

云训练中基于自适应副本策略的容错研究

陈志佳, 朱元昌, 邸彦强, 冯少冲

(军械工程学院 电子与光学工程系, 河北 石家庄 050003)

摘 要: 为提升系统容错能力, 降低容错开销, 提出一种适应于云训练系统的自适应副本容错策略. 首先分析了云训练系统内涵以及容错结构. 通过分析确定自适应副本策略待解决的三个问题: 节点选择, 副本数量以及位置分布. 引入节点活跃度评价节点是否需要生成副本; 结合容错需求, 得到需要生成的副本数量; 通过加权升序匹配算法, 实现位置分布的确定. 实验中, 引入容错度等概念对策略进行评价, 数据表明自适应副本容错策略可以有效保证云训练的容错能力, 降低容错开销.

关键词: 云训练; 容错; 自适应; 副本策略; 加权升序匹配

中图分类号: TP39

文献标识码: A

文章编号: 1000-7180(2016)02-0039-03

Research on Fault Tolerance Based on Self-Adaptive Backup Strategy in Cloud Training System

CHEN Zhi-jia, ZHU Yuan-chang, DI Yan-qiang, FENG Shao-chong

(Department of Electronic and Optics, Ordnance Engineering College, Shijiazhuang 050003, China)

Abstract: To improve the fault tolerance capability and decrease the fault tolerance overhead, a self-adaptive backup strategy suitable for cloud training system is proposed. The connotation and fault tolerance architecture of cloud training are analyzed. Three problems of the self-adaptive backup strategy are analyzed: node select, the number of backups and the location distribution. The node activity degree is introduced to evaluate the backup demands of nodes. The backup number is obtained combined with fault tolerance demands. By weighted ascending matching select, the location distribution is determined. Fault tolerance degree is introduced to the experiments. The results show that the proposed self-adaptive backup fault tolerance strategy can effectively improve the fault tolerance capability and decrease the overhead of fault tolerance.

Key words: cloud training; fault tolerance; self-adaptive; backup strategy; weighted ascending match

1 引言

IaaS (Infrastructure as a Service, 基础设施即服务) 模式云计算^[1]为装备模拟训练向网络化方向发展提供了借鉴. 借鉴 IaaS 云计算模式, 提出一种 IaaS 模式“云训练”(下文简称“云训练”), 通过对训练资源的虚拟化以及虚拟资源的网络化共享, 为用户提供独立的虚拟训练设备, 使得多个用户可以通过网络访问训练中心, 完成维修、操作等训练任务, 为实现大规模装备网络化模拟训练提供

新的思路.

随着云训练系统运行时间的推进以及训练成员的增加, 系统的可靠性逐渐下降, 导致服务可用性也逐渐下降. 因而有效的预防和处理训练资源不足和节点崩溃等情况, 是提高系统容错能力、保证服务高可用性的关键, 云计算中的虚拟化技术^[2]为解决上述问题的可行性提供了保障. 为了提高云计算系统与仿真系统的容错性, 有学者做了大量的研究工作. Daeyong Jung 等人^[3]针对 Amazon EC2 的特殊服务 Spot Instance 出现的不可靠

收稿日期: 2015-04-13; 修回日期: 2015-05-29

基金项目: 装备预研基金(9140A04030214JB34001)

性提出了一种通过虚拟机 (VM) 迁移来进行容错的方案,但针对性较强,主要针对特殊的云计算方式,即 Amazon 的 Spot Instance 来实施.文献[4]采用虚拟化技术和负载均衡技术,对故障节点、正常节点和提交的任务进行重新规划部署,以提升任务执行成功率. Ganesan Radhakrishnan^[5]通过迁移技术将节点资源耗尽的危害降到最低程度,通过计算节点的数据吞吐率和任务的等待时间来决定任务和服务节点之间的定向关系,一定程度上提升了系统可用性.

在前述研究基础上,同时借鉴云计算应用中的典型容错策略^[6],提出一种适应于虚拟化环境的云训练系统自适应副本容错方法.

