



基于 mean-variance 的服务集群负载均衡方法

包晓安,魏雪,陈磊,胡国亨,张娜

(浙江理工大学,浙江 杭州 310018)

摘要:大量并发请求任务进行分配时,负载调度机制是通过最小化响应时间及最大化节点利用率实现网络中节点的负载均衡,在基于遗传算法的负载均衡算法中,适应度函数设计对服务集群负载均衡效率产生重要的影响。对此提出了一种基于 mean-variance 的服务集群负载均衡方法对适应度函数进行优化,采用投资组合选择模型 mean-variance 进行最小化响应时间,以得到每个服务器资源利用率的权重,从而获得最优的分配组合,进而提高适应度函数的准确性和有效性。在不同服务环境下与其他模型进行比较,仿真结果表明,本文的负载均衡算法在节点利用率和响应时间方面使服务集群得到了更好的均衡。

关键词:负载均衡;mean-variance 模型;遗传算法;负载调度

中图分类号:TP393

文献标识码:A

doi: 10.11959/j.issn.1000-0801.2017027

Load balancing method of service cluster based on mean-variance

BAO Xiaolan, WEI Xue, CHEN Lei, HU Guoheng, ZHANG Na

Zhejiang Sci-Tech University, Hangzhou 310018, China

Abstract: When a large number of concurrent requests are allocated, the load scheduling mechanism is to achieve the load balancing of nodes in the network by minimizing the response time and maximizing the utilization ratio of nodes. In the load balancing algorithm based on genetic algorithm, the fitness function is designed to have an important influence on the load balancing efficiency. A service cluster load balancing method based on mean-variance was proposed to optimize the fitness function. The investment portfolio selection model mean-variance was used to minimize the response time, which was used to get the weight of each server's resource utilization, so as to obtain the optimal allocation combination. This method improves the accuracy and efficiency of the fitness function. Compared with other models in different service environment, the simulation results show that the load balancing algorithm makes the service cluster get a better balance performance in terms of node utilization and response time.

Key words: load balancing, mean-variance model, genetic algorithm, load scheduling

收稿日期:2016-11-12;修回日期:2017-01-09

基金项目:国家自然科学基金资助项目(No.61379036, No.61502430);国家自然科学基金委中丹合作项目(No.61361136002);浙江省重大科技专项重点工业项目(No.2014C01047);浙江理工大学“521 人才培养计划”基金资助项目

Foundation Items: The National Natural Science Foundation of China(No.61379036, No.61502430), China-Denmark Cooperation Program of the National Natural Science Foundation of China(No.61361136002), Major Science and Technology Projects of Zhejiang Province(No.2014C01047), 521 Talent Project of Zhejiang Sci-Tech University



1 引言

在服务器集群中,用户的请求需要经过负载均衡器将请求任务分配到后台的服务器进行处理。由于同一时间会有大量任务等待被分配,为了解决请求分配的流量拥塞及控制问题,负载均衡器在接收到来自内部或外部的资源请求时,根据服务器集群的负载情况通过均衡调度算法进行合理分配。负载均衡机制^[1,2]的目的是能够高效地为服务集群分配任务提供好的解决方案。服务集群在实现负载均衡这一领域,已经取得了很多研究成果^[3-5],但是这些传统的调度算法操作简单,不适于工作复杂的现实环境,并且点到点式的算法在搜索过程中往往会产生大量错误的峰值点,从而影响最佳结果的判定。Zomaya 和 Teh^[6]提出将遗传算法(genetic algorithm, GA)用到负载均衡策略上并且得到广泛应用。GA 的本质是一种求解问题的高度并行性全局搜索算法,能在搜索过程中自动获取和积累有关搜索空间的知识,并控制搜索过程以求得最优解。当服务集群有大量任务等待处理时负载均衡器会通过 GA 进行调度,其目标函数是最小化执行时间,充分利用节点的利用率并均衡网络负载。但简单遗传作为一种启发式搜索算法,寻优理论还不完善。有各种类型的负载均衡算法^[7,8],其中服务器的性能和已使用情况不同,则当前的节点利用率也将不同。但是,更多的负载均衡算法忽略了各个服务器应有的差异权重,针对节点利用率的计算是采用简单的求和,降低了适应值的准确性和有效性。

为了实现高效的基于遗传算法的负载均衡,特别是降低频繁访问节点的响应时间还需要进行研究。适应度函数是 GA 进行最优选择的关键步骤。如果提高了其有效性将在一定的资源利用率基础上节约响应时间。mean-variance 投资组合选择理论^[9]主要研究如何使金融资产进行合理配置与选择,使用证券收益方差度量风险,从而实现收益率最大化与风险最小化间的均衡,为投资者进行决策提供了指导。继而 mean-variance 投资组合选择理论还被用于应用层路由中,路由的多路径权值采用 mean-variance 模型,在求解约束条件下求解最优化问题而获得的。本文是基于 GA 对负载任务进行合理分配,在一定的资源利用率情况下最小化响应时间,mean-variance 模型适合适应度函数中二者的特殊关系。通过 mean-variance 模型改进传统适应度函数,以一定水平的资源利用率尽量缩短用户请求的等待时延,从而获得最优的分配组合,提高

