基于软交换的异构集群媒体服务器中一种 LRV 负载均衡算法

吴乃星[1,2], 廖建新[1], 杨孟辉[1], 朱晓民[1]
(1. 北京邮电大学网络与交换技术国家重点实验室, 北京 100876; 2 中国联通深圳分公司, 广东深圳 518040)

摘 要: 针对基于软交换的集群媒体服务器的系统特征, 本文提出了一种LRV(Limited Resource Vector) 负载均衡算法, 该算法考虑了系统中资源的协调使用并有效防止了系统性能的剧烈变化. 通过在集群媒体服务器负载均衡系统 Petri 网模型上的大量仿真模拟, 对 LRV 负载均衡算法与其他传统负载均衡算法在异构环境下的均衡能力、系统吞吐量、系统响应时间和性能平稳性几个方面做了比较和分析. 结果表明, LRV 算法具有优越的性能, 对集群媒体服务器的异构环境有良好的适应能力.

关键词: 负载均衡算法; 异构集群; 媒体服务器; 软交换; 随机 Petri 网
中图分类号: TP301.6 文献标识码: A 文章编号: 0372-2112 (2005) 10-1745-06

A Limited Resource Vector Load-Balancing Algorithm for Softswitch-Based Heterogeneous Clustered Media Server

WU Nai-xing[1,2], LIAO Jian-xin[1], YANG Meng-hui[1], ZHU Xiao-min[1]
(1. *State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China*; 2. *China United Telecommunications Corp Shenzhen Subsidiary, Shenzhen, Guangdong 518040, China*)

Abstract: Based on the system feature of softswitch-based heterogeneous clustered media server, this paper proposed a limited resource vector load-balancing algorithm. Harmonious usage of system resources was considered and violent shaking of the system performance was avoided in the algorithm. A lot of simulations on the Petri net model of balance system had been conducted and the algorithm had been compared with some traditional algorithms on balancing ability for heterogeneity, system throughput, request response time, and performance stability. Results of simulation experiments show that the algorithm can make system get higher performance and it has excellent ability to deal with the heterogeneity of clustered media server.

Key words: load-balancing algorithm; heterogeneous cluster; media server; softswitch; stochastic Petri net

1 引言

近年来, 可以提供集语音、图像和数据于一体的多媒体综合业务的 NGN(Next Generation Network) 已经成为电信和计算机领域研究和开发的重要热点. 在 NGN 架构下, 多媒体业务平台包含多种设备. 媒体服务器在软交换设备的控制下为多媒体业务提供各种媒体资源和媒体操作环境, 是该平台中的关键设备之一. 本文把这种软交换设备控制下的媒体服务器称为基于软交换的媒体服务器.

在因特网上, 已经实现了多种流媒体服务器. 但它们在系统结构和功能上并不能满足基于软交换的媒体服务器的基本要求. 文献[1] 提出了一种称为 SCMS(Softswitch-based Clustered Multimedia Server) 的基于软交换的集群媒体服务器系统结构. SCMS 比现有的流媒体服务器系统更加灵活, 更加具有可扩展性和具 有更强大的服务功能.

SCMS 实际上是一种异构高可用性集群系统, 它采用基于前端 Dispatcher(分发器) 的体系结构, 这种结构已经在 Web 服务器集群的应用中被证明是性能最好的[2].

负载均衡算法是集群系统中的关键技术之一, 目前通常使用的算法有: 随机调度算法、最少请求优先算法和最小期望等待时间算法[3,4]. 为了提高负载均衡系统的性能, 人们还提出了各种改进的算法, 例如邸烁等人提出的 LTI 算法[5]、周幼英等人提出的 Locality 型调度算法[6]. 但是, 由于随机调度算法、最少请求优先算法和最小期望等待时间算法不考虑系统的内部结构特征, 而各种改进算法只针对特定的应用场景, 因此当它们用于 SCMS 时, 都存在不同程度的局限性和缺陷. 寻找一种适用于 SCMS 系统结构的负载均衡算法成为是该领域亟需解决的重要问题之一. SCMS 有以下主要特征: (1) SCMS

收稿日期: 2004-10-26; 修回日期: 2005-07-20
基金项目: 高等学校博士学科点专项科研基金(No. 20030013006); 国家移动通信产品研究开发专项基金(下一代移动智能网络的开发及应用); 电子信息产业发展基金(下一代网络核心业务平台); 电子信息产业发展基金(移动通信增值服务平台及应用系统)

不是普通的高性能计算服务器. 与普通高性能计算服务器注重高速计算、数据可靠相比, SCMS 更加注重数据实时处理和存储子系统、网络子系统的性能. (2) SCMS 不是普通的内容服务器, 普通的内容服务器的业务过程是一个简单的存取传输过程, 而 SCMS 提供内容服务的同时还有许多特有的业务功能(如交互式语音应答, 多媒体会议, 传真, 统一消息等), 这些功能对系统部件的性能需求是不均衡的. (3) 电信级应用的 SCMS 对性能的平稳性有更高的要求. (4) 为了实现新旧网络的平滑演进, 保护运营商已有的设备投资, SCMS 中可能存在较高程度的异构现象.

