实施例采取的技术方案还包括：在所述步骤 c 中，所述利用多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型具体包括：每个服务节点上的智能体通过预测模块调整需要调度的资源大小，标记出需要调度出去的虚拟机，根据待调度状态的虚拟机产生调度策略，各个服务节点分别计算自身的回报值并汇总求和得到总回报值，并根据总回报值调整各个预测模块的参数。

本申请实施例采取的技术方案还包括：在所述步骤 d 中，所述将智能体模

型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度具体为：将训练好的智能体模型部署到真实环境中对应的服务节点上，所述智能体模型感知到所在服务器上的一段时间内的状态信息作为输入，预测得到当前服务器需要释放掉的资源，并使用背包算法选出最接近标准的虚拟机将其标记为待调度状态；之后通过调度模块收集到所有服务器上的预测结果与被标记为待调度状态的虚拟机，再按需将待调度状态的虚拟机指派给适合的服务器产生调度策略，将调度命令分发至对应服务节点执行调度操作；在执行调度策略之前对每个调度命令进行校验是否合法，若不合法则反馈一个惩罚奖励更新参数，重新产生调度策略；若合法则执行调度操作，并获得反馈的奖励值更新智能体参数。

本申请实施例采取的另一技术方案为：一种多智能体强化学习调度系统，包括：

信息收集模块：用于收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息；

强化学习模型构建模块：用于使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建立多智能体的深度强化学习模型；

智能体模型训练模块：用于利用所述多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型；

智能体部署模块：用于将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

本申请实施例采取的技术方案还包括预处理模块，所述预处理模块用于将收集到的服务器参数和虚拟机负载信息进行规范化预处理操作；所述规范化预处理操作包括：定义每个服务节点虚拟机信息为一个多元组，所述多元组包括虚拟机的数量与其各自的配置，每个虚拟机包括两个调度状态，分别为待调度状态和运行状态，每个服务节点包括两个状态，分别为饱和状态和饥饿状态，各个虚拟机占用的资源比之和少于所在服务器配置的上限。

本申请实施例采取的技术方案还包括：所述强化学习模型构建模块包括预测模块和调度模块，所述预测模块包括：

状态感知单元：用于通过各个服务节点输入的信息对当前状态下需要调度出去的资源进行预测；

动作空间单元：用于根据当前服务节点的配置信息将动作空间映射到当前服务节点的总容量之内；

所述调度模块根据标记出来的待调度状态的虚拟机，进行重新调度分配产生调度策略，各个服务节点上的智能体根据产生的调度动作计算回报函数；

所述预测模块还包括：

奖励函数单元：用于度量调度策略的好坏，使整个网络中各个服务节点负载均衡。

本申请实施例采取的技术方案还包括：所述智能体模型训练模块利用多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型具体为：每个服务节点上的智能体通过预测模块调整需要调度的资源大小，标记出需要调度出去的虚拟机，根据待调度状态的虚拟机产生调度策略，各个服务节点分别计算自身的回报值并汇总求和得到总回报值，并根据总回报值调整各个预测模块的参数。

本申请实施例采取的技术方案还包括：所述智能体部署模块将智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度具体为：将训练好的智能体模型部署到真实环境中对应的服务节点上，所述智能体模型感知到所在服务器上的一段时间内的状态信息作为输入，预测得到当前服务器需要释放掉的资源，并使用背包算法选出最接近标准的虚拟机将其标记为待调度状态；之后通过调度模块收集到所有服务器上的预测结果与被标记为待调度状态的虚拟机，再按需将待调度状态的虚拟机指派给适合的服务器产生调度策略，将调度命令分发至对应服务节点执行调度操作；在执行调度策略之前对每

个调度命令进行校验是否合法，若不合法则反馈一个惩罚奖励更新参数，重新产生调度策略;若合法则执行调度操作,并获得反馈的奖励值更新智能体参数。

本申请实施例采取的又一技术方案为：一种电子设备，包括：

至少一个处理器；以及

与所述至少一个处理器通信连接的存储器；其中，

所述存储器存储有可被所述一个处理器执行的指令,所述指令被所述至少一个处理器执行,以使所述至少一个处理器能够执行上述的多智能体强化学习调度方法的以下操作：

步骤a：收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息;

步骤b：使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建立多智能体的深度强化学习模型;

步骤c：利用所述多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型;

步骤d：将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

相对于现有技术，本申请实施例产生的有益效果在于：本申请实施例的多智能体强化学习调度方法、系统及电子设备通过虚拟化技术将服务器上运行的服务虚拟化，通过调度虚拟机的方式来进行负载均衡，因为调度范围不在局限于单个服务器内部,当一台服务器处于高负载状态下的时候可以将其中的虚拟机调度到其他低负载的服务器上运行，相比分配资源的方案更加宏观。同时，本申请使用了MADDPG框架在AC框架上进行扩展,critic增加了其他智能体的进行决策的额外信息，但是每个actor只能使用本地的信息训练，通过这种框架就可以实现多智能体在复杂的动态环境下产生协作的策略。

附图说明

图 1 是本申请实施例的多智能体强化学习调度方法的流程图;

图 2 是本申请实施例的 MADDPG 调度框架示意图;

图 3 是本申请实施例的调度总体框架示意图;

图 4 是本申请实施例的多智能体强化学习调度系统的结构示意图;

图 5 是本申请实施例提供的多智能体强化学习调度方法的硬件设备结构示意图。

具体实施方式

为了使本申请的目的、技术方案及优点更加清楚明白，以下结合附图及实施例，对本申请进行进一步详细说明。应当理解，此处所描述的具体实施例仅用以解释本申请，并不用于限定本申请。

