

基于分配器队列模糊控制的集群负载平衡

于国防, 王耀才, 庄立运, 贾栋清

(中国矿业大学信息与电子工程学院, 徐州 221008)

摘要: 在分析和比较现有的基于分配器的服务器集群负载平衡方案的基础上, 综合运用排队论知识和模糊控制理论, 提出一种适合于小型 Web 服务器集群的模糊动态控制队列长度的负载分配器, 该分配器在注重服务质量调控的同时, 兼顾了集群的总体负载均衡。通过 Visual C++ 编程构建模拟实验平台, 验证了该方案的可行性。

关键词: Web 服务器集群; TCP 迁移式分配器; 排队论; 模糊控制; 负载平衡

Clusters Load Balance Based on Fuzzy Control of Distributor Queue

YU Guo-fang, WANG Yao-cai, ZHUANG Li-yun, JIA Dong-qing

(College of Information and Electronic Engineering, China University of Mining and Technology, Xuzhou 221008)

【Abstract】 Based on analysis and comparison of existed load balancing program on the cluster server distributor, this paper studies a kind of dynamic fuzzy queue length distributor, which is suited to the small Web server clusters. The distributor focuses on the quality of service regulation, and overall cluster load balance. Feasibility of the scheme on the simulation platform programmed by Visual C++ is tested. Experimental results show that the scheme is effective.

【Key words】 Web server clusters; TCP handoff distributor; queuing theory; fuzzy control; load balance

1 概述

独立服务形式的 Web 服务器负荷的日益加重, 使得提高 Web 服务器集群的负荷能力和服务性能成为因特网应用研究的重点, 因此, 主要研究以下两方面问题:

(1) 集群请求分配器设计。目前的请求分配器更多地采用 L7 集群方式, 包含 TCP 传递^[1]和 TCP 迁移^[2-3]两种请求分配机制。前种机制因应答数据需经过分配器转发给客户, 易形成分配器瓶颈, 效率低; 而后种机制返回信息直接由后台服务器发送给客户, 效率较高、扩展性较好。文献[4]还提出了旨在解决请求网络瓶颈问题的双层请求分配器调度模式。

(2) 分配器的负载平衡算法。目前一些研究者已提出了多种负载平衡算法, 如基于动态反馈的调度算法^[5]、基于轮转法的动态加权算法^[6]、基于最小连接数的动态加权算法^[7]、基于相对负载权重比值的调度算法^[8]等。

但是, 这些负载平衡调度的研究更侧重于集群服务器之间的负载均衡, 缺乏对服务器本身的“超负荷”以至造成服务质量下降问题的研究。

2 基于队列管理的 TCP 迁移式分配器

2.1 分配器的基本结构

分配器的服务队列长度不仅影响其调度效率, 也影响着其后端的服务器服务效率, 笔者在排队论基础上, 研究基于队列管理的 TCP 迁移式服务器集群分配器, 结构如图 1 所示。

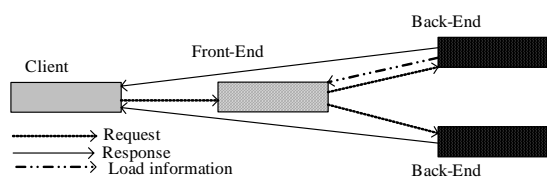


图 1 基于队列管理的 TCP 迁移式分配器

在任意一台服务器上配置服务监测器(因为已假设各服务器服务率相同), 探测和计算服务器的平均服务率和平均响应时间, 并向分配器通报, 探测周期为 T ; 也在分配器上配置监测器, 探测和计算顾客的平均到达率。

为了便于问题分析, 这里的 Web 集群采用同配置(软硬件均相同)服务器, 以排队论的三要素角度, 做如下假设:

- (1) 顾客
顾客(请求)以参数 λ 的泊松分布到达。
- (2) 排队规则
1) 顾客: 等待制;
2) 服务: FCFS(先到先服务);
3) 排队空间: 具体, 即队列(分配器)内最多容纳 K 个顾客(请求);
4) 队列数目: 1 个, 即一台负载平衡分配器。
- (3) 窗口
1) 服务窗口有 c 个, 即, 集群中含 c 台服务器;