在深入考虑 SCMS 的系统特征的基础上, 本文提出一种受限资源向量(LRV, Limited Resource Vector) 负载均衡算法. 由于充分考虑了 SCMS 主要资源的协调使用, LRV 负载均衡算法比传统的负载均衡算法拥有更卓越的性能.

2 系统结构及异构性

SCMS 系统的硬件结构见图 1, 可以大致分为前台处理机和后台处理机两个部分. 前台处理机(FM) 掌握了 SCMS 的全局资源信息并负责将资源任务请求路由到后台处理机(BM) . BM 是多种业务类型的资源处理设备的通称, 例如会议资源处理机、IVR(interactive voice response) 资源处理机和视频资源处理机等. 监控台监管整个集群系统. 全部设备通过两个互为备份的 LAN 相连, LAN 通过高性能路由器连接到 IP/ATM 骨干网络上.

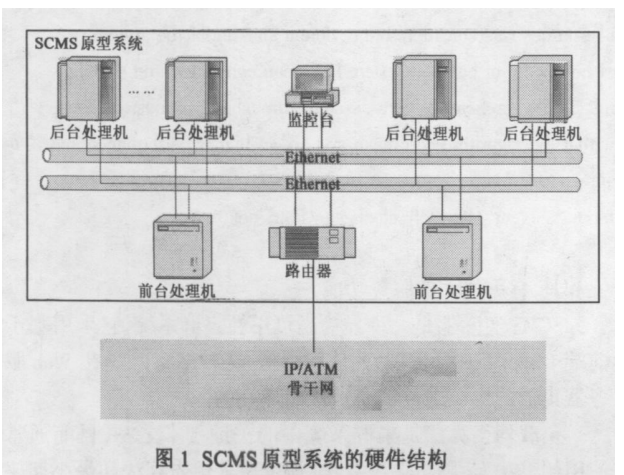


图 1 SCMS 原型系统的硬件结构

SCMS 的软件结构见图 2, RCN(Resource Control Node) 实现了各种多媒体呼叫模型并负责请求分配策略的实施. 各类型 RPN(Request Processing Node) 中的请求处理模块负责协调并实施媒体处理功能. 对不同类型的 RPN, 可能还会包含不同的专用资源处理软件, 例如 IVR RPN 中的混音软件、文语转换软件、自动语音识别软件等.

集群系统的异构性可以包含两个方面: 类型异构性和资源异构性, 类型异构性是指计算节点指令集结构(ISA) 的不同和操作系统(OS) 类型的不同; 资源异构性是指计算节点的多种资源(CPU、内存、网络子系统、本地存储子系统) 拥有量的不同. SCMS 的异构主要是 BM 的资源异构.

CPU 资源异构性、网络子系统资源异构性、本地存储子系

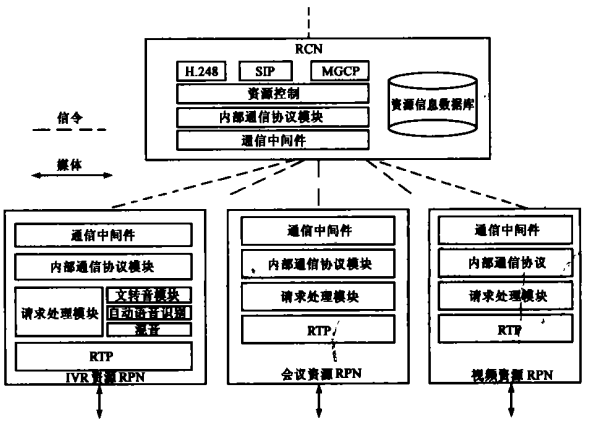


图 2 SCMS 原型系统的软件结构

统资源异构性做如下定义: CPU 资源异构性 $H_{CPU}=\sqrt{\frac{\sum_{i=1}^{N}(W_{CPU}(i)-\overline{W_{CPU}})^2}{N}}$, 其中 $W_{CPU}(i)=\frac{R_{CPU}(i)}{\max_{j=1}^{N}(R_{CPU}(j))}$, $R_{CPU}(i)$ 为 SCMS 系统第 i 个 BM 节点的处理器速率. $\overline{W_{CPU}}=\frac{\sum_{j=1}^{N}W_{CPU}(j)}{N}$; N 为系统中 BM 的节点总数.

网络子系统资源异构性 H_{net}, $H_{net}=\sqrt{\frac{\sum_{i=1}^{N}(W_{net}(i)-\overline{W_{net}})^2}{N}}$, 其中 $W_{net}(i)=\frac{R_{net}(i)}{\max_{j=1}^{N}(R_{net}(j))}$, $R_{net}(i)$ 为 SCMS 中第 i 个 BM 节点的网络子系统的数据处理速率. $\overline{W_{net}}=\frac{\sum_{j=1}^{N}W_{net}(j)}{N}$;

本地存储子系统资源异构性 H_{st}, $H_{st}=\sqrt{\frac{\sum_{i=1}^{N}(W_{st}(i)-\overline{W_{st}})^2}{N}}$, 其中 $W_{st}(i)=\frac{R_{st}(i)}{\max_{j=1}^{N}(R_{st}(j))}$, $R_{st}(i)$ 为 SCMS 中第 i 个 BM 节点的本地存储子系统的数据处理速率. $\overline{W_{st}}=\frac{\sum_{j=1}^{N}W_{st}(j)}{N}$;

从定义可知, H_{CPU}、H_{net}、H_{st} 的值越大, SCMS 的资源异构性越大.