为了解决现有技术中存在的不足，本申请实施例的多智能体强化学习调度方法通过使用强化学习领域中的多智能体强化学习技术，根据在云服务的环境下各个服务节点上的负载信息建模，使用循环神经网络学习时序信息进行决策，为每个服务器训练一个智能体，在多个不同任务的智能体进行竞争或协同工作来维护整个网络拓扑结构下的负载均衡。完成初步训练后将各个智能体下放到真实的服务节点，之后根据各个节点的负载情况进行调度，在决策和调度的同时每个智能体根据当前独立环境与其他节点的决策记忆继续学习完善，使得每个智能体能够与其他节点的智能体互相协作产生调度策略，实现各个服务节点的负载均衡。

具体的，请参阅图 1，是本申请实施例的多智能体强化学习调度方法的流程图。本申请实施例的多智能体强化学习调度方法包括以下步骤:

步骤 100: 收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息;

步骤 100 中，收集的服务器参数具体包括: 收集真实场景下一段时间的各

个服务器的配置信息，内存与硬盘存储空间等；收集的虚拟机负载信息具体包括：收集每台服务器上运行的虚拟机占用资源的参数，例如 CPU 占用率、内存与硬盘占用率等。

步骤 200：将收集到的服务器参数和虚拟机负载信息进行规范化等预处理操作；

步骤 200 中，预处理操作具体包括：定义每个服务节点虚拟机信息为一个多元组，多元组包括虚拟机的数量与其各自的配置，包括 CPU、内存、硬盘及当前所处状态，每个虚拟机包括两个调度状态，分别为待调度状态和运行状态，每个服务节点包括两个状态，分别为饱和状态和饥饿状态，各个虚拟机占用的资源比之和不能够多于所在服务器配置的上限。

步骤 300：使用预处理后的数据建立虚拟仿真环境，并建立多智能体的深度强化学习模型；

步骤 300 中，建立多智能体的深度强化学习模型具体包括：将收集到的时序动态信息（服务器参数和虚拟机负载信息）进行建模创建模拟环境进行离线训练，模型采用多智能体的深度强化学习模型，为了充分利用时序数据的影响，模型中深度网络部分采用 LSTM 模型来提取时序信息，避免瞬时状态下异常数据波动对决策产生的影响。模型采用 MADDPG（即为 Multi-Agent Deep Deterministic Policy Gradient，来自于 OpenAI 的 Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments）框架，MADDPG 框架是 DDPG（来自于 Google DeepMind 发表的 continuous control with deep reinforcement learning 文章中）算法在多智能体领域的拓展，DDPG 算法将深度强化学习应用到连续动作空间上。深度学习部分得出的动作空间设定为待调度状态的虚拟机的资源占比，即调度走多少空间才可以保持当前服务节点的负载平衡。根据得出的待调度空间来将大小合适的虚拟机标记为待调度状态，然后计算整个网络中各个服务节点上处于待调度状态的虚拟机与各服务节点的回报奖励，使用虚拟机分配到服务节点所获得的奖励值作为距离度量，产生调度策略，最后校验调度策略是否可执行，可执行则将处于待调度状态的虚拟机调度到其他合适

的服务节点上，不可执行的策略将返回一个负反馈惩罚，智能体重新产生调度策略。详细调度框架如图 2 所示。

本申请实施例中，为了解决动态环境下某些瞬时异常负载波动带来的影响，使用循环神经网络 LSTM(长短时记忆网络)取代深度强化学习中的全连接神经网络，让智能体可以学习到时序数据之间隐藏的信息，从而实现基于时空感知的自适应调度。

上述中，利用各个服务节点上的智能体将虚拟机标记为待调度状态采用了背包问题解法，将预测得到的待调度空间作为背包空间，每个虚拟机的占用资源作为物品重量与价值，只需计算背包能够装入的最大价值，将装入的虚拟机标记为待调度状态即可。然后统计服务节点上预测得出的待调度空间（其中存在负数表示需要调度进来多少资源能够充分利用资源），目标是待调度空间占用与各个服务节点的待调度之和最小，通过计算可得出调度策略。

本申请实施例中，MADDPG 框架将深度强化学习的技术拓展到了多智能体领域，算法适用于多智能体环境下的集中式学习（Centralized learning）和分散式执行（Decentralized execution），使用该框架可以使多智能体之间学会协作与竞争。

具体的，MADDPG 算法通过考虑多个参数化 $\theta = \{\theta_1, \theta_2, \theta_3, \ldots \theta_n\}$ 的多个智能体的博弈来计算策略 Policy，可将所有智能体的策略定义为 $\pi = \{\pi_1, \pi_2, \pi_3, \ldots \pi_n\}$，第 i 个智能体的期望收益为 $J(\theta_i) = E[R_i]$，则在考虑确定性策略 $\mu_{\theta i}$ 的 θ_i 为参数时，梯度可表示为下式：

$$\nabla_{\theta_i} J(\mu_{\theta i}) = E_{x,a\sim D}[\nabla_{\theta_i} \mu_{\theta i}(a_i|o_i) \nabla_{a_i} Q_i^{\mu}(x, a_1 \ldots a_n)|_{a_i=\mu_i(o_i)}] \quad (1)$$

