(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 117453419 A
(43)申请公布日 2024.01.26

(21)申请号 202311552801.4

(22)申请日 2023.11.20

(71)申请人 北京航空航天大学
地址 100191 北京市海淀区学院路37号

(72)发明人 吴文峻 丁嵘 杨开元 陈睿博

(74)专利代理机构 北京永创新实专利事务所 11121
专利代理师 易卜

(51)Int.Cl.
G06F 9/50(2006.01)
G06F 9/54(2006.01)
G06N 3/006(2023.01)
G06N 20/00(2019.01)
G06F 17/18(2006.01)

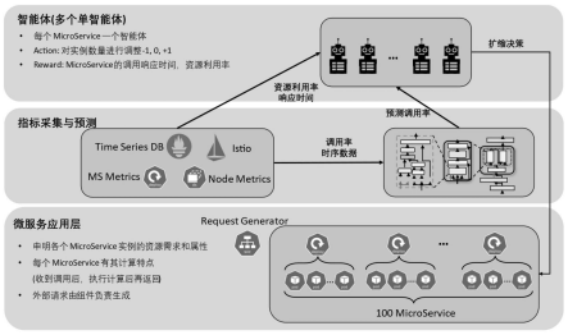
权利要求书3页 说明书6页 附图2页

(54)发明名称

一种基于负载预测和强化学习的微服务管理方法

(57)摘要

本发明公开了一种基于负载预测和强化学习的微服务管理方法,属于微服务资源管理领域。首先通过数据收集系统收集每个周期平台中各个微服务的调用次数和平均响应时间作为指标,并按结构化的形式,存储至时序数据库中。然后负载预测模块利用存储的时序数据获取各个微服务调用率数据,并对下一个时间周期的各微服务调用率进行预测,将预测结果输入强化学习智能体,对强化学习智能体进行训练,通过训练后的策略网络输出扩缩决策。最后水平扩缩控制器按照强化学习智能体输出的扩缩决策,对相应微服务副本数量进行调整,增加、减少或保持其副本数量。本发明可以帮助智能体更好地决策,避免用户请求响应时间短时间内恶化的抖动现象。



CN 117453419 A

1.一种基于负载预测和强化学习的微服务管理方法,其特征在于,具体步骤如下:

步骤一、对于每一个时间周期,通过数据收集系统收集平台中各个微服务的调用次数和平均响应时间作为指标,并按结构化的形式,存储至时序数据库中;

步骤二、负载预测模块通过存储的时序数据以时间序列的形式获取各个微服务调用率数据,并对下一个时间周期的各微服务调用率进行预测;

步骤三、将预测结果输入强化学习智能体,对强化学习智能体进行训练,通过训练后的策略网络输出扩缩决策;

通过M个episode,每个episode执行N个step对强化学习智能体进行训练,具体步骤如下:

步骤301:初始时,获取智能体的状态空间表征和动作空间表征,并随机初始化策略网络θ_0和状态估值网络φ_0;

状态空间表征为S,包括下一个周期调用率预测值和当前微服务副本数量;

动作空间表征为$\mathcal{A}$,$\mathcal{A}=\{\chi_0,\chi_1,\chi_2\}$,动作$\chi_0$表示将微服务副本减少一个,动作$\chi_1$表示不对微服务副本数量进行操作,动作$\chi_2$表示将微服务副本增加一个;

步骤302,从k＝0episode开始,进行该episode中从t＝0开始每个时间周期的循环迭代;

步骤303,从t＝0的周期开始,根据此时智能体的状态空间表征,得到智能体当前的状态s_t,通过策略π_{θ_k}获得智能体的动作a_t;

步骤304,当智能体对微服务施加动作a_t后,得到此时微服务的平均响应时间,并结合用户指定的期望响应时间lim,设计奖励函数r;

奖励函数r包括动作奖励函数r_{base}和合法性奖励函数r_{legal},公式为:

$r=r_{base}+r_{legal}$

其中,动作奖励函数r_{base}表示为:

$$r_{base}=\begin{cases}a_t=\chi_0\begin{cases}rt_{t+1}>lim & -4K\\ rt_{t+1}\le lim & 2K\end{cases}\\ a_t=\chi_1\begin{cases}rt_{t+1}>lim & -4K\\ rt_{t+1}\le lim & 0\end{cases}\\ a_t=\chi_2\begin{cases}rt_{t+1}>lim & 4K\\ rt_{t+1}\le lim & -3K\end{cases}\end{cases}$$