3 负载均衡算法性能评价指标

在 SCMS 中, 对不同的负载均衡算法采用以下三个性能评价指标.

1. 平均系统吞吐量: SCMS 单位时间内完成的请求数量.

2. 平均响应时间: 根据 Little 结论[8], 在 SCMS 运行平衡状态下, 请求的平均到达率乘以平均响应时间等于系统中存在的平均请求数. 令 λ 为系统的请求到达率, l 为系统中存在的平均请求数, t 为系统平均响应时间, 则

$$t = l/\lambda$$

3. 均衡度: 在具有 N 个 BM 节点的 SCMS 中, 定义系统负载的均衡度为:

$$B = \sum_{i=1}^{N}(r_i - r_0)^2$$

其中 $r_{\cdot i} = T_i / C_i$; $r_{\cdot 0} = \sum_{i=1}^{N} T_i \Big\backslash \sum_{i=1}^{N} C_i$, $T_{\cdot i}$ 为第 i 个 BM 节点的吞吐量, $C_{\cdot i}$ 为该 BM 与拥有最大资源量的 BM 之间的资源量比.

在不同的负载均衡算法下, 系统获得的平均吞吐量越高、平均响应时间越低, 则算法的性能越好. 在不同异构程度下, 均衡度的变化曲线可以反映算法对异构环境的适应能力.

4 LRV 负载均衡算法

对于异构 SCMS, 负载均衡算法应当考虑具有同种业务功能的 BM 上多种资源之间的协调使用, 在提高资源使用效率的同时达到系统负载的平衡, 从而减少和避免资源使用瓶颈.

本文主要考虑 BM 上 CPU、网络子系统和本地存储子系统三种资源, 定义节点 $m_{\cdot i}$ 的负载向量 $V(m_{\cdot i}) = (type(m_{\cdot i}), res(m_{\cdot i}))$. 其中, 节点业务类型向量 $type(m_{\cdot i}) = (Serv(m_{\cdot i}), Subserv(m_{\cdot i}))$ 表示 $m_{\cdot i}$ 节点支持的业务类型和业务子类型. 节点资源向量 $res(m_{\cdot i}) = (U_{\cdot CPU}(m_{\cdot i}), U_{\cdot net}(m_{\cdot i}), U_{\cdot st}(m_{\cdot i}))$ 表示节点的资源使用情况, $U_{\cdot CPU}(m_{\cdot i})$ 为 CPU 资源使用率, $U_{\cdot net}(m_{\cdot i})$ 为网络子系统资源使用率, $U_{\cdot st}(m_{\cdot i})$ 为本地存储子系统资源使用率.

资源向量的各个分量表示不同资源的使用情况, 向量的角度可以显示资源使用的平衡状态. 向量的模提供了一种对 BM 节点负载状况进行比较的有效方法.

在 SCMS 系统各 BM 节点的资源为固定的情况下, 当分配给各 BM 的任务请求超过一定数量后, 会导致整个 SCMS 的吞吐量急剧下降, 同时请求的响应时间急剧上升. 这种状态称为系统的临界状态. 系统频繁进出临界状态将导致性能的剧烈震荡, 这种不稳定性对电信级应用的 SCMS 来说是不可接受的. 负载均衡系统应该避免这种情况的发生.

从资源协调使用和防止性能剧烈震荡的角度出发, 本文提出了一种 LRV 负载均衡算法. 假设通过预先测试, SCMS 的 FM 掌握了每一个 BM 在未进入临界状态时所能接纳的最大请求数, 则 LRV 负载均衡算法可以描述为: 采用资源向量作为均衡负载的参数进行请求分配, 但同时限制向接近临界状态的 BM 分配任务请求的一种算法. LRV 算法允许在 FM 上进行一定数量的请求缓冲.

算法对一个新到 SCMS 的请求的处理步骤如下:

第一步　选择可以执行该类业务的 BM 节点的集合 $M_{\cdot E\cdot}$.

$M_{\cdot E\cdot} = \{m_{\cdot i\cdot} \mid m_{\cdot i\cdot} \in M$ 且 $Rtype = type(m_{\cdot i\cdot})\}$, 其中, M 为 SCMS 中所有 BM 节点的集合. $Rtype = (Serv, Subserv)$, 是用来表示请求消息的业务类型和业务子类型的向量.

若 $M_{\cdot E\cdot}$ 为空, 算法停止, 请求消息被丢弃. 否则进入第二步.

第二步　如果 FM 中用来缓冲该类请求的缓冲区为空, 则从 $M_{\cdot E\cdot}$ 的所有节点中确定具有最小资源向量的模的 BM 点集合 E, 然后算法进入第三步. 如果 FM 中用来缓冲该类请求的缓冲区不为空, 请求被放入缓冲区, 算法停止.