其中 $x = (o_1 \ldots o_n)$。

具体的，深度强化学习模型包括预测模块和调度模块，预测模块包括状态

感知单元、动作空间单元和奖励函数单元，具体功能如下：

状态感知单元：通过各个节点输入的信息对当前状态下需要调度出去的资源进行预测，输入状态通过各个节点的负载信息以及运行的虚拟机所占资源进行定义；

动作空间单元：根据当前节点的配置信息将动作空间映射到当前服务节点的总容量之内；

调度模块：根据标记出来的待调度状态的虚拟机，进行重新调度分配产生调度策略，各个服务节点上的智能体根据产生的调度动作计算回报函数；

奖励函数单元：度量调度策略的好坏，其目标是整个网络中各个服务节点负载均衡，其中每个服务节点上的回报函数是单独来计算的；回报函数公式如下：

$$r_i = \frac{1}{1+\beta\times\left(e^{\max(0,c-\alpha)}-1\right)} \quad (2)$$

上式中，r_i是每个服务节点上的奖励回报，其中 c 代表第 i 台机器上的 CPU 占用率，α，β是惩罚系数。α可以根据情况设定，表示希望服务器 CPU 占用率负载保持稳态的阈值。

$$R = \sum_i^n r_i \quad (3)$$

上式中，R 为整体回报函数，最终优化目标为各个智能体协作产生的调度策略得到最大的 R。

步骤 400：利用多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型；

步骤 400 中，在根据真实数据所建立的模拟环境下进行离线训练，对每个服务节点分别创建一个智能体，每个服务节点上的智能体通过预测模块调整需要调度的资源大小，标记出需要调度出去的虚拟机，根据待调度状态的虚拟机产生调度策略，然后各个服务节点分别计算出自身的回报值并汇总求和得到总回报值，最后根据总回报值调整各个预测模块的参数。

步骤500：将训练好的智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

步骤500中，将训练好的各个智能体模型下放到真实环境中对应的服务节点上，智能体首先感知到所在服务器上的一段时间内的状态信息作为输入，通过智能体的预测模块预测得到当前服务器希望释放掉的资源，然后使用背包算法选出最接近标准的虚拟机将其标记为待调度状态；之后通过调度模块收集到所有服务器上的预测结果与被标记为待调度状态的虚拟机，再按需将待调度状态的虚拟机指派给合适的服务器产生调度策略，将调度命令分发至对应节点执行调度操作。在执行调度策略之前需要对每个调度命令进行校验是否合法，若不合法则反馈一个惩罚奖励更新参数，重新产生调度策略，反复迭代直到全部调度策略均可执行。若合法则执行并获得反馈的奖励值更新智能体参数。具体的调度总体框架如图3所示。

对于普通的多智能体强化学习通常情况下会根据环境输入直接得到调度动作，但是在复杂的网络拓扑结构中对于虚拟机调度策略来说的动作空间过于庞大，在如此庞大动作空间上或导致算法难以收敛，而且使用此种方式便需要将每一个运行在其中的虚拟机都配置一个全局id，用来指定调度的目标，但是需要注意的是虽然id可以索引到虚拟机，但是虚拟机占用的资源是有可能在运行过程中发生变化的，所以在学习过程中学到的策略是不可靠的。即便假设虚拟机的占用资源不会变化，此时如果新增加一个虚拟机，那么基于上述算法所训练的智能体在决策时是不会考虑新增加的虚拟机的。因此本申请在上述算法的基础上加以改进，使模型的动作空间替换为当前服务器希望释放的资源，即表示希望从中调度出多少资源来保持整体网络拓扑结构下的负载均衡。这样的设置可以避免使用全局id来标记各个虚拟机，即便中途增加新的虚拟机仍然可以工作，所以使得调度算法更加灵活可以自适应更广泛的场景。

请参阅图4，是本申请实施例的多智能体强化学习调度系统的结构示意图。本申请实施例的多智能体强化学习调度系统包括信息收集模块、预处理模块、强化学习模型构建模块、智能体模型训练模块和智能体部署模块。

信息收集模块：用于收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息；其中，收集的服务器参数具体包括：收集真实场景下一段时间的各个服务器的配置信息，内存与硬盘存储空间等；收集的虚拟机负载信息具体包括：收集每台服务器上运行的虚拟机占用资源的参数，例如 CPU 占用率、内存与硬盘占用率等。

预处理模块：用于将收集到的服务器参数和虚拟机负载信息进行规范化等预处理操作；其中，预处理操作具体包括：定义每个服务节点虚拟机信息为一个多元组，多元组包括虚拟机的数量与其各自的配置，包括 CPU、内存、硬盘及当前所处状态，每个虚拟机包括两个调度状态，分别为待调度状态和运行状态，每个服务节点包括两个状态，分别为饱和状态和饥饿状态，各个虚拟机占用的资源比之和不能够多于所在服务器配置的上限。

强化学习模型构建模块：用于使用预处理后的数据建立虚拟仿真环境，并建立多智能体的深度强化学习模型；其中，建立多智能体的深度强化学习模型具体包括：将收集到的时序动态信息（服务器参数和虚拟机负载信息）进行建模创建模拟环境进行离线训练，模型采用多智能体的深度强化学习模型，为了充分利用时序数据的影响，模型中深度网络部分采用 LSTM 模型来提取时序信息，避免瞬时状态下异常数据波动对决策产生的影响。模型采用 MADDPG 框架，MADDPG 框架是 DDPG 算法在多智能体领域的拓展，DDPG 算法将深度强化学习应用到连续动作空间上。深度学习部分得出的动作空间设定为待调度状态的虚拟机的资源占比，即调度走多少空间才可以保持当前服务节点的负载平衡。根据得出的待调度空间来将大小合适的虚拟机标记为待调度状态，然后计算整个网络中各个服务节点上处于待调度状态的虚拟机与各服务节点的回报奖励，使用虚拟机分配到服务节点所获得的奖励值作为距离度量，产生调度策略，最后校验调度策略是否可执行，可执行则将处于待调度状态的虚拟机调度到其他合适的服务节点上，不可执行的策略将返回一个负反馈惩罚，智能体重新产生调度策略。

本申请实施例中，为了解决动态环境下某些瞬时异常负载波动带来的影

响，使用循环神经网络 LSTM(长短时记忆网络)取代深度强化学习中的全连接神经网络，让智能体可以学习到时序数据之间隐藏的信息，从而实现基于时空感知的自适应调度。

上述中，利用各个服务节点上的智能体将虚拟机标记为待调度状态采用了背包问题解法，将预测得到的待调度空间作为背包空间，每个虚拟机的占用资源作为物品重量与价值，只需计算背包能够装入的最大价值，将装入的虚拟机标记为待调度状态即可。然后统计服务节点上预测得出的待调度空间（其中存在负数表示需要调度进来多少资源能够充分利用资源），目标是待调度空间占用与各个服务节点的待调度之和最小，通过计算可得出调度策略。