2) 服务器在运行中所表现出的服务能力为 μ (这里以单位时间内所处理的请求数为衡量标准, 但从严格意义上说, 在实际服务中, 由于请求类型等方面的差异, 因此同配置的服务器所表现出的服务能力并不同, 但从平均服务率角度来看, 由于配置相同, 顾客到达概率均等, 因此各服务器所表现出的服务能力却是基本相同的), 即: 服务服从 $1/\mu$ 的指数分布; 由于服务器服务能力均等, 因此队列前到达服务窗口(服务器)的顾客(请求)会均匀地分布到各服务窗口(服务器), 相当于采用轮循平衡算法的分配器。

作者简介: 于国防(1965 -), 男, 副教授, 主研方向: 最优控制, 因特网 QoS; 王耀才, 教授、博士生导师; 庄立运、贾栋清, 硕士研究生

收稿日期: 2007-06-08 E-mail: gfyu90@163.com

根据以上假设，可以将分配器和集群服务器看作是一个 $M/M/c(K)$ 排队系统，如图 2 所示。

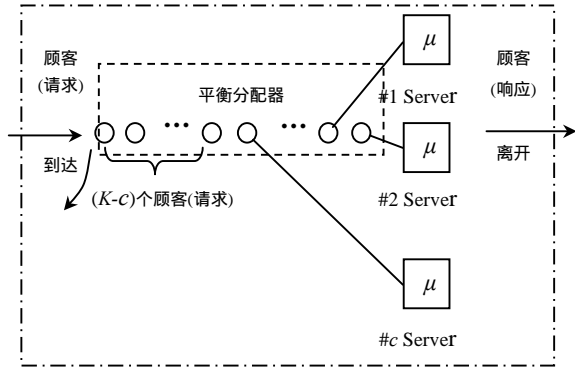


图 2 Web 服务集群及平衡分配器

上述参数中，顾客的平均到达率 λ 及服务器的平均服务率 μ 分别由分配器和服务器上的监测器探测，并进一步计算获得。

2.2 分配器的工作过程

集群工作开始时，分配器设定服务队列长度初值 n 。如果采用的是 Apache Web 服务器，其服务线程数为 S (可以通过其配置文件 httpd 的 ThreadsPerChild 参数设定)，不妨设

$$n = c \times S \quad (1)$$

集群工作 T 周期后，根据以下排队论公式^[9]

$$p_0 = \left[\sum_{n=0}^{c-1} \frac{1}{n!} \rho^n + \sum_{n=c}^K \frac{1}{c! c^{n-c}} \rho^n \right]^{-1} \quad (2)$$

$$p_j = \begin{cases} \frac{\rho^j}{j!} p_0 & 1 \leq j < c \\ \frac{\rho^j}{c! c^{j-c}} p_0 & c \leq j \leq K \end{cases} \quad (3)$$

$$q_j = \frac{p_j}{1 - p_k}, j = 0, 1, \dots, K-1 \quad (4)$$

即可计算出顾客(请求)在系统中的平均逗留时间为

$$\bar{W} = \sum_{j=c}^{K-1} \frac{j-c+1}{c\mu} q_j + \frac{1}{\mu} \quad (5)$$

设顾客期望的平均响应时间为 \bar{w}' 。根据负载调度原则，当 $\bar{w} > \bar{w}'$ 时，应减小队列长度 K ，降低系统负荷，从而缩短平均响应时间；而当 $\bar{w} < \bar{w}'$ 时，应增加队列长度 K ，以提高系统负荷，从而尽可能地发挥系统服务效能。但是，直接根据 \bar{w} 与 \bar{w}' 之间的变化关系，利用上述的排队论公式，需要进行频繁、复杂的计算，不但增加了分配器的系统负荷，影响其负载均衡效率，也削弱了系统的稳定性。鉴于此，本文采用模糊控制技术^[10]，并结合运用排队论进行模糊集划分，从而达到服务队列模糊动态控制的目的。

设，监测的平均响应时间范围为 $\bar{w}'_{\min} \sim \bar{w}'_{\max}$ ，且 $\bar{w}' = (\bar{w}'_{\max} - \bar{w}'_{\min})/2$ 。