$E = \{m_{\cdot i\cdot} \mid m_{\cdot i\cdot} \in M_{\cdot E\cdot}$ 且 $\| res(m_{\cdot i\cdot}) \| = \min(\| res(m_{\cdot 1\cdot}) \|, \| res(m_{\cdot 2\cdot}) \|, \ldots, \| res(m_{\cdot k\cdot}) \|)\}$

第三步　在 E 中选择初始缓冲区未满的 BM 节点组成集合 C. 初始缓冲区指的是分配到 BM 的任务请求首先需进入的缓冲区.

$C = \{m_{\cdot i\cdot} \mid m_{\cdot i\cdot} \in E$ 且 $G(m_{\cdot i\cdot}) < Q(m_{\cdot i\cdot})\}$

$G(m_{\cdot i\cdot})$ 为 $m_{\cdot i\cdot}$ 节点的初始缓冲区中的请求数量, $Q(m_{\cdot i\cdot})$ 为初始缓冲区容量. 若 C 为空, 算法停止, 请求消息将在 FM 节点中缓冲. 若 C 中 BM 节点数量等于 1, 请求消息直接进入该节点. 否则算法进入第四步.

第四步　在 C 中选择请求未达到临界状态请求数的 BM 节点组成集合 B.

$B = \{m_{\cdot i\cdot} \mid m_{\cdot i\cdot} \in C$ 且 $R(m_{\cdot i\cdot}) < P(m_{\cdot i\cdot})\}$

$R(m_{\cdot i\cdot})$ 为 $m_{\cdot i\cdot}$ 节点中正处理的请求总数, $P(m_{\cdot i\cdot})$ 为 $m_{\cdot i\cdot}$ 节点在进入临界状态前的最大请求数量. 若 B 为空, 请求消息将在 FM 中进行缓冲, 算法停止. 若 B 中 BM 节点数量等于 1, 请求消息直接进入该节点, 若 B 中的 BM 节点数量大于 1, 则对所有 BM 采取随机分配策略, 算法停止.

当 FM 节点中存在请求缓冲区非空的情况时, 算法将对缓冲区中的请求循环执行下述步骤.

第一步　在可执行该类业务请求的 BM 节点集合 $M_{\cdot E\cdot}$ 中的选择集合 E.

第二步　在 E 中选择集合 C.

若 C 为空, 算法停止, 请求消息继续在 FM 节点中缓冲. 若 C 中 BM 节点数量等于 1, 请求消息直接进入该节点, 算法停止. 否则进入下一步.

第三步　在 C 中集合 B.

若 B 为空, 请求消息继续在 FM 节点缓冲, 算法停止. 若 B 中 BM 节点数量等于 1, 请求消息直接进入该节点, 若 B 中的 BM 节点数量大于 1, 则对所有 BM 采取随机分配策略, 算法停止.

5 模拟与分析

5.1 SCMS 负载均衡系统 Petri 网模型

近年来发展迅速的 SPN(Stochastic Petri Net, 随机 Petri 网)技术为 SCMS 负载均衡系统的建模提供了一个新的、有效的途径. Petri 网已经广泛地应用在计算机科学、通信网络和工业控制领域[9]. 美国 DUKE 大学开发的 SPNP 软件包是采用计算机求解 SPN 模型的一个有效工具[10]. 本文利用 SPNP 作为 LRV 算法的性能分析工具, 首先将为 BM 建立系统模型, 然后对 SCMS 负载均衡系统进行建模.

在 SCMS 中, BM 承担媒体服务器的核心功能: 媒体数据的处理. 结合 SCMS 的软件结构并考虑一般性, 我们可以将 BM 对多媒体请求的处理过程描述如下: BM 接收到来自 FM 的多媒体请求消息, 由内部协议栈进程按照协议规定对消息

进行解析,解析后的消息通过进程之间的通信送入请求处理进程,由请求处理进程执行消息的动作、协调 BM 的软硬件资源实现指令的控制意图.在请求处理进程的协调过程中,将产生两个重要的分支,一个是对本地存储子系统的数据读写和处理,一个是对网络子系统的数据的收发和处理,请求处理进程循环的协调这些动作直到消息执行完毕.

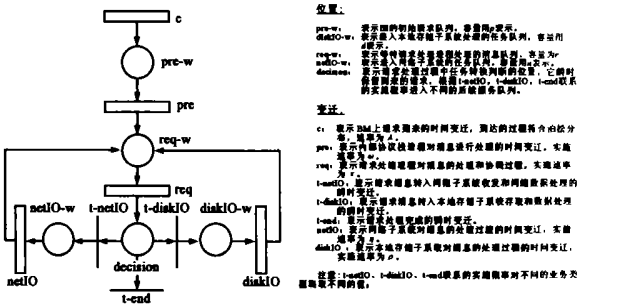


图 3 BM 节点的 SPN 模型

图 3 给出了 BM 的 SPN 模型,对这个模型我们做如下的约定:(1) 系统包含多个请求缓冲位置,位置的缓冲容量是有限的.(2) 系统对请求不区分优先级,即请求获得处理的概率是相等的.(3) 请求到达 BM 的过程为泊松过程,当 BM 的请求初始缓冲区为满时,请求到达终止.(4) 模型中存在多个服务变迁,它们具有不同的处理速率.服务速率是独立的、指数分布的.(5) 所有的时间变迁只有在变迁的后向位置未达到缓冲容量的极限时才可以实施.