本申请实施例中，MADDPG 框架将深度强化学习的技术拓展到了多智能体领域，算法适用于多智能体环境下的集中式学习（Centralized learning）和分散式执行（Decentralized execution），使用该框架可以使多智能体之间学会协作与竞争。

具体的，MADDPG 算法通过考虑多个参数化 $\theta = \{\theta_1, \theta_2, \theta_3, \ldots \theta_n\}$ 的多个智能体的博弈来计算策略 Policy，可将所有智能体的策略定义为 $\pi = \{\pi_1, \pi_2, \pi_3, \ldots \pi_n\}$，第 i 个智能体的期望收益为 $J(\theta_i) = E[R_i]$，则在考虑确定性策略 $\mu_{\theta i}\theta_i$ 为参数时，梯度可表示为下式：

$$\nabla_{\theta_i} J(\mu_{\theta i}) = E_{x,a\sim D}[\nabla_{\theta_i} \mu_{\theta i}(a_i|o_i) \nabla_{a_i} Q_i^{\mu}(x, a_1 \ldots a_n)|_{a_i=\mu_i(o_i)}] \quad (1)$$

其中 $x = (o_1 \ldots o_n)$。

进一步地，强化学习模型构建模块包括预测模块和调度模块，预测模块包括状态感知单元、动作空间单元和奖励函数单元，具体功能如下：

状态感知单元：通过各个节点输入的信息对当前状态下需要调度出去的资源进行预测，输入状态通过各个节点的负载信息以及运行的虚拟机所占资源进

行定义;

动作空间单元:根据当前节点的配置信息将动作空间映射到当前服务节点的总容量之内;

调度模块:根据标记出来的待调度状态的虚拟机,进行重新调度分配产生调度策略,各个服务节点上的智能体根据产生的调度动作计算回报函数;

奖励函数单元:度量调度策略的好坏,其目标是整个网络中各个服务节点负载均衡,其中每个服务节点上的回报函数是单独来计算的;回报函数公式如下:

$$r_i = \frac{1}{1+\beta\times\left(e^{\max(0,c-\alpha)}-1\right)} \quad (2)$$

上式中,r_i是每个服务节点上的奖励回报,其中c代表第i台机器上的CPU占用率,α, β是惩罚系数。α可以根据情况设定,表示希望服务器 CPU 占用率负载保持稳态的阈值。

$$R = \sum_i^n r_i \quad (3)$$

上式中,R为整体回报函数,最终优化目标为各个智能体协作产生的调度策略得到最大的R。

智能体模型训练模块:用于利用多智能体的深度强化学习模型和模拟环境进行离线训练和学习,为每个服务器分别训练一个智能体模型;其中,在根据真实数据所建立的模拟环境下进行离线训练,对每个服务节点分别创建一个智能体,每个服务节点上的智能体通过预测模块调整需要调度的资源大小,标记出需要调度出去的虚拟机,根据待调度状态的虚拟机产生调度策略,然后各个服务节点分别计算出自身的回报值并汇总求和得到总回报值,最后根据总回报值调整各个预测模块的参数。

智能体部署模块:用于将训练好的智能体模型部署到真实的服务节点,并根据各个服务节点的负载情况进行调度。其中,将训练好的各个智能体模型下放到真实环境中对应的服务节点上,然后通过智能体的预测模块进行预测修改

待调度状态，调度模块统一分配产生调度策略，将调度命令分发至对应节点执行调度操作，调度动作执行之前需要判断动作能否执行，若无法执行或执行失败则反馈一个惩罚奖励更新参数，重新产生调度策略，反复迭代直到全部调度策略均可执行。

对于普通的多智能体强化学习通常情况下会根据环境输入直接得到调度动作，但是在复杂的网络拓扑结构中对于虚拟机调度策略来说的动作空间过于庞大，在如此庞大动作空间上或导致算法难以收敛，而且使用此种方式便需要将每一个运行在其中的虚拟机都配置一个全局 id，用来指定调度的目标，但是需要注意的是虽然 id 可以索引到虚拟机，但是虚拟机占用的资源是有可能在运行过程中发生变化的，所以在学习过程中学到的策略是不可靠的。即便假设虚拟机的占用资源不会变化，此时如果新增加一个虚拟机，那么基于上述算法所训练的智能体在决策时是不会考虑新增加的虚拟机的。因此本申请在上述算法的基础上加以改进，使模型的动作空间替换为当前服务器希望释放的资源，即表示希望从中调度出多少资源来保持整体网络拓扑结构下的负载均衡。这样的设置可以避免使用全局 id 来标记各个虚拟机，即便中途增加新的虚拟机仍然可以工作，所以使得调度算法更加灵活可以自适应更广泛的场景。

图 5 是本申请实施例提供的多智能体强化学习调度方法的硬件设备结构示意图。如图 5 所示，该设备包括一个或多个处理器以及存储器。以一个处理器为例，该设备还可以包括：输入系统和输出系统。

处理器、存储器、输入系统和输出系统可以通过总线或者其他方式连接，图 5 中以通过总线连接为例。

存储器作为一种非暂态计算机可读存储介质，可用于存储非暂态软件程序、非暂态计算机可执行程序以及模块。处理器通过运行存储在存储器中的非暂态软件程序、指令以及模块，从而执行电子设备的各种功能应用以及数据处理，即实现上述方法实施例的处理方法。