用户体验。

2 负载均衡机制

2.1 量化负载

当大量用户访问网络时,不同服务所需的时间和所消耗的计算资源是千差万别的。其中请求服务的类型不同,当前网络带宽或服务器资源利用的情况不同等都是影响因素。例如,负载比较轻的请求或许只需要读一个 HTML 页面进行简单的计算,然而一些负载比较重的请求则需要计算密集的查询、数据库访问及很长的响应数据流,所以需要对不同的请求任务进行合理量化。根据服务器的日志文件进行分析,然后将其中涉及的请求文档进行分类^[10],通过不同文档类型在日志文件中所占的比重及不同的服务请求类型形成的不同负载值,对用户的请求任务进行负载值量化。

2.2 mean-variance 模型

投资组合优化问题作为现代金融学的一个核心课题^[11],主要研究如何对金融资产进行合理配置与选择,从而实现收益率最大化与风险最小化。

负载均衡策略的主要目标是在最小化响应时间的情况下最大化节点利用率。负载均衡问题符合马克维茨模型的几个条件:在一定的平均资源利用率基础上,期望的均衡时间最少;在一定的均衡时间上,期望的资源利用率最大;每一次的适应度函数的取值与前一次的资源利用率分布情况相关联;负载均衡的时间与平均资源利用率息息相关。因此,针对 m 个服务器的节点利用率及响应时间问题,本文采用 mean-variance 模型^[10,11]在约束条件下进行设计,通过方差度量负载调度的响应时间,在最小化响应时间的状态下得到期望利用率,从而增加了适应度值计算的有效性,更好地实现服务集群的负载均衡。

2.3 自适应阈值函数设计

自适应的阈值策略中的阈值^[12,13]表示处理器是重负载或轻负载。每一个处理器在达到或完成任务时会直接向中央调度程序报告,然后根据目前的系统负载和新调度任务的负载得到单个服务器的负载均值。系统设置了重阈值($L_{\max}$)和轻阈值($L_{\min}$)来判断服务器当前所处的负载环境,基于平均负载值可导出:

$$L_{\text{ave}} = \frac{CSL + NTL}{N} \quad (1)$$

$$L_{\max} = R \times L_{\text{ave}} \quad (2)$$

$$L_{\min}=D \times L_{\text{ave}} \quad (3)$$

其中,CSL(current system load)定义为目前的节点负载量,表示负载均衡器中遗传调度算法未给服务器集群分配任务,服务集群依旧在执行前一次调度算法分配的任务量。

NTL(new tasks load)定义为即将分配的负载量,表示遗传调度算法通过迭代操作已经选择出最优分配组合,即将分配给服务集群的任务量。

L_{ave} 是根据目前的节点负载量(CSL)和即将分配的负载量(NTL)之和与总服务节点量 N 的比值求得。而 R 和 D 给负载均衡机制添加了灵活性和有效性, R 是比 1 大的值,表明处理器的负载量大于平均负载值小于重阈值;同时 D 是比 1 小的值,表明处理器的负载量小于平均负载值,大于轻阈值。超负载的服务器将不再分配任务,否则会出现负载过重或过轻的不均衡问题。其中, R 和 D 的值根据任务的数量及服务器的性能进行设置。自适应阈值策略可以灵活地调动整个负载系统,是负载均衡系统的重要保障。

3 负载均衡优化

根据当前时刻 t 的系统状态采集各服务节点的信息,当有空闲出现时需要负载均衡器调用均衡算法进行新一轮的任务分配。由于本文负载均衡调度是结合 GA 解最优化问题,采用生存的优胜劣汰技术交换信息进行个体创新。其中遗传方法需要根据历史信息及预期对新搜索点进行改善。这里的负载策略主要包括参数编码和适应度设计。

3.1 编码

在编码机制中,参数的编码方式有多种^[14],其中二进制编码因为简单易行且处理模式数最多成为最常用的编码方法,但是编码串太长使得空间搜索量变大且将一直占用内存,导致计算机资源的使用率较低,因此提出采用三维十进制对空间的候选解进行参数编码。其中每个组成数组的字符串具有固定大小,在搜索空间中的每个节点都有字符串代表,而且其代表的字符串是唯一的,从而三维十进制编码也将提高遗传操作的准确性。