模型中变迁和位置的含义见图 3.

如果将 BM 的服务过程采用一个时间变迁 s 来表示,则集群的负载均衡系统 SPN 模型可以用图 4 来描述.将 BM 模型耦合到集群负载均衡系统模型中,可以得到 SCMS 负载均衡系统的整体模型,见图 5.图中只表示了支持一类业务的模型,包含了处理该类业务的两个 BM 节点模型,本文以下内容主要以该模型为分析对象.对该模型的分析结论可以扩展到对多个 BM 节点和多种业务类型的分析.可以发现,BM 模型中的位置 $pre\text{-}w$ 和图 4 模型中的 q 是相对应的,在实际系统中它们代表同一个队列,我们称它为 BM 的初始缓冲区.

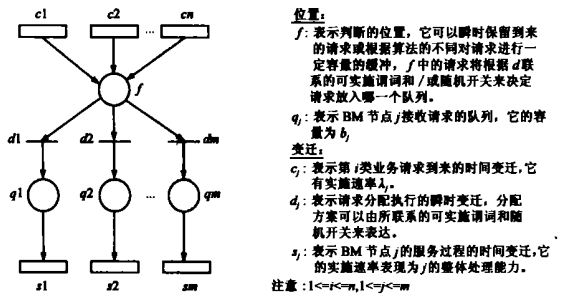


图 4 SCMS 负载均衡系统 SPN 模型 (BM 的服务过程表示为 s)

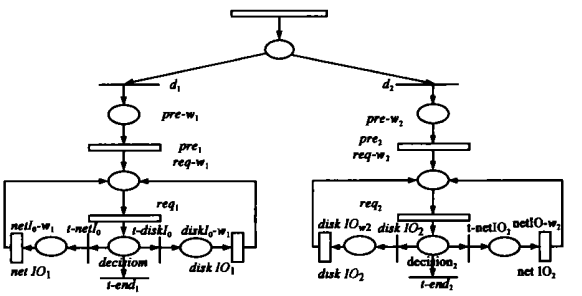


图 5 SCMS 负载均衡系统的 SPN 模型 (耦合后)

5.2 对比算法

采用四种负载均衡算法来与 LRV 负载均衡算法进行效果比较,这四种算法分别为:(1) 随机调度算法(RR).(2) 最小初始缓冲区请求数优先算法(SQR).(3) 最小系统中正处理的请求总数优先算法(SAQR).(4) 最小请求等待处理期望时间优先算法(SEDR).

使用与林闯在文献[9]中已经使用的数学符号对瞬时变迁 $d_{\cdot i\cdot}$ 的可实施谓词和随机开关做数学表达.假设 SCMS 中的 BM 节点模型数量为 m,则对 LRV 和对比算法下负载均衡系统模型中瞬时变迁 $d_{\cdot i\cdot}$ 的可实施谓词 $y_{\cdot i\cdot}$ 和随机开关 $g_{\cdot i\cdot}$ 表示如下:

(1) RR

$$y_{\cdot i\cdot}: M(pre\text{-}w_{\cdot i\cdot}) < p_{\cdot i\cdot}$$

$$g_{\cdot i\cdot}(M) = \begin{cases} \dfrac{1}{\| RR(M) \|} & \text{if} \quad i \in RR(M) \\ 0 & \text{else} \end{cases}$$

其中 $RR(M) = \{k \mid M(pre\text{-}w_{\cdot i\cdot}) < p_{\cdot i\cdot}\}$

(2) SQR

$$y_{\cdot i\cdot}: (M(pre\text{-}w_{\cdot i\cdot}) < p_{\cdot i\cdot})\, I((\text{for}\forall k \neq i,\ M(pre\text{-}w_{\cdot i\cdot}) \leqslant M(pre\text{-}w_{\cdot k\cdot})) Y(\text{for}\forall k \neq i,\ M(pre\text{-}w_{\cdot k\cdot}) = p_{\cdot k\cdot}))$$

$$g_{\cdot i\cdot}(M) = \begin{cases} \dfrac{1}{\| SQR(M) \|} & \text{if} \quad i \in SQR(M) \\ 0 & \text{else} \end{cases}$$

其中 $SQR(M) = \{k \mid M(pre\text{-}w_{\cdot k\cdot}) = \min(M(pre\text{-}w_{\cdot 1\cdot}), M(pre\text{-}w_{\cdot 2\cdot}), \cdots, M(pre\text{-}w_{\cdot m\cdot}))$ 且 $M(pre\text{-}w_{\cdot k\cdot}) < p_{\cdot k\cdot}\}$