2 云训练内涵与容错结构分析

云训练的整体基本结构为“中心—终端”结构,如图1所示.中心包含了模拟训练系统的基础设施资源、模型资源及其他软件资源,通过虚拟化技术,封装为以虚拟机为单位的模拟训练节点,通过远程传输协议为训练用户提供服务.终端按照功能可分为两大类,一类为训练终端,一类为开发终端,在硬件方面均可采用瘦客户端,减少终端基础设施部署与维护成本,提高开发训练效率.中心和终端通过网络进行互联.终端用户采用远程传输协议与云中心进行数据交互,开发终端将开发的训练项目发布到云中心,中心的监控与调度服务将训练项目封装到虚拟机中.训练用户提出所需机器配置的申请,选择不同的装备科目模型进行训练.

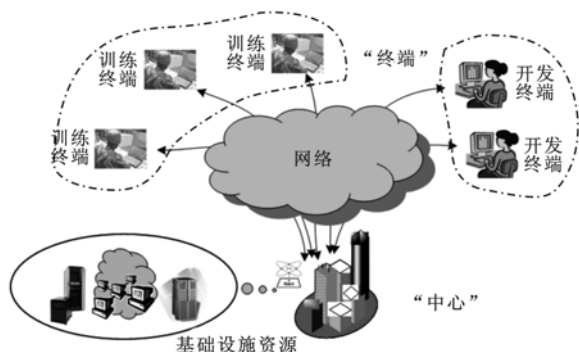


图1 IaaS 模式云训练结构图

根据云训练中的故障类型,一般可划分为两大类^[7]:应用层故障和资源层故障.应用层故障是指云训练仿真中间件和应用出现的故障;资源层故障是指在仿真应用运行过程中用到的资源,尤指物理资源或虚拟物理资源出现的故障,比如宕机、网络故

障等.由于应用层故障根据应用的不同,其故障原因以及影响等各不相同,因而对应用层容错的研究不具有普适性;相反,由于云训练系统资源的普遍性及通用性,对资源层容错的研究对保证整个云训练系统的可靠性和服务可用性更具有普适性,因此本文主要针对资源层故障开展研究.

带有容错系统的云训练中心结构分为三层,分别是物理资源层、虚拟化层、容错管理层和服务层,如图2所示.在物理资源层,分别在不同位置构建多个数据中心,也即资源池,为创建虚拟机提供基础硬件设施.虚拟化层包含虚拟化软件连同其生成的虚拟资源,将这些资源按需组合为用于不同训练任务的虚拟机.虚拟机可以像物理机一样为用户提供训练服务.虚拟机底层管理对上层用户是透明的,根据用户需求实现相关的装备训练服务.同时,虚拟机结构的易扩展性以及管理的灵活性为云训练容错服务提供了良好的支持.在容错层,分析 VM 备份的三个待解决问题:节点选择、副本数量以及位置分布,通过数学模型计算及自适应匹配方法,实现系统可靠性和容错能力的提升.通过服务层,云训练系统为用户提供包括装备模拟训练,装备操作视频演示和试验推演模型解算等服务.

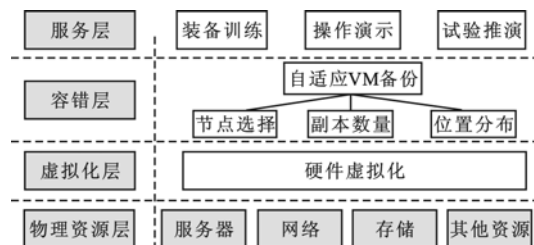


图2 带有容错系统的云训练中心结构图

3 自适应副本容错策略

虚拟化技术为副本冗余提供了必要条件,借助虚拟化技术,可以方便地为训练系统提供备份虚拟机,避免单个训练节点故障导致的系统崩溃,达到提升容错效果的目的.图3所示为副本策略中训练节点与备份节点之间的关系,通过物理隔离保证当训练节点出现故障后不会对备份节点产生灾难性影响.设定运行时心跳时间,训练节点定时向备份节点发送备份更新的信息,备份节点会向训练节点反馈心跳信息,如步骤①所示.如果在规定的心跳周期内备份节点没有收到训练节点的更新信息,则认为训练节点已经出现故障.此时,将备份节点设置为系统中的训练节点并更新其状态,并继续训练过程,如步