存储器可以包括存储程序区和存储数据区，其中，存储程序区可存储操作系统、至少一个功能所需要的应用程序；存储数据区可存储数据等。此外，存

储器可以包括高速随机存取存储器，还可以包括非暂态存储器，例如至少一个磁盘存储器件、闪存器件、或其他非暂态固态存储器件。在一些实施例中，存储器可选包括相对于处理器远程设置的存储器，这些远程存储器可以通过网络连接至处理系统。上述网络的实例包括但不限于互联网、企业内部网、局域网、移动通信网及其组合。

输入系统可接收输入的数字或字符信息，以及产生信号输入。输出系统可包括显示屏等显示设备。

所述一个或者多个模块存储在所述存储器中，当被所述一个或者多个处理器执行时，执行上述任一方法实施例的以下操作：

步骤 a：收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息；

步骤 b：使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建立多智能体的深度强化学习模型；

步骤 c：利用所述多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型；

步骤 d：将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

上述产品可执行本申请实施例所提供的方法，具备执行方法相应的功能模块和有益效果。未在本实施例中详尽描述的技术细节，可参见本申请实施例提供的方法。

本申请实施例提供了一种非暂态（非易失性）计算机存储介质，所述计算机存储介质存储有计算机可执行指令，该计算机可执行指令可执行以下操作：

步骤 a：收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息；

步骤 b：使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建立多智能体的深度强化学习模型；

步骤 c：利用所述多智能体的深度强化学习模型和模拟环境进行离线训练

和学习，为每个服务器分别训练一个智能体模型；

步骤 d：将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

本申请实施例提供了一种计算机程序产品，所述计算机程序产品包括存储在非暂态计算机可读存储介质上的计算机程序，所述计算机程序包括程序指令，当所述程序指令被计算机执行时，使所述计算机执行以下操作：

步骤 a：收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息；

步骤 b：使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建立多智能体的深度强化学习模型；

步骤 c：利用所述多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型；

步骤 d：将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

本申请实施例的多智能体强化学习调度方法、系统及电子设备通过虚拟化技术将服务器上运行的服务虚拟化，通过调度虚拟机的方式来进行负载均衡，因为调度范围不在局限于单个服务器内部，当一台服务器处于高负载状态下的时候可以将其中的虚拟机调度到其他低负载的服务器上运行，相比分配资源的方案更加宏观。同时，本申请使用了 MADDPG 框架在 AC 框架上进行扩展，critic 增加了其他智能体的进行决策的额外信息，但是每个 actor 只能使用本地的信息训练，通过这种框架就可以实现多智能体在复杂的动态环境下产生协作的策略。

对所公开的实施例的上述说明，使本领域专业技术人员能够实现或使用本申请。对这些实施例的多种修改对本领域的专业技术人员来说将是显而易见的，本申请中所定义的一般原理可以在不脱离本申请的精神或范围的情况下，在其它实施例中实现。因此，本申请将不会被限制于本申请所示的这些实施例，而是要符合与本申请所公开的原理和新颖特点相一致的最宽的范围。

权 利 要 求

1、一种多智能体强化学习调度方法，其特征在于，包括以下步骤：

步骤 a：收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息；

步骤 b：使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建立多智能体的深度强化学习模型；

步骤 c：利用所述多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型；

步骤 d：将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

2、根据权利要求 1 所述的多智能体强化学习调度方法，其特征在于，所述步骤 a 还包括：将收集到的服务器参数和虚拟机负载信息进行规范化预处理操作；所述规范化预处理操作包括：定义每个服务节点虚拟机信息为一个多元组，所述多元组包括虚拟机的数量与其各自的配置，每个虚拟机包括两个调度状态，分别为待调度状态和运行状态，每个服务节点包括两个状态，分别为饱和状态和饥饿状态，各个虚拟机占用的资源比之和少于所在服务器配置的上限。

3、根据权利要求 1 或 2 所述的多智能体强化学习调度方法，其特征在于，在所述步骤 b 中，所述多智能体的深度强化学习模型具体包括预测模块和调度模块，所述预测模块通过各个服务节点输入的信息对当前状态下需要调度出去的资源进行预测，根据当前服务节点的配置信息将动作空间映射到当前服务节点的总容量之内；所述调度模块根据标记出来的待调度状态的虚拟机，进行重新调度分配产生调度策略，各个服务节点上的智能体根据产生的调度动作计算回报函数；

所述预测模块度量调度策略的好坏，使整个网络中各个服务节点负载均衡。

4、根据权利要求 3 所述的多智能体强化学习调度方法，其特征在于，在所述步骤 c 中，所述利用多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型具体包括：每个服务节点上的智能体通过预测模块调整需要调度的资源大小，标记出需要调度出去的虚拟机，根据待调度状态的虚拟机产生调度策略，各个服务节点分别计算自身的回报值并汇总求和得到总回报值，并根据总回报值调整各个预测模块的参数。

5、根据权利要求 4 所述的多智能体强化学习调度方法，其特征在于，在所述步骤 d 中，所述将智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度具体为：将训练好的智能体模型部署到真实环境中对应的服务节点上，所述智能体模型感知到所在服务器上的一段时间内的状态信息作为输入，预测得到当前服务器需要释放掉的资源，并使用背包算法选出最接近标准的虚拟机将其标记为待调度状态；之后通过调度模块收集到所有服务器上的预测结果与被标记为待调度状态的虚拟机，再按需将待调度状态的虚拟机指派给适合的服务器产生调度策略，将调度命令分发至对应服务节点执行调度操作；在执行调度策略之前对每个调度命令进行校验是否合法，若不合法则反馈一个惩罚奖励更新参数，重新产生调度策略；若合法则执行调度操作，并获得反馈的奖励值更新智能体参数。

6、一种多智能体强化学习调度系统，其特征在于，包括：

信息收集模块：用于收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息；

强化学习模型构建模块：用于使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建立多智能体的深度强化学习模型；