再设，监测的平均服务率范围为 $\mu_{\min} \sim \mu_{\max}$ 。

由公式可得出以下关系：

$$\bar{w}' \uparrow \Rightarrow K \uparrow, \mu \uparrow \Rightarrow K \downarrow; \bar{w}' \downarrow \Rightarrow K \downarrow, \mu \downarrow \Rightarrow K \uparrow$$

由式(1)~式(5)可推导出以下 2 个函数表达式：

$$K_{\min} = f_1(\bar{w}'_{\min}, \mu_{\max})$$

$$K_{\max} = f_2(\bar{w}'_{\max}, \mu_{\min})$$

按照基本的模糊集划分法，将输入和输出均划分为 5 个模糊集，分配器的逗留时间和服务率输入模糊集分别如图 3(a) 和图 3(b) 所示。分配器的输出模糊集，如图 4 所示。

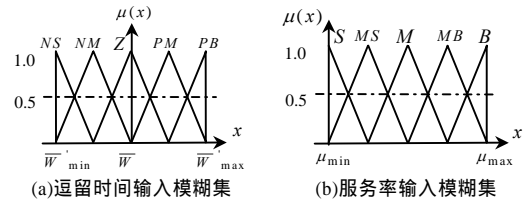


图 3 逗留时间输入模糊集和服务率输入模糊集

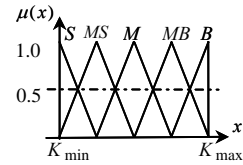


图 4 分配器的输出模糊集

模糊集中 S, M, Z, B 分别表示小、中、零和大；而 N 和 P 则分别表示负和正。采用 Mamdani 模糊推理机^[11]，建立如图 5 所示的模糊调度控制器，建立 25 条模糊推理规则。

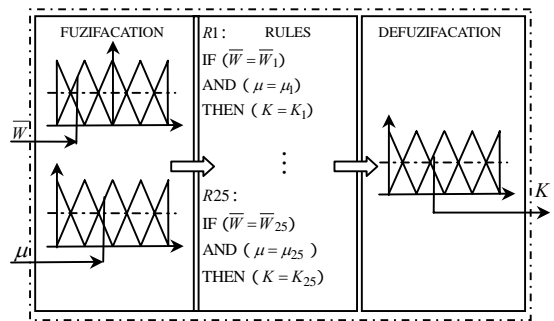


图 5 模糊调度控制器

3 实验及分析

依据上述模糊调度控制器原理，利用 Visual C++ 编程设计用于实验的 Web 浏览器和服务器^[12]，分别安装于两台实验客户机和 3 台实验服务器上，运行中设置周期为 5 s 的信息探测和分配器队列调整，并设定顾客的期望平均逗留时间(忽略网络延迟，看作平均服务时间)为 100 ms，测试中逐渐增加并发请求数量，并逐渐加大每个请求的任务比重(如，网页包含大且多的图片)。实验观测结果如图 6 和图 7 所示。为了便于对比，图 6 将分配器队列及逗留时间趋势曲线绘制于同一图中， K/W 单位为 ms。可以看出，随着并发数量的增加，分配器的队列长度逐渐增加，平均逗留时间也随之延长，但当达到预定的队列长度之后就不再增加了，此时，平均逗留时间还未达到临界平均逗留时间；如果请求任务比重加大，以至服务器的负载逐渐加重，平均逗留时间继续延长，超过了临界平均逗留时间，此时，队列长度开始缩短，服务器的负载减轻，平均逗留时间也开始缩短，达到服务质量的调控目的。因为集群的负载比较均衡，所以达到了负载平衡的控制目的。

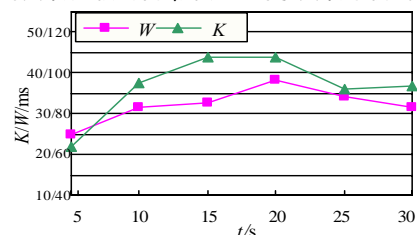


图 6 分配器的队列及逗留时间

(下转第 136 页)