在三维十进制编码中,每个解被编码为一个由 3 个属性表示的十进制数组,记作 $\langle T_{\alpha}, T_{\beta}, P_i \rangle$,这里的 T_{α} 、 T_{β} 分别表示任务的编号和任务的负载量,而 P_i 表示此任务被分配的服务器 i ,其中 $i=1,2,\dots,m$ 。表 1 是将 10 个并发任务分配给 4 个服务器,其中第一组表示服务器 1 上的任务 3 的负载量为 7,第二组表示服务器 4 上的任务 5 的负载量为 3 等。

表 1 任务编码

T_{α}	T_{β}	P_i	数组
3	7	1	<3,7,1>
5	3	4	<5,3,4>
2	4	3	<2,4,3>
7	1	4	<7,1,4>
10	9	2	<10,9,2>
1	5	2	<1,5,2>
4	16	1	<4,16,1>
6	2	3	<6,2,3>
9	8	4	<9,8,4>
8	14	1	<8,14,1>

采用随机函数产生 Q 个初始结构数据,每个结构数据称为一个字符串,也称为一个个体,因此 Q 个字符串构成了一个群体。在 GA 中将随机产生的适量初始串结构数据作为初始种群。

3.2 适应度函数构建

负载策略目的是使分配给服务集群的负载更均衡,而在 GA 中适应度函数是评价字符串性能的唯一标准,所以有效地构建适应度函数就变得极其重要。将适应度函数分为 3 部分,包括资源利用率、响应时间和可接受分配服务器的概率。

3.2.1 负载指数

通过服务器运行时各方面的参数得到负载指数,主要包括 CPU 使用量、内存和带宽利用率。变量中 CPU、内存及带宽的利用情况可以通过负载均衡器监测到。

其中,每个服务器的内存利用率定义为:

$$D_{\text{ui}} = \left(\frac{V_{\text{di}}}{P_{\text{di}}} \right) \times 100\% \quad (4)$$

其中, V_{di} 为服务器的已用内存, P_{di} 为服务器 i 的总内存。CPU、带宽利用率的定义与内存利用率具有相似形式,分别为:

$$C_{\text{ui}} = \left(\frac{V_{\text{ci}}}{P_{\text{ci}}} \right) \times 100\% \quad (5)$$

$$B_{\text{ui}} = \left(\frac{V_{\text{bi}}}{P_{\text{bi}}} \right) \times 100\% \quad (6)$$

其中, V_{ci} 、 V_{bi} 分别为服务器已被占用的 CPU 和带宽, P_{ci} 、 P_{bi} 则是服务器的总资源。

对于负载指数的衡量,在不同类型的系统应用中,各个参数的重要程度也有所不同。在典型的 Web 应用环境下,可使用的内存资源和响应时间就非常重要,如果用户以长的数据库事务为主^[15-17],则 CPU 使用率和可用内存就



相对重要一些。因而在资源利用率的问题上不可将以上因素同等看待,为了方便在系统运行过程中针对不同的应用对各个参数的比例进行适当调整,为每一个参数设定一个常量系数 $k_i(i=1,2,3)$,用来表示各个负载参数的权值。所以服务器的资源利用率表示如下:

$$R_w = k_1 D_w + k_2 C_w + k_3 B_w \quad (7)$$

其中, k_1, k_2, k_3 为常数,且 $k_1 + k_2 + k_3 = 1$ 。常量系数根据系统的应用环境进行设置。

3.2.2 资源利用率和响应时间分析

由于不同的服务器的性能和使用的情况不同,引入了 mean-variance 模型计算服务集群的资源利用率及响应时间^[7]。假设对 m 个服务器进行资源利用率的配置,对应的资源利用率为随机变量 $R_w(R_{w1}, R_{w2}, \dots, R_{wm})$,则系统中总的节点利用率为:

$$R_p = \sum_{i=1}^m w_i \times R_{wi} \quad (8)$$

其中, $w_i(i=1,2,\dots,m)$ 代表 m 个服务器资源利用率的比,即权重因子。系统中总的期望 u_p 和响应时间 $Makespan$ 为:

$$u_p = E[R_p] = \sum_{i=1}^m E[w_i R_{wi}] = \sum_{i=1}^m w_i u_i \quad (9)$$

$$Makespan = \sigma_p^2 = var(R_p) = \sum_{i=1}^m \sum_{j=1}^m w_i w_j cov(R_{wi}, R_{wj}) = \sum_{i=1}^m \sum_{j=1}^m w_i \sigma_i \sigma_j \quad (10)$$