智能体模型训练模块：用于利用所述多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型；

智能体部署模块：用于将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

7、根据权利要求 6 所述的多智能体强化学习调度系统，其特征在于，还包括预处理模块，所述预处理模块用于将收集到的服务器参数和虚拟机负载信息进行规范化预处理操作；所述规范化预处理操作包括：定义每个服务节点虚拟机信息为一个多元组，所述多元组包括虚拟机的数量与其各自的配置，每个虚拟机包括两个调度状态，分别为待调度状态和运行状态，每个服务节点包括两个状态，分别为饱和状态和饥饿状态，各个虚拟机占用的资源比之和少于所在服务器配置的上限。

8、根据权利要求 6 或 7 所述的多智能体强化学习调度系统，其特征在于，所述强化学习模型构建模块包括预测模块和调度模块，所述预测模块包括：

状态感知单元：用于通过各个服务节点输入的信息对当前状态下需要调度出去的资源进行预测；

动作空间单元：用于根据当前服务节点的配置信息将动作空间映射到当前服务节点的总容量之内；

所述调度模块根据标记出来的待调度状态的虚拟机，进行重新调度分配产生调度策略，各个服务节点上的智能体根据产生的调度动作计算回报函数；

所述预测模块还包括：

奖励函数单元：用于度量调度策略的好坏，使整个网络中各个服务节点负载均衡。

9、根据权利要求 8 所述的多智能体强化学习调度系统，其特征在于，所述

智能体模型训练模块利用多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型具体为：每个服务节点上的智能体通过预测模块调整需要调度的资源大小，标记出需要调度出去的虚拟机，根据待调度状态的虚拟机产生调度策略，各个服务节点分别计算自身的回报值并汇总求和得到总回报值，并根据总回报值调整各个预测模块的参数。

10、根据权利要求 9 所述的多智能体强化学习调度系统，其特征在于，所述智能体部署模块将智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度具体为：将训练好的智能体模型部署到真实环境中对应的服务节点上，所述智能体模型感知到所在服务器上的一段时间内的状态信息作为输入，预测得到当前服务器需要释放掉的资源，并使用背包算法选出最接近标准的虚拟机将其标记为待调度状态；之后通过调度模块收集到所有服务器上的预测结果与被标记为待调度状态的虚拟机，再按需将待调度状态的虚拟机指派给适合的服务器产生调度策略，将调度命令分发至对应服务节点执行调度操作；在执行调度策略之前对每个调度命令进行校验是否合法，若不合法则反馈一个惩罚奖励更新参数，重新产生调度策略；若合法则执行调度操作，并获得反馈的奖励值更新智能体参数。

11、一种电子设备，包括：

至少一个处理器；以及

与所述至少一个处理器通信连接的存储器；其中，

所述存储器存储有可被所述一个处理器执行的指令，所述指令被所述至少一个处理器执行，以使所述至少一个处理器能够执行上述 1 至 5 任一项所述的多智能体强化学习调度方法的以下操作：

步骤 a：收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负

载信息；

步骤 b：使用所述服务器参数和虚拟机负载信息建立虚拟仿真环境，并建立多智能体的深度强化学习模型；

步骤 c：利用所述多智能体的深度强化学习模型和模拟环境进行离线训练和学习，为每个服务器分别训练一个智能体模型；

步骤 d：将所述智能体模型部署到真实的服务节点，并根据各个服务节点的负载情况进行调度。

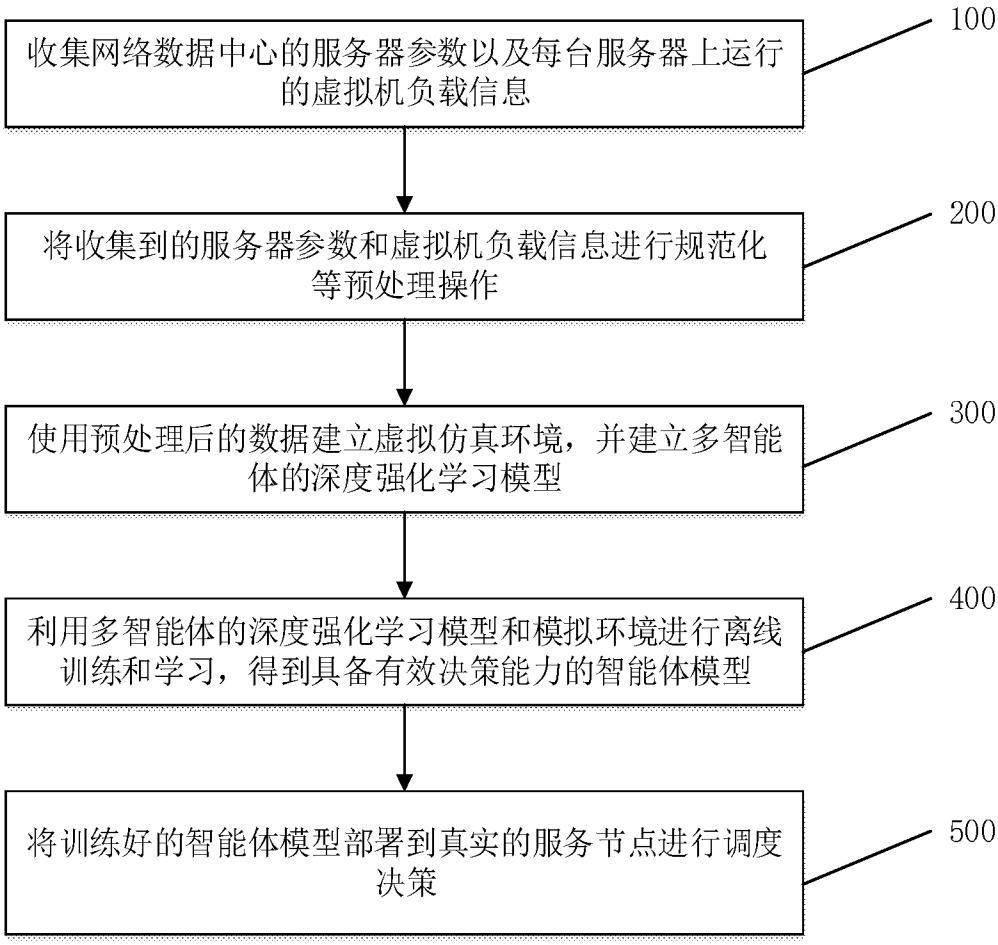
收集网络数据中心的服务器参数以及每台服务器上运行的虚拟机负载信息
100
将收集到的服务器参数和虚拟机负载信息进行规范化等预处理操作
200
使用预处理后的数据建立虚拟仿真环境，并建立多智能体的深度强化学习模型
300
利用多智能体的深度强化学习模型和模拟环境进行离线训练和学习，得到具备有效决策能力的智能体模型
400
将训练好的智能体模型部署到真实的服务节点进行调度决策
500