其中,rt_{t+1}表示当前动作施加到系统环境后,下一个周期微服务的平均响应时间;K代表奖励的基本数值,以常数的形式出现;

当智能体决策使用动作χ_0或χ_2时,判断微服务副本数量是否合法,得到合法性奖励函数r_{legal}表示为:

$$r_{legal}=\begin{cases}-4K & a_t\text{ 执行后进入非法状态}\\ 0 & a_t\text{ 执行后保持合法状态}\end{cases}$$

步骤305,判断当前的时间周期是否满足t≤N,如果是,返回步骤303继续当前episode的迭代;否则,计算当前策略网络的优势函数,并执行步骤306;

对于当前策略网络θ_k驱动的策略π,即π_{θ_k},其优势函数表示为

$$A^{\pi_{\theta_k}}(s,a)=Q^{\pi}(s,a)-V^{\pi}(s)$$

其中,$Q^{\pi}(s,a)$为Bellman方程中的Q函数,表达在状态s中执行动作a后,按照策略π行动获得的期望回报;$Q^{\pi}(s,a)$为Bellman方程中的V函数,表达从状态s开始,按照策略π行动获得的期望回报;

步骤306:根据优势函数,通过梯度上升的方法,对如下目标函数执行优化:

$$L(s,a,\theta_k,\theta)=\min(\frac{\pi_{\theta}(a|s)}{\pi_{\theta_k}(a|s)}A^{\pi_{\theta_k}}(s,a),g(\epsilon,A^{\pi_{\theta_k}}(s,a)))$$

其中,$g(\epsilon,A)=\begin{cases}(1+\epsilon)A & A\geq 0\\(1-\epsilon)A & A<0\end{cases}$,常数∈为策略优化函数的截断区间;$\theta_k$表示驱动策略网络的当前参数,$\theta$表示执行目标函数梯度上升时,网络优化的方向;

步骤307:判断是否满足k≤M,如果是,返回步骤302继续迭代;否则,获得了最终策略网络的最终权重θ,进入步骤308;

步骤308:策略网络在使用策略π_{θ}时,将下一个周期调用率预测值和当前微服务副本数量作为输入,输出扩缩决策;

步骤四、水平扩缩控制器按照强化学习智能体输出的扩缩决策,对相应微服务副本数量进行调整,增加、减少或保持其副本数量。

2.根据权利要求1所述的一种基于负载预测和强化学习的微服务管理方法,其特征在于,所述指标的结构化形式包括名称name、标签label、时间戳timestamp和相应指标数值value,并将标签用于对同一指标在不同情况下的分类,具体表示为name{label=`标签值`}value timestamp。

3.根据权利要求1所述的一种基于负载预测和强化学习的微服务管理方法,其特征在于,所述负载预测模块进行微服务调用率预测的步骤如下:

步骤201:对当前系统中微服务的调用数据进行特征工程,根据实际情况选择负载预测模块使用的模型;

步骤202:将历史数据划分为测试集和训练集,对模型进行训练,确定其最优参数;同时选择执行每次预测时,使用历史调用率数据数量的最优值,即历史数据窗口长度;

步骤203:将训练好的模型用于负载预测:在决策周期内,通过Prometheus相关接口,获得历史数据窗口的调用率数据,结合训练好的模型获得下一个周期调用率的预测值。

4.根据权利要求3所述的一种基于负载预测和强化学习的微服务管理方法,其特征在于,所述特征工程包括调用数据的复杂程度和历史数据的数量,模型包括ARIMA数学模型和基于注意力机制的深度学习模型。

5.根据权利要求1所述的一种基于负载预测和强化学习的微服务管理方法,其特征在于,所述奖励函数指导智能体的动作,具体为:

(1)在当前副本数量不足,将导致响应时间超出阈值的情况下,指导智能体选择唯一能提供正奖励的动作χ_2;

(2)在当前副本数增加至恰好能保证阈值时,指导智能体选择操作χ_1保持副本数量的不变,保证智能体动作的有效性;

(3)在当前副本数量超过满足响应时间的阈值时,指导智能体选择唯一能提供正奖励的动作χ_0。

一种基于负载预测和强化学习的微服务管理方法

技术领域

[0001] 本发明属于微服务资源管理领域,具体涉及一种基于负载预测和强化学习的微服务管理方法。

背景技术

[0002] 在微服务支撑平台中,面向海量微服务的复杂场景,管理目标主要考虑以下两个方向:

[0003] 第一个方向是优化时间指标:在微服务领域,时间指标主要指对用户请求的响应时间,也代表了微服务应用的性能。当微服务应用负载过重时,来自用户的请求可能得不到高效的处理,从而导致用户请求响应时间的延长。通常用户与云平台提供商会就SLA(Service Level Agreement,服务水平协议)达成统一,如99%的服务响应时间小于1ms。优化响应时间的目标即尽力避免违反SLA的情况出现。

[0004] 第二个方向是优化资源使用:在微服务领域,资源使用主要指整个微服务系统平台中的CPU和内存资源。

[0005] 如何在保证微服务响应时间的前提下,尽可能节约资源是长期的研究问题。

[0006] 对微服务执行水平扩缩是可靠的研究方向之一:在微服务资源紧张,导致响应时间恶化的情况下,增加微服务的副本数量,提高其可用资源,能有效保障微服务的时间指标;当用户请求稀少时,通过减少微服务的副本数量,节约集群资源。

[0007] 基于Kubernetes平台,现有水平扩缩方法多为规则式扩缩,如通过检测当前微服务的资源使用情况,进行副本数量调整,这一类的方法有如下问题:

[0008] 1.存在一定滞后性;以基于资源利用率的规则为例:负载增加时,当达到设定的阈值,触发扩增动作,此时需要数十秒甚至数分钟才能在真实环境中生效,存在一定滞后性;当负载波动时,可能做出错误的缩减动作,导致响应时间恶化。

[0009] 2.规则设定需一定经验;对于不同的微服务,需要运维人员根据经验设计扩缩阈值。若阈值过低,则导致微服务副本数量多于实际需要的数量,导致资源浪费;若阈值过高,则在负载增加时,微服务副本扩增不及时,导致时间指标的恶化。

[0010] 因此,使用负载预测和强化学习方法建立微服务管理系统,避免滞后性,提高扩缩策略的鲁棒性是更好的选择。

发明内容

[0011] 本发明针对上述问题,提出了一种基于负载预测和强化学习的微服务管理方法,使用时序预测方法对微服务的未来负载情况进行预测,避免微服务扩缩动作的滞后性;通过强化学习算法训练、施加扩缩策略,不完全依赖于人工根据经验设计的阈值,高效地满足在保障微服务响应时间的前提下,节约资源的需求。

[0012] 所述基于负载预测和强化学习的微服务管理方法,具体步骤如下:

[0013] 步骤一、对于每一个时间周期,通过数据收集系统收集平台中各个微服务的调用

次数和平均响应时间作为指标,并按结构化的形式,存储至时序数据库中。

[0014] 指标的结构化形式包括名称name、标签label、时间戳timestamp和相应指标数值value,并将标签用于对同一指标在不同情况下的分类,具体表示为name{label=`标签值`}value timestamp。

[0015] 步骤二、负载预测模块通过存储的时序数据以时间序列的形式获取各个微服务调用率数据,并对下一个时间周期的各微服务调用率进行预测。

[0016] 所述负载预测的步骤如下:

[0017] 步骤201:对当前系统中微服务的调用数据进行特征工程,根据实际情况选择负载预测模块使用的模型。

[0018] 所述特征工程包括调用数据的复杂程度和历史数据的数量,可选择的模型包括ARIMA数学模型和基于注意力机制的深度学习模型。

[0019] 步骤202:将历史数据划分为测试集和训练集,对模型进行训练,确定其最优参数;同时选择执行每次预测时,使用历史调用率数据数量的最优值,即历史数据窗口长度。

[0020] 步骤203:将训练好的模型用于负载预测:在决策周期内,通过Prometheus相关接口,获得历史数据窗口的调用率数据,结合训练好的模型获得下一个周期调用率的预测值。

[0021] 步骤三、将预测结果输入强化学习智能体,对强化学习智能体进行训练,通过训练后的策略网络输出扩缩决策。

[0022] 通过M个episode(迭代周期),每个episode执行N个step对强化学习智能体进行训练,具体步骤如下:

[0023] 步骤301:初始时,获取智能体的状态空间表征和动作空间表征,并随机初始化策略网络θ_0和状态估值网络φ_0;

[0024] 状态空间表征为S,包括下一个周期调用率预测值和当前微服务副本数量;

[0025] 动作空间表征为$\mathcal{A}$,$\mathcal{A}=\{\chi_0,\chi_1,\chi_2\}$,动作$\chi_0$表示将微服务副本减少一个,动作$\chi_1$表示不对微服务副本数量进行操作,动作$\chi_2$表示将微服务副本增加一个。

[0026] 步骤302,从k=0episode开始,进行该episode中从t=0开始每个时间周期的循环迭代;