骤②所示.系统尝试在原训练节点所在服务器上创建备份节点,如果该故障服务器无法启动,则被标注为“死机”,任何存储于“死机”内的数据状态均变为无效.为了保证可靠的容错性能,系统会在与“死机”相近的主机中创建备份节点,开始新一轮的备份,并与当前训练节点建立心跳信息交互,如步骤③所示.

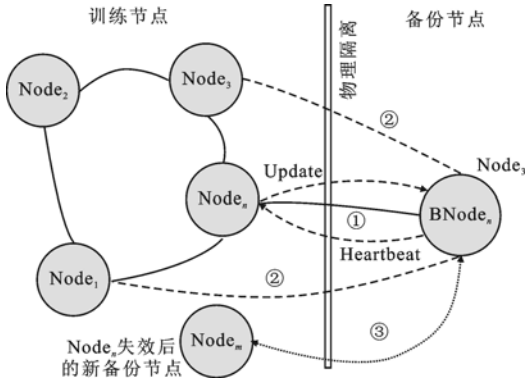


图3 副本策略中训练节点与备份节点关系示意

备份容错策略主要考虑以下问题:(1)明确待复制节点的选择以保证容错效果;(2)确定副本数量以降低系统维护的代价;(3)选择副本位置,以降低由于分布不合理造成的带宽资源占用和数据交互延时.

3.1 节点选择

为了提高效率降低开销,参考帕累托定律及时间局部性定律^[6],根据用户数据交互频率与数据吞吐量来确定节点的活跃度,当活跃度超过一定的阈值后,则会触发备份操作.

为计算节点 f_k 的活跃度 d_k ,首先引入遗忘参数 $\xi(t_s, t_p)$,它是该节点从初始时刻 t_s 到当前时刻 t_p 的时间间隔以及交互数据量的负指数函数.遗忘参数 ξ 越小,表示节点活跃度越高,反之则活跃度低.

$$\xi(t_s, t_p) = e^{-\lambda \cdot \frac{k}{t_p - t_s}} \quad (1)$$

令 $\Delta t = t_p - t_s$, k 值为训练节点 f_k 在 Δt 时间内与终端桌面交互数据量,决定了活跃度曲线在 Δt 时间内的衰减率.当 $\frac{k}{\Delta t} \rightarrow 0$ 时, $\xi \rightarrow 1$;当 $\frac{k}{\Delta t} \rightarrow \infty$ 时, $\xi \rightarrow 0$.

初始时刻 t_s 到当前时刻 t_p 的间隔内,节点 f_k 的活跃度 d_k 为

$$d_k = \sum_{t_s}^{t_p} [n_k(t_i, t_{i+1}) \cdot (1 - \xi(t_i, t_p))] \quad (2)$$

式中, $n_k(t_i, t_{i+1})$ 为时间间隔 $[t_i, t_{i+1})$ 内数据的交互次数.由式(2)可知,节点数据交互量越大,遗忘参

数越小,则节点活跃度越高.

某个节点是否进行复制由复制参数 φ_k 确定,定义为节点的活跃度与节点重要程度权重 w_k 的乘积,即:

$$\varphi_k = d_k \cdot w_k \quad (3)$$

如果该参数大于某个设定的阈值,则认为该节点是一个关键节点,会触发节点的复制操作;反之则认为该节点并非系统的关键节点,可不进行备份.