图 1

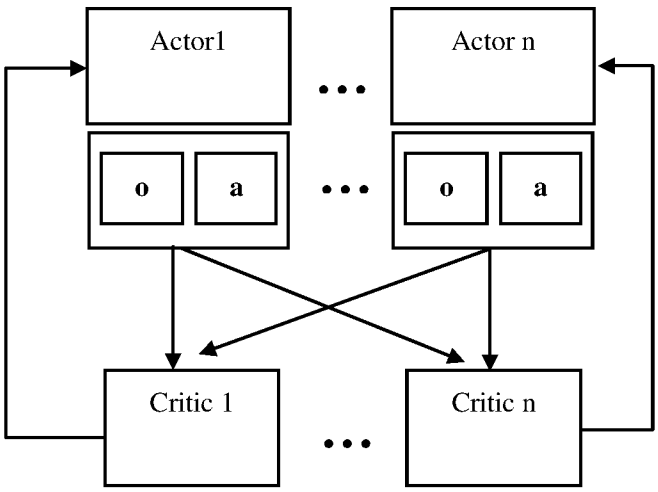
Actor1
Actor n
o
a
o
a
Critic 1
Critic n

图 2

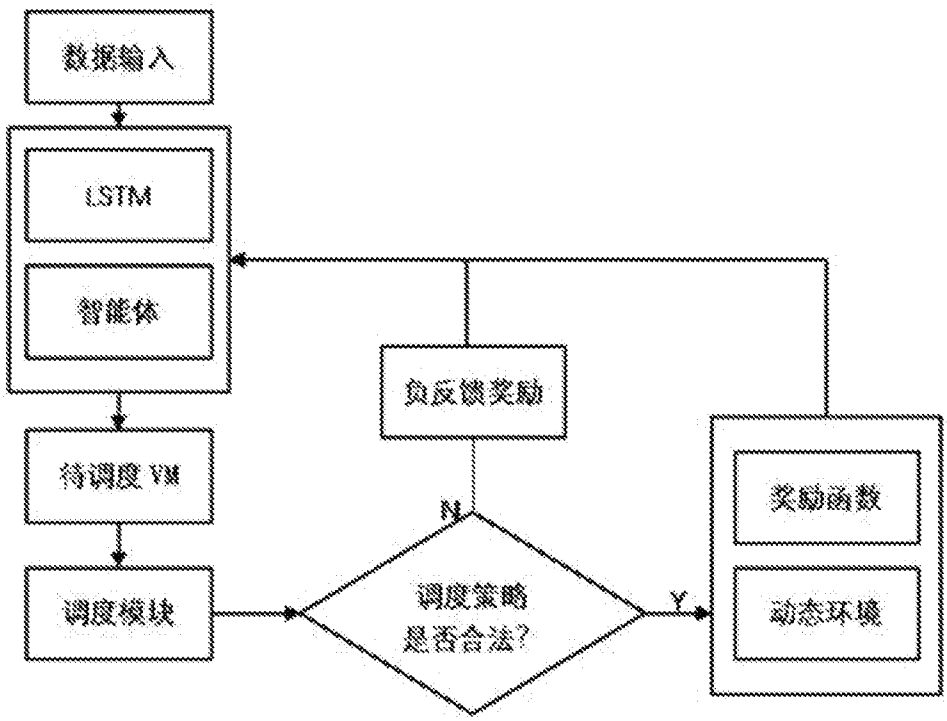
数据输入
LSTM
智能体
待调度 VM
调度模块
负反馈奖励
N
调度策略
是否合法?
Y
奖励函数
动态环境

图 3

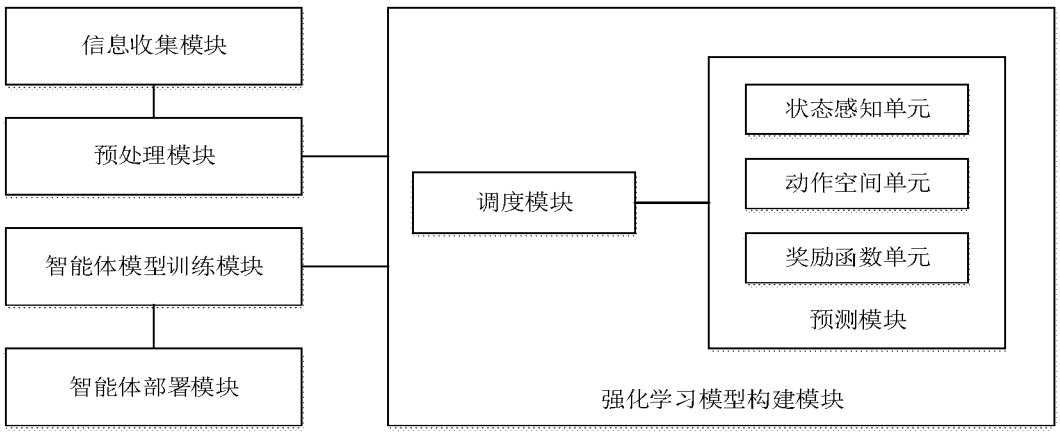
信息收集模块
预处理模块
智能体模型训练模块
智能体部署模块
调度模块
状态感知单元
动作空间单元
奖励函数单元
预测模块
强化学习模型构建模块