(3) SAQR

令 $Z_{\cdot i\cdot} = M(pre\text{-}w_{\cdot i\cdot}) + M(req\text{-}w_{\cdot i\cdot}) + M(netIO\text{-}w_{\cdot i\cdot}) + M(diskIO\text{-}w_{\cdot i\cdot})$

$$y_{\cdot i\cdot}: (M(pre\text{-}w_{\cdot i\cdot}) < p_{\cdot i\cdot})\, I((\text{for}\forall k \neq i,\ Z_{\cdot i\cdot} \leqslant Z_{\cdot k\cdot}) Y(\text{for}\forall k \neq i,\ M(pre\text{-}w_{\cdot k\cdot}) = p_{\cdot k\cdot}))$$

$$g_{\cdot i\cdot}(M) = \begin{cases} \dfrac{1}{\| SAQR(M) \|} & \text{if} \quad i \in SAQR(M) \\ 0 & \text{else} \end{cases}$$

其中 $SAQR(M) = \{k \mid Z_{\cdot k\cdot} = \min(Z_{\cdot 1\cdot}, Z_{\cdot 2\cdot}, \cdots, Z_{\cdot m\cdot})$ 且 $M(pre\text{-}w_{\cdot k\cdot}) < p_{\cdot k\cdot}\}$

(4) SEDR

令 $Z_{\cdot i\cdot} = M(pre\text{-}w_{\cdot i\cdot})/\omega_{\cdot i\cdot} + M(req\text{-}w_{\cdot i\cdot})/\tau_{\cdot i\cdot} + \max(M(netIO\text{-}w_{\cdot i\cdot})/\eta_{\cdot i\cdot}, M(diskIO\text{-}w_{\cdot i\cdot})/\rho_{\cdot i\cdot})$

$$y_{\cdot i\cdot}: (M(pre\text{-}w_{\cdot i\cdot}) < p_{\cdot i\cdot})\, I((\text{for}\forall k \neq i,\ Z_{\cdot i\cdot} \leqslant Z_{\cdot k\cdot}) Y(\text{for}\forall k \neq i,\ M(pre\text{-}w_{\cdot k\cdot}) = p_{\cdot k\cdot}))$$

$$g_{\cdot i\cdot}(M)=\begin{cases}\dfrac{1}{\| SEDR(M)\|} & \text{if} \quad i\in SEDR(M)\\ 0 & \text{else}\end{cases}$$

其中 $SEDR(M)=\{k \mid Z_{\cdot k\cdot}=\min(Z_{\cdot 1\cdot}, Z_{\cdot 2\cdot}, \cdots, Z_{\cdot m\cdot})$ 且 $M(prew_{\cdot k\cdot})<p_{\cdot k\cdot}\}$

(5) LRV

令 $Z_{\cdot i\cdot}=(M(prew_{\cdot i\cdot})/\omega_{\cdot i\cdot}+M(reqw_{\cdot i\cdot})/\tau_{\cdot i\cdot})^2+(M(netIOw_{\cdot i\cdot})/\eta_{\cdot i\cdot})^2+(M(diskIOw_{\cdot i\cdot})/\rho_{\cdot i\cdot})^2$

$L_{\cdot i\cdot}=M(prew_{\cdot i\cdot})+M(reqw_{\cdot i\cdot})+M(netIOw_{\cdot i\cdot})+M(diskIOw_{\cdot i\cdot})$

$y_{\cdot i\cdot}:(M(prew_{\cdot i\cdot})<p_{\cdot i\cdot})I(((\text{for}\forall k\neq i,\ \sqrt{Z_i}\leqslant\sqrt{Z_k})I(L_{\cdot i\cdot}<P(i)))Y((\text{for}\forall k\neq i,\ \sqrt{Z_i}\leqslant\sqrt{Z_k})I(\text{for}\forall k\neq i, M(prew_{\cdot i\cdot})=p_{\cdot k\cdot})))$

$P(i)$ 为 BM 节点 i 的临界请求总数

$$g_{\cdot i\cdot}(M)=\begin{cases}\dfrac{1}{\| LRV_1(M)\|} & if \quad i\in LRV_1(M)\\ 1 & \text{if} i\in LRVH_{\cdot 2\cdot}(M),\ (M(prew_{\cdot i\cdot})<p_{\cdot i\cdot}\\ & \text{and for}\ \forall k\neq i, M(prew_{\cdot k\cdot})=p_{\cdot k\cdot}\\ 0 & \text{else}\end{cases}$$

其中 $LRV_{\cdot 1\cdot}(M)=\{k \mid \sqrt{Z_k}=\min(\sqrt{Z_1}, \sqrt{Z_2}, \cdots, \sqrt{Z_m})$ 且 $M(prew_{\cdot k\cdot})<p_{\cdot k\cdot}, L_{\cdot i\cdot}<P(i)\}$

$LRV_{\cdot 2\cdot}(M)=\{k \mid \sqrt{Z_k}=\min(\sqrt{Z_1}, \sqrt{Z_2}, \cdots, \sqrt{Z_m})\}$