3.2 副本数量

系统运行过程中,容错需求会动态发生变化.即在需求合理的提高或降低时,需要新增的副本数量 $n_k(\text{inc})$ 为

$$n_k(\text{inc}) = n_k(\text{new}) - n_k(\text{old}) \quad (4)$$

式中, $n_k(\text{new})$ 为新的副本数量, $n_k(\text{old})$ 为旧的副本数量. $n_k(\text{inc})$ 为正值说明需要新增副本,若为负值则说明副本数量应该减少.对训练节点 f_k 来说,新增副本后其可靠性 μ_{new} 与源节点可靠性 μ 之间存在如下关系:

$$1 - \mu_{\text{new}} = (1 - \mu)^{n_k(\text{new})} \quad (5)$$

同时,可靠性 μ_{new} 可以根据下式计算:

$$\mu_{\text{new}} = \mu_{\text{old}} + \alpha(1 - \mu_{\text{old}}) \quad (6)$$

式中, μ_{old} 表示节点 f_k 原有副本的可靠性, α 为副本数量调整参数,与复制参数有以下函数关系:

$$\alpha = \frac{\varphi_k}{\sum_{i=1}^{\text{num}_s} \varphi_i} \quad (7)$$

式中, num_s 是需要生成的副本数量.联立式(5),式(6)和式(7),可得:

$$1 - (1 - \mu)^{n_k(\text{new})} = \mu_{\text{old}} + \frac{\varphi_k}{\sum_{i=1}^{\text{num}_s} \varphi_i} \cdot (1 - \mu_{\text{old}}) \quad (8)$$

对上式进行变换,

$$n_k(\text{new}) = \frac{\ln(1 - \mu_{\text{old}}) + \frac{\varphi_k}{\sum_{i=1}^{\text{num}_s} \varphi_i} \cdot (1 - \mu_{\text{old}})}{\ln(1 - \mu)} \quad (9)$$

又,

$$1 - \mu_{\text{old}} = (1 - \mu)^{n_k(\text{old})} \quad (10)$$

联立式(9)和式(10),可得:

$$n_k(\text{new}) = \frac{\ln(1 - \mu)^{n_k(\text{old})} + \frac{\varphi_k}{\sum_{i=1}^{\text{num}_s} \varphi_i} \cdot ((1 - \mu)^{n_k(\text{old})})}{\ln(1 - \mu)} \quad (11)$$

代入式(4),新增副本数量应为

$$n_k(\text{inc}) = \frac{\varphi_k \cdot (1 - \mu)^{n_k(\text{old})}}{\ln(1 - \mu) \cdot \sum_{i=1}^{\text{num}_i} \varphi_i} \quad (12)$$

3.3 位置分布

副本放置遵循以下几点原则:(1) 降低源节点与副本节点之间的传输距离,以降低开销;(2) 备份节点所在服务器资源充足,维持备份节点的正常运行;(3) 尽量避免多个副本位于同一台服务器,以降低副本同时失效的概率. 节点间距离采用经过的网络交换次数衡量,即 $d_{ij} = \sum c_k$, c_k 为节点 i 至节点 j 之间经过的网络交换次数.

为满足原则(2)和原则(3),根据实时监控得到源节点虚拟机资源使用情况,构造资源需求向量 $\mathbf{u} = \{u_{\text{CPU}}, u_{\text{Mem}}, u_{\text{Net}}, u_{\text{GPU}}\}$. 同时,构造各放置目标服务器资源剩余量向量 $\mathbf{r} = \{r_{\text{CPU}}, r_{\text{Mem}}, r_{\text{Net}}, r_{\text{GPU}}\}$. 对比目标服务器的资源剩余量 r_j 与源节点虚拟机资源需求最大值 u_i ,以保证副本虚拟机放置到该服务器不会出现过载的情况^[8]. 对于所有向量元素,如果均有对应的 $r_i > u_i$,则标记该服务器为待选目标服务器. 若存在多个待选目标服务器,采用加权升序匹配方法确定副本位置.