其中, u_i 是服务器 i 资源利用率的期望,而 $cov(R_{wi}, R_{wj})$ 表示任意两个服务器资源使用情况的协方差,协方差也可表示为 σ_{ij} 。mean-variance 模型通过求解约束优化问题来获得最优的权重向量:

$$\min \sum_{i=1}^m \sum_{j=1}^m w_i \sigma_i \sigma_j \quad (11)$$

根据期望及权重因子可以得到目标函数的两个约束条件:

$$\sum_{i=1}^m w_i R_{wi} = u_p \quad (12)$$

$$\sum_{i=1}^m w_i = 1 \quad (13)$$

在限制条件下求解 R_w 资源利用率组合时的最小负载执行时间,关于最值问题,可通过拉格朗日目标函数求得。构建拉格朗日式如下:

$$L = \sum_{i=1}^m \sum_{j=1}^m w_i w_j \sigma_i \sigma_j - \lambda_1 \left(\sum_{i=1}^m w_i - 1 \right) - \lambda_2 \left(\sum_{i=1}^m w_i R_{wi} - u_p \right) \quad (14)$$

其中, λ_1 和 λ_2 是拉格朗日乘数,通过计算 L 相对于 w_i 和拉格朗日乘子的导数为 0 的等式来获得最优的权重向量。分别对 $w_i, \lambda_1, \lambda_2$ 求偏导:

$$\frac{\partial L}{\partial w_i} = \sum_{j=1}^m \sigma_j w_j - \lambda_1 - \lambda_2 R_{wi} = 0, \quad i=1,2,\dots,m \quad (15)$$

$$\frac{\partial L}{\partial \lambda_1} = \sum_{j=1}^m w_j - 1 = 0 \quad (16)$$

$$\frac{\partial L}{\partial \lambda_2} = \sum_{j=1}^m w_j R_{wj} - u_p = 0 \quad (17)$$

通过求偏导数得到权值因子 w_i ,从而在式(8)和式(10)中可得到目标函数的总资源利用率和响应时间。

3.2.3 可接受分配的服务器

虽然服务器的资源利用率得到提高,执行时间缩短但仍旧可能出现部分服务器超载运行的状况,所以负载均衡的下一个目标是提高可接受分配服务器 (number of acceptable distribution server, ADS) 的概率。

自适应阈值策略得到重阈值 (L_{max}) 和轻阈值 (L_{min}),每个服务器目前的负载与将要分配的负载之和不能超过重阈值或低于轻阈值:

$$L_{min} \leq CLS + NTL \leq L_{max} \quad (18)$$

可接受分配的服务器表示服务器目前的负载值符合以上条件,即式(18),且允许被调度算法再次分配任务。但是,服务器的负载值超过 L_{max} 或低于 L_{min} 则此服务器是不正常运作,也将被定义为不可接受分配任务的服务器。在整个字符串中可接受分配任务的服务器的数量越多,表示调度任务的分配更均衡。

3.2.4 组合适应度函数设计

通过 3 个目标函数的计算可得到组合适应度函数为:

$$fitness = \frac{1}{Makespan} \times R_p \times \frac{ADS}{N} \quad (19)$$

这里的适应度函数是用来评价负载调度任务的质量。其中,式(14)构建的拉格朗日式可以得到权值因子 w_i ,从而通过式(8)和式(10)中可得到目标函数的总资源利用率 R_p 和响应时间 $Makespan$ 。当响应时间缩短,资源利用率提高,且可接受分配服务器 ADS 的概率也较高时,适应度值就会增大,这就表示负载均衡策略越好。

服务集群按照组合适应度函数进行评估操作,从而实现分配组合的优胜劣汰。以下算法利用 mean-variance 模型

计算分配组合的适应度函数值,通过遗传操作并行搜索得到最优解。首先,按轮盘赌选择方法对适应性强的字符串进行选择复制,对选择的优秀字符串进行交叉、变异运算;再次,在完成变异操作后,字符串将产生新的适应度值,所以需要被重新评估,然后产生新的幸存概率,这些值将被用来定义下一轮循环中轮盘的插槽值;最后,在需要判断是否达到 K 次循环后,再决定是否将得到的最优字符串解码并用于任务分配或继续迭代。对于新系统状态 $t+1$ 时刻,检查是否有空闲处理器出现,若有空闲,将启动负载均衡进行新任务的分配。

算法 负载均衡的最优解设计

```
Load balance algorithm{
RequestQ in sliding-window //请求队列
ServerList//服务节点负载信息
Do{
    Requesti= RequestQ.Out_Q //请求出队
    Shedule IGA //通过遗传算法分配请求
    Initialize newPop //初始化种群
    Evaluate fitness of  $P(t)$  based mean-variance //“适者生存”遗传判断
    While(not Terminate-Condition) //不满足终止条件时,循环
    {
        Select operation for  $P(t)$  //选择操作
        Crossover operation for  $P(t)$  //交叉操作
        Mutation operation based Polynomial Mutation for  $P(t)$  //变异操作
         $P(t+1) = P(t)$  //得到下一代群体  $P(t+1)$ ,循环
        GA-operation //遗传操作
        Evaluate fitness of  $P(t)$  //适应度判断
    }
    End While
    Output(Requesti)
}}
```