[0027] 步骤303,从t=0的周期开始,根据负载预测模块获取下一个周期调用率预测值,并结合当前微服务副本数量,作为当前的状态s_t,通过策略π_{θ_k}获得智能体的动作a_t;

[0028] 步骤304,当智能体对微服务施加动作a_t后,得到此时微服务的平均响应时间,并结合用户指定的期望响应时间lim,设计奖励函数r;

[0029] 奖励函数r包括动作奖励函数r_{base}和合法性奖励函数r_{legal},公式为:

[0030] $r=r_{base}+r_{legal}$

[0031] 其中,动作奖励函数r_{base}表示为:

[0032] $$r_{base}=\begin{cases} a_t=\chi_0 \begin{cases} rt_{t+1}>lim & -4K \\ rt_{t+1}\le lim & 2K \end{cases} \\ a_t=\chi_1 \begin{cases} rt_{t+1}>lim & -4K \\ rt_{t+1}\le lim & 0 \end{cases} \\ a_t=\chi_2 \begin{cases} rt_{t+1}>lim & 4K \\ rt_{t+1}\le lim & -3K \end{cases} \end{cases}$$

[0033] 其中,rt_{t+1}表示当前动作施加到系统环境后,下一个周期微服务的平均响应时间;K代表奖励的基本数值,以常数的形式出现。

[0034] 动作奖励函数r_{base}的具体含义为:在智能体执行动作a_t后,当微服务的响应时间rt_{t+1}超过阈值时,若执行的动作$a_t\in\{\chi_0,\chi_1\}$,则获得惩罚-4K;若执行动作χ_2,获得奖励4K;当响应时间rt_{t+1}满足阈值时,当执行动作χ_0时,给予奖励2K;当执行动作χ_1时,不给予奖励或惩罚;当执行动作χ_2时,给与惩罚-3K。

[0035] 当智能体决策使用动作χ_0或χ_2时,判断微服务副本数量是否合法,得到合法性奖励函数r_{legal}表示为:

[0036] $$r_{legal}=\begin{cases} -4K & a_t \text{ 执行后进入非法状态} \\ 0 & a_t \text{ 执行后保持合法状态} \end{cases}$$

[0037] 步骤305,判断当前的时间周期是否满足t≤N,如果是,返回步骤303继续当前episode的迭代;否则,计算当前策略网络的优势函数$A^{\pi_{\theta_k}}$,并执行步骤306;

[0038] 对于当前策略网络θ_k驱动的策略π,即π_{θ_k},其优势函数A^{π}表示为

[0039] $A^{\pi}(s,a)=Q^{\pi}(s,a)-V^{\pi}(s)$

[0040] 其中,$Q^{\pi}(s,a)$为Bellman方程中的Q函数,表达在状态s中执行动作a后,按照策略π行动获得的期望回报;$Q^{\pi}(s,a)$为Bellman方程中的V函数,表达从状态s开始,按照策略π行动获得的期望回报。

[0041] 步骤306:根据优势函数,通过梯度上升的方法,对如下目标函数执行优化:

[0042] $$L(s,a,\theta_k,\theta)=\min\left(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}A^{\pi_{\theta_k}}(s,a),g(\epsilon,A^{\pi_{\theta_k}}(s,a))\right)$$

[0043] 其中,$g(\epsilon,A)=\begin{cases}(1+\epsilon)A & A\ge 0\\(1-\epsilon)A & A<0\end{cases}$,常数∈为策略优化函数的截断区间。$\theta_k$表示驱动策略网络的当前参数,θ表示执行目标函数梯度上升时,网络优化的方向。

[0044] 步骤307:判断是否满足k≤M,如果是,返回步骤302继续迭代;否则,获得了最终策略网络的最终权重θ,进入步骤308;

[0045] 步骤308:策略网络在使用策略π_θ时,将下一个周期调用率预测值和当前微服务副本数量作为输入,输出扩缩决策。

[0046] 步骤四、水平扩缩控制器按照强化学习智能体输出的扩缩决策,对相应微服务副本数量进行调整,增加、减少或保持其副本数量。

[0047] 本发明的优点在于:

[0048] 1)本发明中添加了强化学习预测模块,可以帮助智能体更好地决策。如当预测到

某个微服务访问量将大幅上涨,有很大概率影响用户请求的响应时间的情况下,智能体可以主动式、提前地执行扩容动作,进而避免恶化发生时,用户请求响应时间短时间内恶化的抖动现象。

[0049] 2)本发明通过使用强化学习算法控制微服务的副本数量,智能体策略的透明性、动态性和自适应性在复杂多变的云环境中有其固有优势。

附图说明

[0050] 图1为本发明中基于时序预测和强化学习的微服务管理方法的框架图;