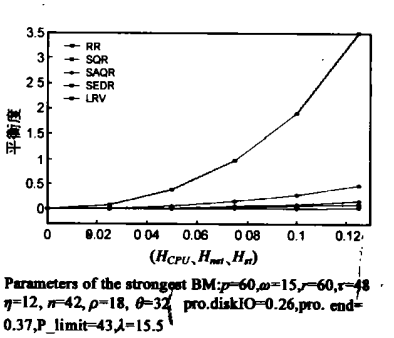
5.3　仿真参数选取

以在文献[1]SCMS 原型系统上运行的 IVR 业务为例, 选取仿真参数. 对原型系统只配备一台 IVR BM, 该 BM 的配置为: 440MHz CPU、512MB 内存, 本地磁盘容量为 20G, 磁盘平均速率 36MB/s、网络适配器速率为 100Mb/s, 同时 BM 配备了两块型号为 NMS-CG6000C 的语音板卡, 每块板卡可同时支持 120 路呼叫. 实验用 IVR 业务具有以下呼叫特征: 单个呼叫持续平均时长为 20s、平均操作本地存储的次数为 0.7 次、平均存取本地数据为 2MB, 每个呼叫平均操作板卡次数为 1 次. 经过测试, 当 SCMS 达到临界状态时, BM 内部协议栈的处理速率为 15tasks/s, 请求处理进程的速率为 48tasks/s, 在软件中设定的请求缓冲总容量大约为 194. 令 *pro. diskIO*、*pro. netIO*、*pro. end* 分别表示瞬时变迁 $t\text{-}diskIO$、$t\text{-}netIO$、$t\text{-}end$ 的随机开关, 本文将上述 BM 配置作为 SCMS 中的能力最大的 BM 配置. 因此可以为该 BM 的 SPN 模型选择以下仿真参数: $p=60$; $\omega=15$; $r=60$; $\tau=48$; $n=42$; $\eta=12$; $\theta=32$; $\rho=18$; *pro. diskIO* = 0.26; *pro. netIO* = 0.37; *pro. end* = 0.37.

5.4　不同异构程度下算法均衡能力评价

利用 SPNP 软件包对图 5 模型的 BM 节点分别在不同的资源异构程度下进行仿真实验, 实验中能力最大的 BM 配置如 5.3 节所述, 另一个 BM 节点参数的选择则根据异构度的不同做了相应比例的缩小.

仿真得到的平衡度/$(H_{\cdot CPU\cdot}B, H_{\cdot net\cdot}, H_{\cdot st\cdot})$ 变化曲线见图 6. 图中的曲线表明, 随着系统资源异构程度的增加, RR 和 SQR 算法的平衡度变化较大, 说明这两种算法对异构集群环境的适应性较差, 不适合在异构集群媒体服务器上使用. 事实上, 由于 RR 从本质上讲没有考虑集群中 BM 之间负载容量的差异, 而 SQR 对 BM 内部结构的部件性能差异不加区别, 因此, 不能对异构集群的负载均衡有良好表现. SAQR、SEDR、LRV 算法的平衡度曲线变化平缓, 是可供选择的均衡算法. 在本文以下的内容中, 只针对这三种算法做进一步比较和讨论.



Parameters of the strongest BM: p=60, ω=15, r=60, τ=48, η=12, n=42, ρ=18, θ=32, pro.diskIO=0.26, pro. end=0.37, P_limit=43, λ=15.5

图 6　不同异构程度下各算法均衡能力比较

5.5　平均吞吐量和平均响应时间对比

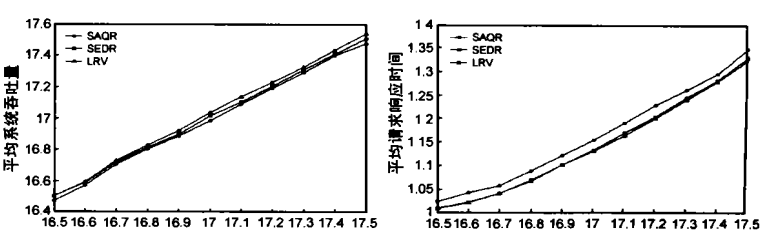
在 $H_{\cdot CPU\cdot}$、$H_{\cdot net\cdot}$、$H_{\cdot st\cdot}$ 为 0.2 的情况下, 利用 SPNP 软件包进一步对 SAQR、SEDR、LRV 算法下的图 5 模型进行不同请求到达速率的仿真实验. 仿真得到的曲线见图 7.

图中的曲线表明, LRV 在吞吐量上比另外两种算法都高, SAQR 算法的吞吐量最低. 在响应时间上, SAQR 比另外两种算法都高, LRV 和 SEDR 算法在响应时间上几乎是接近的. 这种结果可以这样解释: SAQR 算法忽视了系统中各时间变迁的速率因素, 而时间变迁的速率是系统性能参数中的重要组成部分. LRV 和 SEDS 都考虑了时间变迁的速率, 但 LRV 突出了对各种资源的协调使用, 因此更容易实现对资源的充分利用, 达到更高的吞吐量.