4 仿真实验设计与结果

针对适应度函数中响应时间和资源利用率的有效性计算问题,引入投资组合选择模型 mean-variance 来计算服务集群中节点利用率的权重,优化遗传算法中的适应度函数,以获得最优的分配组合。

4.1 实验设计

为了验证本文的优化算法增加了适应度函数的有效性,以提高服务集群的负载均衡表现。本文使用 OPNET14.5 仿真平台进行测试,其中服务集群的负载均衡策略遵循三层建模规则,包括网络模型、节点模型和进程处理模型。服务集群采用集中式负载均衡网络拓扑结构,100 Mbit/s 带宽的局域网架构,集群的 3 个处理器分别为 6MIPS、3MIPS、1MIPS。客户端为 4 个以太子网,每个子网包括 20 个用户,子网彼此通过 64 口以太网交换机连接,负载均衡单元与客户端在 10 Mbit/s 线路上交流,而 16 口的 Hub 为服务集群与负载均衡单元的交流服务。

节点模型中负载均衡器的设计遵循 OSI 建模规则^[18],在 IP 处理层采用 NET 实现基于 mean-variance 的负载均衡算法。该工作的实现在进程模型中,首先,初始化负载均衡单元;然后,对获得的数据分组进行源、目的地址分析并根据其端口号进行分类,根据用户请求任务调用负载均衡算法分配给对应的服务器;最后,把响应的数据分组进行 NET 地址转换再反馈给客户端。

完成网络、节点及进程建模后,通过图 1 的负载均衡系统实现本文算法的性能表现,其中设计的负载均衡模型包含负载均衡调度模块(基于 mean-variance 的负载均衡算法、FCFS(first come first served)算法^[4]、DGA(dynamic genetic load balancing)算法^[31])、数据分组分析模块、负载信息采集模块、请求转发模块及服务集群模块。当客户端传入数据分组后,根据网络协议的端口判断请求任务的类型^[9],将请求数据分组传递给负载均衡调度模块。在 mean-variance 模型的基础上通过 GA 并行搜索得到最优分配组合(其中负载采集模块需要描述服务节点的 CPU 使用量),内存和带宽利用率等采集间隔时间设置为 10 s。改写用户的请求数据分组的地址为对应服务器的内部 IP 地址,再将数据分组发送至服务集群中。

4.2 实验结果与分析

对于算法效果的验证,将本文提出的方法与 FCFS 及 DGA 进行比较。实验运行基于表 2 的默认参数。

根据以上的参数设置,算法会有不同的性能表现。在负载均衡机制中迭代次数和并发请求量是影响节点利用率和响应时间两个目标函数的关键因素,但随机分配中部分参数只与 GA 相关。例如,迭代次数的变化在节点利用率和响应时间方面对 FCFS 算法没有影响。所以进行实验时,在