首先定义虚拟机离散度 ρ 为虚拟机多个副本在服务器中的分布分散程度. 其中虚拟机 i 的副本在服务器中的离散度 ρ_i 表示为

$$\rho_i = \frac{N_i}{\sum_i M_{N_i}} \times 100\% \quad (13)$$

式中, N_i 为节点 i 的副本数量, M_{N_i} 为这些副本所占据的服务器数量. 当新增一个副本后,该副本所在位置不同会对离散度产生影响. 对于新增副本,按照其所在不同位置计算节点间距离和离散度,分别构造距离向量和离散度向量. 根据容错需求,定义位置决策量 s_{ij} ,它是距离向量和离散度向量中对应元素的加权和,即:

$$s_{ij} = \mu \cdot d_{ij} + (1 - \mu) \cdot \frac{1}{\rho_i}.$$

对决策量进行升序排列,则第一个元素即为副本综合最佳放置位置. 算法伪代码如下所示.

算法 1 副本位置分布决策算法

输入: 节点间距离 d_{ij} ; 节点资源需求向量 $\mathbf{u}$; 各目标服务器资源剩余向量 $\mathbf{r}$;

输出: 副本放置位置 s

for $i = 1$ to NoVM //NoVM 为待创建副本总数

for $j = 1$ to NoServ //NoServ 为服务器总数

比较 r_j 和 u_i 中的相应元素;

if $r_j > u_i$

标记该服务器为待选副本位置;

end if

end for

for $j = 1$ to NoTS // NoTS 为待选位置总数

计算副本放置在该位置处的离散度 ρ ;

计算副本位置决策量 s_{ij} ;

对决策量进行升序排序;

end for

选择排序后的首个决策量作为输出值;

end for

4 实验与分析

在仿真云平台 CloudSim^[9] 下对容错优化算法进行验证. 创建虚拟机的配置如表 1 所示. 训练用户任务数量从 50 ~ 3 200 个不等, 参数如表 2 所示.

表 1 CloudSim 环境下的虚拟机配置参数

PEs	MIPS	RAM	BW	存储
1	1 000	512	1 000	10 000

表 2 CloudSim 环境下的仿真任务参数

长度	文件大小	数量
1 000	300	50 ~ 3 200

引入容错度 τ 评价算法的容错能力, 定义为成功执行任务的数量 N_c 与所有提交任务数量 N_s 的比值, 即:

$$\tau = \frac{N_c}{N_s} \times 100\%.$$

图 4 所示为采用不同容错策略时的系统容错度. 横轴表示训练任务数量, 从 50 个任务增加至 3 200 个任务后, 本文介绍的自适应副本容错策略利用虚拟化技术, 通过自适应资源冗余佐证了系统高可靠性, 能够使系统容错度保持在 80% 之上.

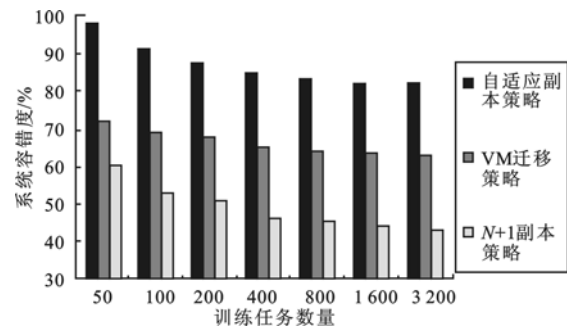


图 4 容错度对比

图 5 所示为采用不同容错策略时的物理资源开

销.横轴所示为 VM 数量,纵轴表示硬件资源开销.由图可知,自适应副本策略有效降低了物理资源开销.