图 4

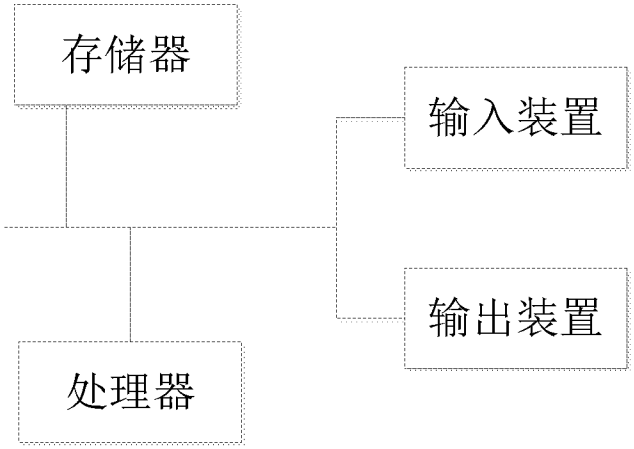

图 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2019/130582

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/50(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WPI, EPODOC, BAIDU, CNPAT, CNKI, IEEE: 智能体, 负载, 虚拟机, 服务器, 节点, 模型, 多, 预测, 调度, agent, load, virtual, machine, node, model, multi, forecast, schedule

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 109947567 A (SHENZHEN INSTITUTES OF ADVANCED TECHNOLOGY) 28 June 2019 (2019-06-28) claims 1-11	1-11
X	魏亮 (WEI, Liang). "面向云网融合的资源调度算法及实验平台研究 (Research on Resource Scheduling Algorithm and Experimental Platform for Cloud-network Integration)" *CNKI中国博士学位论文全文数据库 (CNKI, China Doctoral Dissertations Full-text Database),* 30 September 2018 (2018-09-30), chapters 3 and 4	1-3, 6-8, 11
A	WO 2018076791 A1 (HUAWEI TECHNOLOGIES CO., LTD.) 03 May 2018 (2018-05-03) entire document	1-11
A	CN 108829494 A (HANGZHOU HARMONY CLOUD TECHNOLOGY CO., LTD.) 16 November 2018 (2018-11-16) entire document	1-11
A	CN 105607952 A (SPACE STAR TECHNOLOGY CO., LTD.) 25 May 2016 (2016-05-25) entire document	1-11
A	CN 103873569 A (LAN, Yuqing) 18 June 2014 (2014-06-18) entire document	1-11

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
15 March 2020	**27 March 2020**
Name and mailing address of the ISA/CN **China National Intellectual Property Administration (ISA/CN)** **No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088** **China** Facsimile No. **(86-10)62019451**	Authorized officer Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2019/130582

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN 109947567 A	28 June 2019	None	
WO 2018076791 A1	03 May 2018	CN 108009016 A	08 May 2018
		US 2019253490 A1	15 August 2019
		EP 3525096 A1	14 August 2019
CN 108829494 A	16 November 2018	None	
CN 105607952 A	25 May 2016	None	
CN 103873569 A	18 June 2014	None	

国际检索报告

国际申请号
PCT/CN2019/130582

A. 主题的分类

G06F 9/50(2006.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

WPI, EPODOC, BAIDU, CNPAT, CNKI, IEEE: 智能体，负载，虚拟机，服务器，节点，模型，多，预测，调度，agent, load, virtual, machine, node, model, multi, forecast, schedule

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
PX	CN 109947567 A (深圳先进技术研究院) 2019年 6月 28日 (2019-06-28) 权利要求1-11	1-11
X	魏亮,. "面向云网融合的资源调度算法及实验平台研究," CNKI中国博士学位论文全文数据库, 2018年 9月 30日 (2018-09-30), 论文第三章和第四章	1-3, 6-8, 11
A	WO 2018076791 A1 (华为技术有限公司) 2018年 5月 3日 (2018-05-03) 全文	1-11
A	CN 108829494 A (杭州谐云科技有限公司) 2018年 11月 16日 (2018-11-16) 全文	1-11
A	CN 105607952 A (航天恒星科技有限公司) 2016年 5月 25日 (2016-05-25) 全文	1-11
A	CN 103873569 A (兰雨晴) 2014年 6月 18日 (2014-06-18) 全文	1-11

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

"A" 认为不特别相关的表示了现有技术一般状态的文件

"E" 在国际申请日的当天或之后公布的在先申请或专利

"L" 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

"O" 涉及口头公开、使用、展览或其他方式公开的文件

"P" 公布日先于国际申请日但迟于所要求的优先权日的文件

"T" 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

"X" 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

"Y" 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

"&" 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2020年 3月 15日	2020年 3月 27日
ISA/CN的名称和邮寄地址 中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451	受权官员 吴少鸿 电话号码 86-(10)-53961533

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2019/130582

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	109947567	A	2019年 6月 28日	无			
WO	2018076791	A1	2018年 5月 3日	CN	108009016	A	2018年 5月 8日
				US	2019253490	A1	2019年 8月 15日
				EP	3525096	A1	2019年 8月 14日
CN	108829494	A	2018年 11月 16日	无			
CN	105607952	A	2016年 5月 25日	无			
CN	103873569	A	2014年 6月 18日	无			