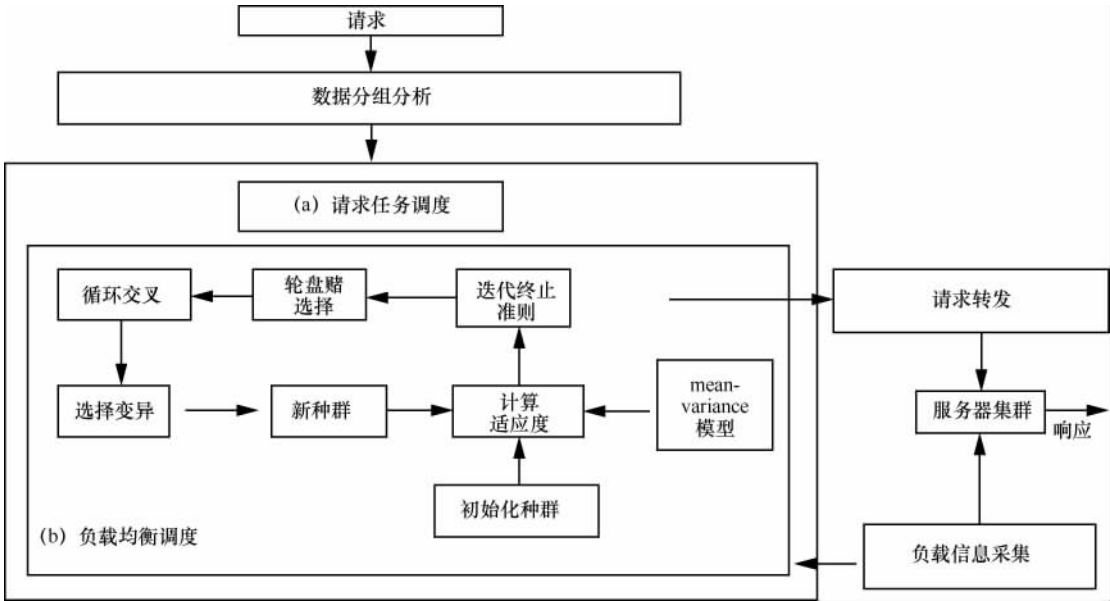


图 1 负载均衡系统模型设计

表 2 实验参数

参数	数值
服务节点	3
负载均衡管理单元	1
任务量/个	3 000
群体大小	20
迭代次数/次	35
交叉概率	0.8%
变异概率	0.15%
乘子 R	1.2
乘子 D	0.8
常量系数 k_1, k_2, k_3	0.4, 0.4, 0.2

GA 算法中改变迭代次数来观察参数对算法性能的影响。

在图 2 和图 3 中可以看出迭代次数从 5 次增加到 60 次时,两种算法的响应时间的增幅逐渐减少,而节点利用率为增加的趋势,最后趋于平缓。并且本文的负载均衡方法在响应时间和节点利用率方面都优于 DGA 算法。已知适应度函数是用来评价负载调度的质量,图 4 的结果中,当迭代次数增加时本文算法的适应度值逐渐增加,达到 35 次时目标值最大。结合图 2、图 3 观察,随着迭代的继续进行使得资源利用率有所提高,但在响应时间上却出现增幅较大的现象。因此对 GA 进行过多的迭代是多余的并且会增加响应时间,违背了负载调度的最初目的。所以将迭代次数设置为 35 次会使得本文算法得到最优适应度,并且本文的算法相对于 DGA 算法的适应度值较高,这也表明了本文算法的优越性。