[0051] 图2为本发明基于负载预测和强化学习的微服务管理方法的流程图;

[0052] 图3为本发明中对强化学习智能体进行训练的流程图。

具体实施方式

[0053] 下面将结合附图和实施例对本发明所述方案作进一步描述说明。此处所描述的具体实施方案仅用于说明和解释本发明,并不用于限制本发明。

[0054] 本发明提出了一种基于负载预测和强化学习的微服务管理框架,如图1所示,微服务平台中管理着多个微服务,并有用户外部请求;Istio和Jaeger负责收集各个微服务地负载,即调用情况,并将指标汇总至Prometheus时序数据库;负载预测模块通过Prometheus提取数据,并对调用率进行预测;强化学习智能体汇总响应时间等数据后,执行扩缩决策。

[0055] 基于以上框架,本发明提出一种基于负载预测和强化学习的微服务管理方法,如图2所示,具体步骤如下:

[0056] 步骤一、对于每一个时间周期,通过Istio和Jaeger收集平台中各个微服务的调用次数和平均响应时间作为指标,并按结构化的形式,存储至时序数据库Prometheus中。

[0057] 指标的结构化形式包括名称name、标签label、时间戳timestamp和相应指标数值value,并将标签用于对同一指标在不同情况下的分类,具体表示为name{label=`标签值`}value timestamp。

[0058] 步骤二、负载预测模块通过Prometheus以时间序列的形式获取各个微服务调用率数据,并对下一个时间周期的调用率进行预测。

[0059] 负载预测的步骤如下:

[0060] 步骤201:对当前系统中微服务的调用数据进行特征工程,根据实际情况选择负载预测模块使用的模型。

[0061] 对于调用数据周期明显,历史数据较少的应用场景,可选择ARIMA等数学模型;对于历史数据多,调用数据模式复杂的应用场景,可选择基于注意力机制的深度学习模型。

[0062] 步骤202:使用历史数据对模型执行训练,并通过独立划分的测试集,确定模型最优参数。其中,除模型自身权重外,同时选择执行每次预测时,使用历史调用率数据数量的最优值,即历史数据窗口长度。

[0063] 步骤203:使用模型执行预测时,在决策周期内,通过Prometheus相关接口,获得历史数据窗口的调用率数据,结合模型获得下一个周期调用率的预测值。

[0064] 步骤三、将预测结果输入强化学习智能体,对强化学习智能体进行训练,通过训练后的策略输出操作微服务副本数量的扩缩决策。

[0065] 通过M个episode(迭代周期),每个episode执行N个step对强化学习智能体进行训练,如图3所示,具体步骤如下:

[0066] 步骤301:初始时,获取智能体的状态空间表征和动作空间表征,并随机初始化策略网络θ_0和状态估值网络φ_0;

[0067] 状态空间表征为S,包括负载预测模块输出的下一个周期调用率预测值和当前微服务副本数量;

[0068] 动作空间表征为$\mathcal{A}$,$\mathcal{A}=\{\chi_0,\chi_1,\chi_2\}$,动作$\chi_0$表示将微服务副本减少一个,动作$\chi_1$表示不对微服务副本数量进行操作,动作$\chi_2$表示将微服务副本增加一个。

[0069] 步骤302,从k=0episode开始,进行该episode中从t=0开始每个时间周期的循环迭代;

[0070] 步骤303,从t=0的周期开始,根据负载预测模块获取下一个周期调用率预测值,并结合当前微服务副本数量,作为当前的状态s_t,通过策略π_{θ_k}获得智能体的动作a_t;

[0071] 步骤304,当智能体对微服务施加动作a_t后,得到此时微服务的平均响应时间,并结合用户指定的期望响应时间lim,设计奖励函数r;

[0072] 奖励函数r包括动作奖励函数r_{base}和合法性奖励函数r_{legal},公式为:

[0073] $r=r_{base}+r_{legal}$

[0074] 对于奖励函数的设计来说,主要考虑两个方面:

[0075] 第一个方面是优先保证处理的平均响应时间小于用户期望响应时间,其次尽可能节约资源。在当前副本数量不足,将导致响应时间超出阈值的情况下,鼓励智能体增加副本数量,选择动作χ_2;在当前副本数量恰能保证阈值时,通过奖励函数的设计,避免智能体连续调整副本数量带来的震荡;当副本数量超过满足响应时间的阈值时,鼓励智能体减少副本数量,选择动作χ_0。