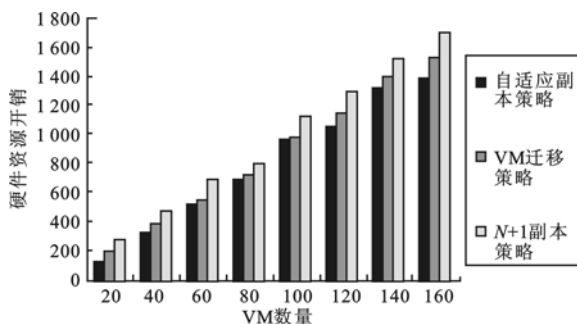


图5 容错资源开销对比

由于虚拟机迁移需要不断复制系统内存状态,其迁移和启动都需要一定时间,因而其时间开销较大.采用副本策略时,由于多个副本与源节点一样执行相同的应用,因而当源节点崩溃后,副本节点可以立即作为训练节点继续执行,因而其时间开销较小.特别的,采用自适应副本策略时,可以有效控制副本数量,当需求增大后,增加副本数量;需求降低后,则减少副本数量,有效降低容错的时间开销.时间开销比较结果如图6所示.

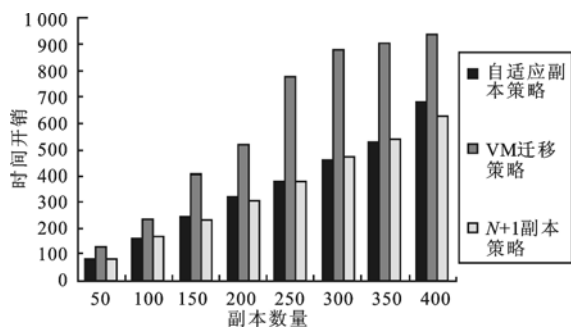


图6 容错时间开销对比

5 结束语

在传统容错方法的基础上,结合虚拟化技术的优势,提出了一种自适应副本容错策略,实现了云训练系统容错能力的优化.研究了自适应副本策略中的节点选择、副本数量以及位置分布等问题,并提出了解决方法.通过上述方法实现高效可靠的云训练系统容错.后续研究重点包括两部分:(1)多副本在保证系统容错能力的同时,加剧了资源的竞争与冲突,后续将对如何降低资源竞争进行进一步研究;

(2)研究副本策略与 VM 迁移策略的结合方法以及二者共同作用下的系统容错效果.

参考文献:

- [1] Qian Ling, Luo Zhiguo, Du Yujian, et al. Cloud computing: an overview[J]. Lecture Notes in Computer Science, 2009, 5931(1):626-631.
- [2] Eva Pajorová, Ladislav Hluchý. Complicated simulation visualization based on grid and cloud computing[C]// The 7th International Conference on Cooperative Design, Visualization & Engineering. Spain, Calvia, 2010: 211-217.
- [3] Daeyong Jung, SungHo Chin, Kwang Sik Chung, et al. VM migration for fault tolerance in spot instance based cloud computing [C]// Grid and Pervasive Computing, Korea, Seoul, 2013:142-151.
- [4] Pranesh Das. Virtualization and fault tolerance in cloud computing[D]. Rourkela: National Institute of Technology Rourkela, 2013.
- [5] Ganesan Radhakrishnan. Adaptive application scaling for improving fault-tolerance and availability in the cloud [J]. Bell Labs Technical Journal, 2012, 17(2):5-14.
- [6] Dawei Sun, Guiran Chang, Changsheng Miao. Analyzing, modeling and evaluating dynamic adaptive fault tolerance strategies in cloud computing environments [J]. Journal of Supercomputing, 2013, 66(1):193-228.
- [7] 刘云生. 大规模分布式仿真系统容错关键技术研究 [D]. 长沙:国防科学技术大学, 2006.
- [8] Anju Bala, Inderveer Chana. VM migration approach for autonomic fault tolerance in cloud computing[C]// International Conference on Grid and Cloud Computing and Applications. USA, Las Vegas, 2013:3-9.
- [9] Rodrigo N Calheiros, Rajiv Ranjan, Anton Beloglazov, et al. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms[J]. Journal of Software: Practice and Experience, 2010, 41(1):23-50.

作者简介:

陈志佳 男, (1986-), 博士研究生. 研究方向为云计算、云仿真. E-mail: youshenshui@163.com.

朱元昌 男, (1960-), 教授, 博士生导师. 研究方向为武器系统建模与仿真.