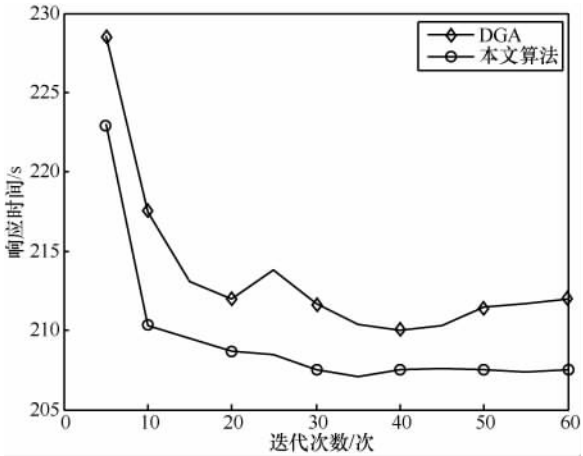


图 2 Makespan: 改变迭代次数

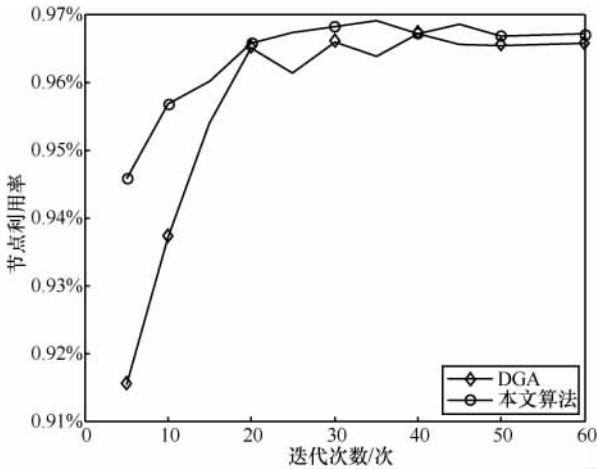


图 3 平均节点利用率:改变迭代次数

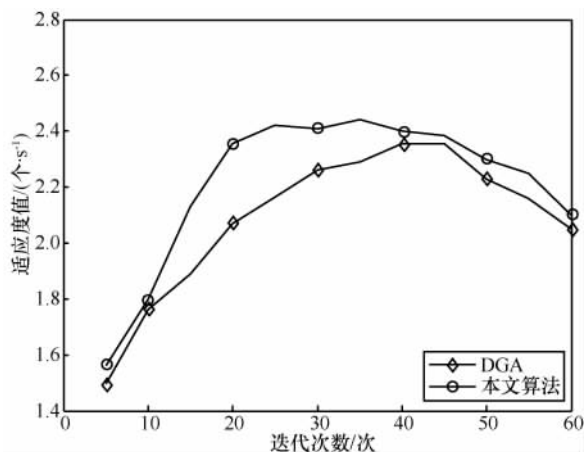


图4 适应度值:改变迭代次数

除了迭代次数外,并发请求量也是均衡算法的另一个关键因素,当任务量增加时3种算法在响应时间和节点利用率上的负载均衡表现如图5、图6所示。

本实验中除了任务数量外其他参数均为默认值,如图5

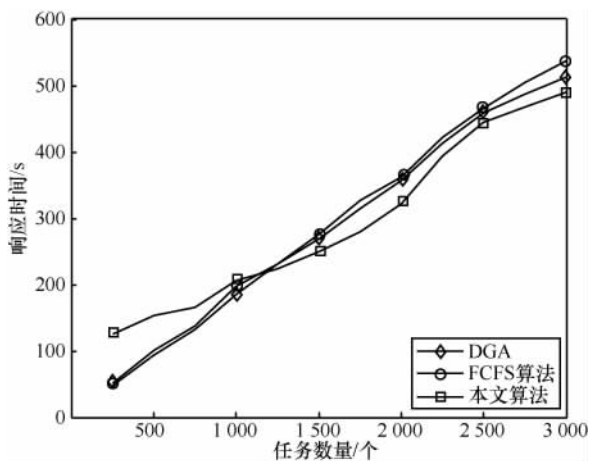


图5 Makespan:改变任务数量

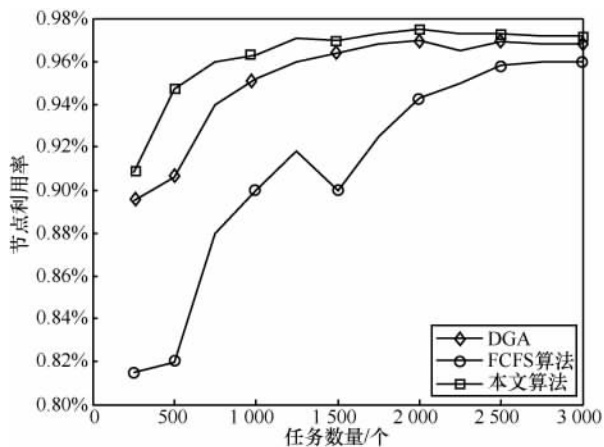


图6 平均节点利用率:改变任务数量

所示,任务数量从250个到3000个进行间隔250个的线性增加,3个算法总响应时间也逐渐增加。在请求任务较少时,本文算法因引入拉格朗日乘数两个约束条件求最优权重因子,并对3个参数求偏导数导致响应时间较长。但随着随机分配的任务数量大于1250左右时,本文算法相对于DGA和FSFC算法的利用率产生一定差距。实验表明,通过增加的mean-variance模型对节点利用率权重的精确计算,使得适应度函数的准确性得到改善,从而节省了遗传算法通过优胜劣汰选择最优分配组合的时间,使得负载均衡算法分配任务到服务集群的响应时间最小化。

如图6所示,随着任务的增加本文算法的服务集群节点利用率逐渐从90.5%增长到96.8%,与DGA相比较增加了0.9%。表明在不同负载比重的任务类型中,采用mean-variance模型有效地计算适应度函数,然后通过拉格朗日乘数求得约束条件下节点利用率的权重因子,此方法是值得肯定的。本文提出的算法使得每个服务器资源利用率的权重因子得到更精确的评估和预测,从而提高了并发任务分配的效率。

5 结束语

有效的任务分配中负载均衡策略至关重要,大部分研究都集中关注这一问题并提出了一系列的解决方案。本文通过对服务集群均衡方法的研究,发现结合投资组合选择理论中的mean-variance模型来设置服务集群中节点利用率的权重,以最小化任务完成时间为条件来获得最优的权值向量,为每个服务器的资源利用率分配权值从而得到更有效的组合适应度函数。实验结果表明,本文所提出的方法在缩短响应时间,提高节点利用率及负载均衡分配方面有很好的表现。但随着应用服务的逐渐升级,在大规模的应用场景中用户还需要一定的QoS保障^[9]。为此,本文将在现有的基础上,进一步结合用户QoS保障机制进行深入研究。

参考文献:

- [1] 李文中,郭胜,许平,等. 服务组合中一种自适应的负载均衡算法[J]. 软件学报, 2006, 17(5): 1068-1077.
LI W Z, GUO S, XU P, et al. An adaptive load balancing algorithm for service composition [J]. Journal of Software, 2006, 17(5): 1068-1077.
- [2] DONG B, LI X, WU Q, et al. A dynamic and adaptive load balancing strategy for parallel file system with large-scale I/O