[0076] 动作奖励函数r_{base}表示为:

[0077] $$r_{base}=\begin{cases} a_t=\chi_0 \begin{cases} rt_{t+1}>lim & -4K \\ rt_{t+1}\le lim & 2K \end{cases} \\ a_t=\chi_1 \begin{cases} rt_{t+1}>lim & -4K \\ rt_{t+1}\le lim & 0 \end{cases} \\ a_t=\chi_2 \begin{cases} rt_{t+1}>lim & 4K \\ rt_{t+1}\le lim & -3K \end{cases} \end{cases}$$

[0078] 其中,rt_{t+1}表示当前动作施加到系统环境后,下一个周期微服务的平均响应时间;K代表奖励的基本数值,以常数的形式出现。

[0079] 执行动作a_t后,当响应时间rt_{t+1}超过阈值时,若执行的动作$a_t\in\{\chi_0,\chi_1\}$,则获得奖励-4K,因为该动作不应在当前状态中被选择执行,故使用负奖励对智能体进行惩罚;若执行动作χ_2,虽然响应时间rt_{t+1}超过阈值,但该操作完全符合当前状态下的期望操作,故获得奖励4K。当响应时间rt_{t+1}满足阈值时,奖励的设计稍复杂一些:当执行动作χ_0时,减少了副本数量,节约了集群的资源,给予奖励2K;当执行动作χ_1时,不改变副本数量,不给予奖励或惩罚;当执行动作χ_2时,增加了副本数量,给与惩罚-3K。

[0080] 奖励函数核心部分r_{base}的设计主要表达如下的趋势与期望:

[0081] 1.在当前副本数量不足,将导致响应时间超出阈值的情况下,指导智能体选择唯一能提供正奖励的动作χ_2。

[0082] 2.在当前副本数增加至恰好能保证阈值时,避免智能体连续多次错误动作导致的副本数量震荡。当副本数阈值达到时,若智能体执行先减(χ_0)后增(χ_2)的操作,将获得-4K-3K=-7K的期望奖励;若智能体执行先增(χ_2)后减(χ_0)的操作,将获得-3K+2K=-K的期望奖励。可见在这种状态下,更鼓励智能体选择操作χ_1保持副本数量的不变,由此保证了智能体动作的有效性。

[0083] 3.在当前副本数量超过满足响应时间的阈值时,指导智能体选择唯一能提供正奖励的动作χ_0。

[0084] 第二个方面是动作的合法性。当智能体决策使用动作χ_0或χ_2时,可能导致微服务副本数量非法,对其进行惩罚,此时定义奖励函数的第二个部分r_{legal}表示为:

[0085] $$r_{legal}=\begin{cases}-4K & a_t \text{ 执行后进入非法状态}\\ 0 & a_t \text{ 执行后保持合法状态}\end{cases}$$

[0086] 步骤305,判断当前的时间周期是否满足t≤N,如果是,返回步骤303继续当前episode的迭代;否则,计算当前策略网络的优势函数$A^{\pi_{\theta_k}}$,并执行步骤306;

[0087] 对于当前策略网络θ_k驱动的策略π,即π_{θ_k},其优势函数表示为

[0088] $$A^{\pi_{\theta_k}}(s,a)=Q^{\pi}(s,a)-V^{\pi}(s)$$

[0089] 其中,$Q^{\pi}(s,a)$为Bellman方程中的Q函数,表达在状态s中执行动作a后,按照策略π行动获得的期望回报;$Q^{\pi}(s,a)$为Bellman方程中的V函数,表达从状态s开始,按照策略π行动获得的期望回报。

[0090] 步骤306:根据优势函数,通过梯度上升的方法,对如下目标函数执行优化:

[0091] $$L(s,a,\theta_k,\theta)=\min(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}A^{\pi_{\theta_k}}(s,a),g(\epsilon,A^{\pi_{\theta_k}}(s,a)))$$

[0092] 其中,$g(\epsilon,A)=\begin{cases}(1+\epsilon)A & A\geq 0\\ (1-\epsilon)A & A<0\end{cases}$,常数∈为策略优化函数的截断区间,通过该常数,可保证策略优化的稳定性。θ_k表示驱动策略网络的当前参数,θ表示执行目标函数梯度上升时,网络优化的方向。

[0093] 步骤307:判断是否满足k≤M,如果是,返回步骤302继续迭代;否则,获得了最终策略网络的最终权重θ,进入步骤308;