- servers [J]. Journal of Parallel & Distributed Computing, 2012, 72(10): 1254-1268.
- [3] RAMAKRISHNA M, KODATI V, GRATZ P, et al. GCA: global congestion awareness for load balance in networks-on-chip [J]. IEEE Transactions on Parallel & Distributed Systems, 2016: 1-8.
- [4] ZHANG Y, LIAO X, JIN H, et al. Inc-part: incremental partitioning for load balancing in large-scale behavioral simulations [J]. IEEE Transactions on Parallel & Distributed Systems, 2015, 26(7): 1900-1909.
- [5] CHANDAKANNA V R, VATSAVAYI V K. A sliding window based self-learning and adaptive load balancer [J]. Journal of Network & Computer Applications, 2015, 56(C): 188-205.
- [6] LI Y, YANG Y, MA M, et al. A hybrid load balancing strategy of sequential tasks for grid computing environments [J]. Future Generation Computer Systems, 2009, 25(8): 819-828.
- [7] KORKHOV V V, MOSCICKI J T, KRZHIZHANOVSKAYA V V. Dynamic workload balancing of parallel applications with user-level scheduling on the Grid [J]. Future Generation Computer Systems, 2009, 25(1): 28-34.
- [8] BALASANGAMESHWARA J, RAJU N. A hybrid policy for fault tolerant load balancing in grid computing environments [J]. Journal of Network & Computer Applications, 2012, 35 (1): 412-422.
- [9] 吴伟平, 高建军, 李端. 多阶段均值-方差资产负债管理的随机控制 [J]. 控制理论与应用, 2015, 32(9): 1200-1207.
- WU W P, GAO J J, LI D. Stochastic control for multiperiod mean-variance asset-liability management [J]. Control Theory & Applications, 2015, 32(9): 1200-1207.
- [10] FOSTER I, KESSELMAN C, TUECKES. The anatomy of the grid: enabling scalable virtual organizations [J]. The International Journal Performance Computing Applications, 2001, 15 (3): 200-222.
- [11] WU Z W, SONG X F, XU Y Y, et al. A note on a minimax rule for portfolio selection and equilibrium price system [J]. Applied Mathematics & Computation, 2009, 208(1): 49-57.
- [12] 韩东升, 丁莎莎, 余萍. 一种基于阈值的无线异构网络基站分簇方法 [J]. 电信科学, 2015, 31(4): 18-21.
- HAN D S, DING S S, YU P. A clustering method based on the threshold in wireless heterogeneous network [J]. Telecommunications Science, 2015, 31(4): 18-21.
- [13] MOHAMED N, AL-JAROUDI J, EID A. A dual-direction technique for fast file downloads with dynamic load balancing in the cloud [J]. Journal of Network & Computer Applications, 2013, 36(4): 1116-1130.
- [14] 邓亮, 赵进, 王新. 基于遗传算法的网络编码优化 [J]. 软件学报, 2009, 20(8): 2269-2279.
- DENG L, ZHAO J, WANG X. Genetic algorithm solution of network coding optimization [J]. Journal of Software, 2009, 20(8): 2269-2279.
- [15] 苏金树, 郭文忠, 余朝龙, 等. 负载均衡感知的无线传感器网络容错分簇算法 [J]. 计算机学报, 2014, 37(2): 445-456.
- SU J S, GUO W Z, YU C L, et al. Fault-tolerance clustering algorithm with load-balance aware in wireless sensor network [J]. Chinese Journal of Computers, 2014, 37(2): 445-456.
- [16] ANGRISANI L, CAPRIGLIONE D, FERRIGNO L, et al. Locality-sensitive task allocation and load balancing in networked multiagent systems: talent versus centrality [J]. Journal of Parallel & Distributed Computing, 2011, 71(6): 822-836.
- [17] DONG B, LI X Q, WU Q M, et al. A dynamic and adaptive load balancing strategy for parallel file system with large-scale I/O servers [J]. Journal of Parallel and Distributed Computing, 2012, 72(10): 1254-1268.
- [18] CHENG D, RAO J, GUO Y, et al. Improving performance of heterogeneous mapreduce clusters with adaptive task tuning [J]. IEEE Transactions on Parallel & Distributed Systems, 2016: 1.
- [19] 郭涛, 李有明, 雷鹏, 等. MIMO 中继系统中一种基于用户 QoS 的资源分配方法 [J]. 电信科学, 2015, 31(4): 115-120.
- GUO T, LI Y M, LEI P, et al. A resource allocation scheme based on user's QoS in MIMO relay system [J]. Telecommunications Science, 2015, 31(4): 115-120.

[作者简介]



包晓安(1973-),男,浙江理工大学教授,主要研究方向为软件工程及软件测试、智能信息处理。

魏雪(1990-),女,浙江理工大学硕士生,主要研究方向为网关负载调度和智能优化算法。

陈磊(1992-),男,浙江理工大学硕士生,主要研究方向为智能信息处理及嵌入式设备视频采集。

胡国亨(1992-),男,浙江理工大学硕士生,主要研究方向为智能信息处理及物联网协议。

张娜(1977-),女,浙江理工大学副教授,主要研究方向为软件工程及软件性能分析、软件测试技术。