[0094] 步骤308:策略网络在使用策略π_θ时,将下一个周期调用率预测值和当前微服务副本数量作为输入,输出调节微服务副本数量的扩缩决策。

[0095] 步骤四、水平扩缩控制器按照强化学习智能体输出的扩缩决策,对相应微服务副本数量进行调整,增加、减少或保持其副本数量。

[0096] 上述内容为本发明一个具体实施例的描述,上述实施例是示例性的,不作为对本发明的限制。

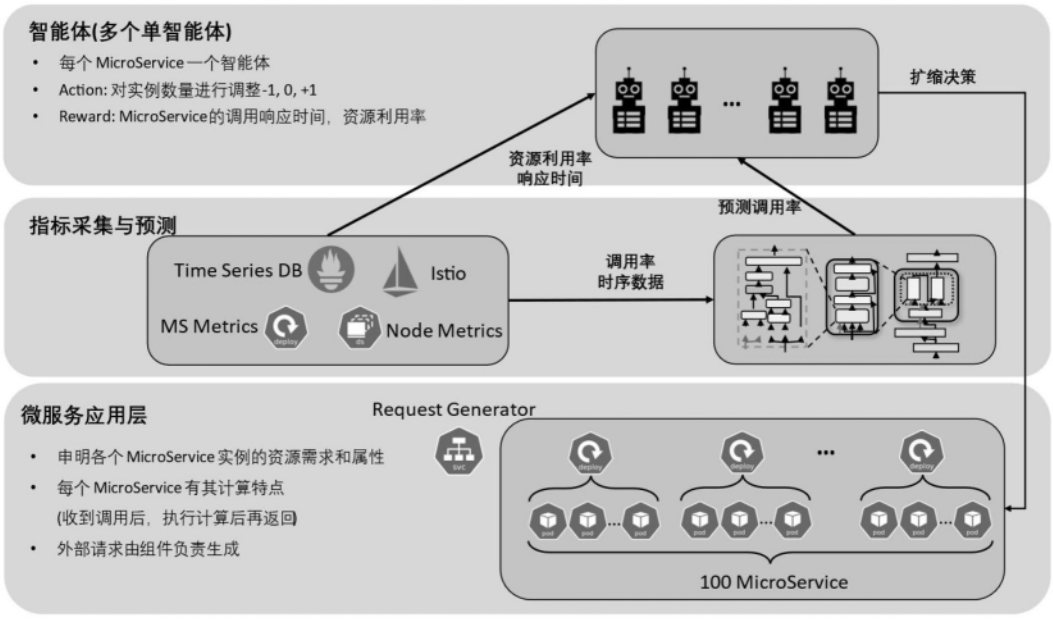

智能体(多个单智能体)
每个 MicroService 一个智能体
Action: 对实例数量进行调整-1, 0, +1
Reward: MicroService的调用响应时间，资源利用率
扩缩决策
资源利用率
响应时间
预测调用率
指标采集与预测
Time Series DB
Istio
MS Metrics
Node Metrics
调用率
时序数据
微服务应用层
Request Generator
申明各个 MicroService 实例的资源需求和属性
每个 MicroService 有其计算特点
(收到调用后，执行计算后再返回)
外部请求由组件负责生成
100 MicroService

图1

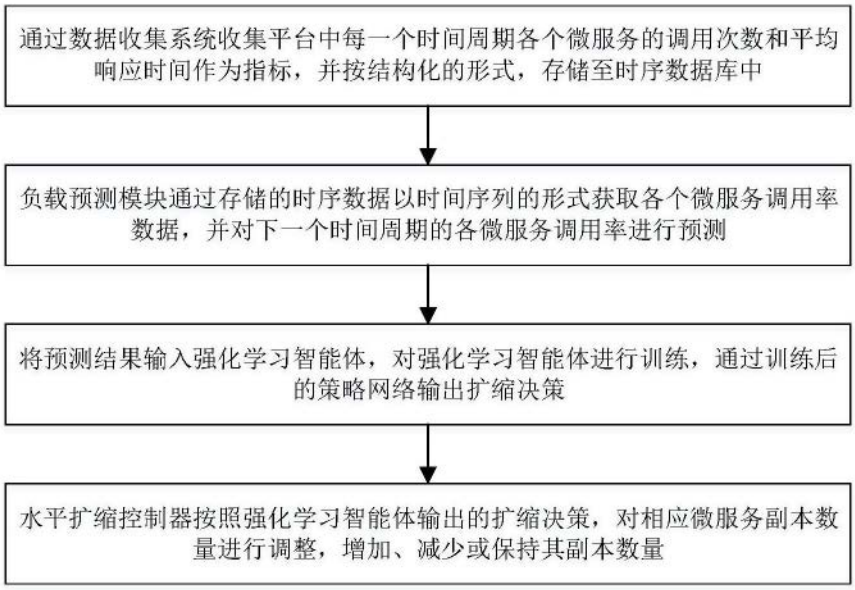

通过数据收集系统收集平台中每一个时间周期各个微服务的调用次数和平均响应时间作为指标，并按结构化的形式，存储至时序数据库中
负载预测模块通过存储的时序数据以时间序列的形式获取各个微服务调用率数据，并对下一个时间周期的各微服务调用率进行预测
将预测结果输入强化学习智能体，对强化学习智能体进行训练，通过训练后的策略网络输出扩缩决策
水平扩缩控制器按照强化学习智能体输出的扩缩决策，对相应微服务副本数量进行调整，增加、减少或保持其副本数量

图2

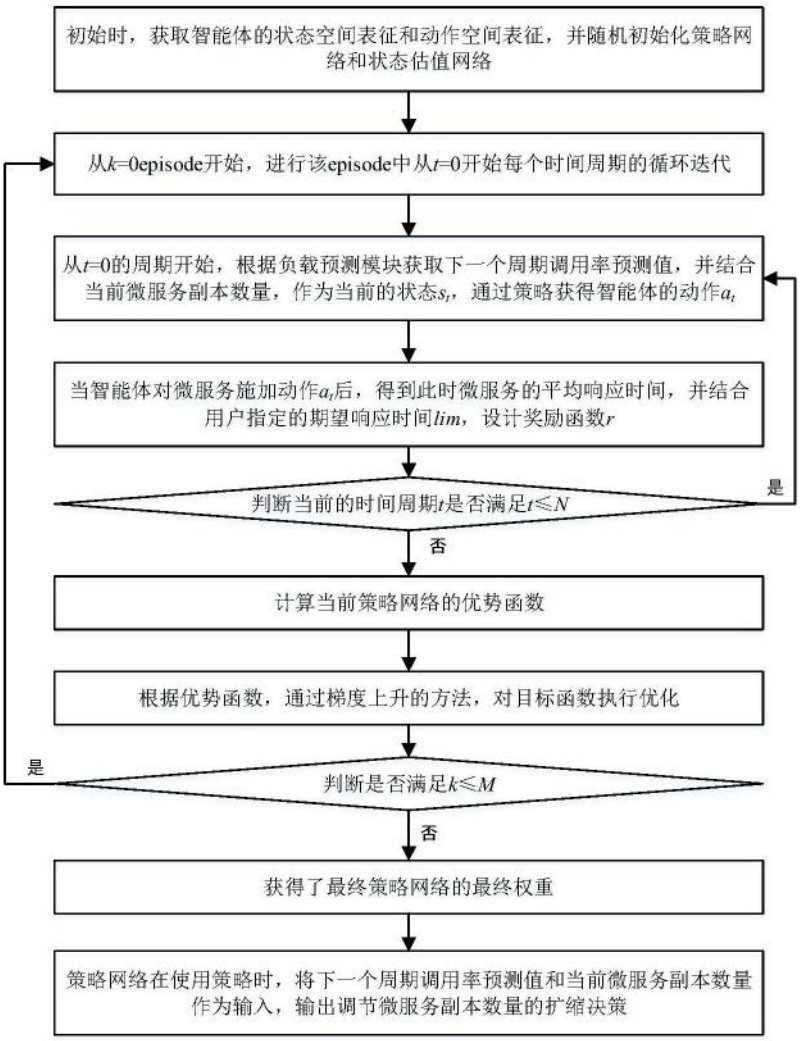

初始时，获取智能体的状态空间表征和动作空间表征，并随机初始化策略网络和状态估值网络
从k=0episode开始，进行该episode中从t=0开始每个时间周期的循环迭代
从t=0的周期开始，根据负载预测模块获取下一个周期调用率预测值，并结合当前微服务副本数量，作为当前的状态s_t，通过策略获得智能体的动作a_t
当智能体对微服务施加动作a_t后，得到此时微服务的平均响应时间，并结合用户指定的期望响应时间lim，设计奖励函数r
判断当前的时间周期t是否满足$t \leqslant N$
是
否
计算当前策略网络的优势函数
根据优势函数，通过梯度上升的方法，对目标函数执行优化
判断是否满足$k \leqslant M$
是
否
获得了最终策略网络的最终权重
策略网络在使用策略时，将下一个周期调用率预测值和当前微服务副本数量作为输入，输出调节微服务副本数量的扩缩决策

图3