5.6　性能平稳性

对性能平稳性的考察所使用的模型参数与 5.5 中相同, 只是扩大了请求到达速率的范围. 仿真得到的系统吞吐量和响应时间曲线见图 8. 曲线图表明, SAQR 和 SEDR 在请求到达速率分别 17.5 和 17.7 时出现了系统性能的急剧下降, LRV 算法不但延长了系统平稳运行的范围, 并且当系统出现性能下降的时候, 下降是平缓的. 因此 LRV 的平稳性要好于另外两种算法.

LRV 算法的平稳性主要来源于限制向接近临界状态的 BM 分配请求这一步骤所做的贡献.



Parameters of BM1: p=60, ω=15, r=60, τ=48, η=12, n=42, ρ=18, θ=32, pro.diskIO=0.26, pro.netIO=0.37, pro.end=0.37, P_limit=43

Parameters of BM2: p=36, ω=9, r=36, τ=29, η=7.2, n=25, ρ=10.8, θ=19.2, pro.diskIO=0.26, pro.netIO=0.37, pro.end=0.37, P_limit=27

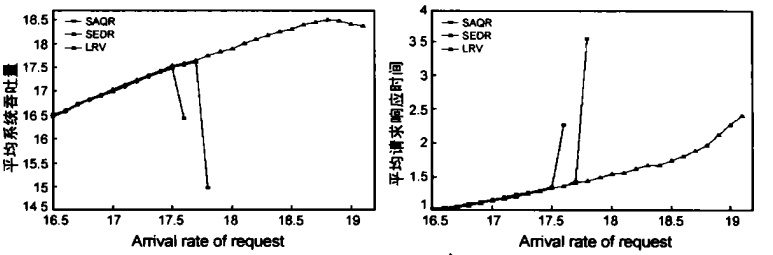
图 7　系统吞吐量和响应时间曲线图

6 结论

基于软交换的集群媒体服务器是 NGN 网络中的重要设备,针对该设备的系统特征,本文提出了一种 LRV 负载均衡算法,LRV 算法考虑了异构集群系统中资源的协调使用并可以有效防止系统性能的剧烈震荡,达到了较好的均衡效果. 仿真结果表明,LRV 负载均衡算法比最小系统中正处理请求总数优先算法和最小请求等待处理期望时间优先算法拥有更好的性能,尤其是在平稳性方面,LRV 负载均衡算法要远远优越于其他传统算法.

由于媒体服务器上内存资源的使用比较复杂,目前算法考虑的资源因素只限于 CPU、网络子系统和本地存储子系统,在完整性上 LRV 算法需进一步完善.

LRV 负载均衡算法对其他资源异构集群系统的负载均衡也具有重要的应用价值.



Parameters of BM1: p=60,ω=15,r=60,τ=48,η=12,n=42,ρ=18,θ=32,pro.diskIO=0.26, pro.netIO=0.37,pro.end=0.37,P_limit=43

Parameters of BM2: p=36,ω=9,r=36,τ=29,η=7.2,n=25,ρ=10.8,θ=19.2,pro.diskIO=0.26, pro.netIO=0.37,pro.end=0.37,P_limit=27

图 8 系统吞吐量和响应时间曲线图

参考文献:

[1] 吴乃星,廖建新,徐鹏,朱晓民. 一种基于软交换的集群媒体服务器的系统结构[J]. 电信科学,2004,20(7):11- 15.
Wu Nai-xing, Liao Jian-xin, Zhu Xiao-min. An architecture of softswitch-based clustered media server[J]. Telecommunications Science, 2004, 20(7): 11- 15. (in Chinese)

[2] V Cardellini, M Colajanni, P S Yu. Dynamic load balancing on web server systems[J]. IEEE Internet Computing, 1999, 39(9): 28- 39.

[3] M Calajanni, P S Yu, V Cardellini. Dynamic load balancing in geographically distributed heterogeneous web servers[A]. Proc. of the 18PthP International Conference on Distributed Computing Systems [C]. USA: IEEE Computer Society Press, 1998. 295.

[4] E D Katz, M Butler, R McGrath. A scalable HTTP server: The NCSA prototype[J]. T Computer Networks and ISDN Systems, 1994, 27(2): 155- 164.

[5] 邱烁,郑纬民,王鼎兴等. 并行 WWW 服务器集群请求分配算法的研究[J]. 软件学报,1999,10(7):713- 718.
Di Suo, Zheng Wei-min, Wang Ding-xing. Research on request dispatching algorithm for web server clusters[J]. Journal of Software, 1999, 10(7): 713- 718. (in Chinese)

[6] 周幼英,李福超,雷迎春. 关于调度算法与 Web 集群性能的分析[J]. 计算机研究与发展,2003,140(3):483- 492.
Zhou You-ying, Li Fu-chao, Lei Ying-chun. Analysis of relationship between scheduling algorithm and the performance of web cluster servers [J]. Journal of Computer Research and Development, 2003, 140(3): 483- 492. (in Chinese)

[7] Xiao L, Zhang X, Qu Y. Effective load sharing on heterogeneous networks of workstation[A]. Proc. of the 2000 Inte4rnational Parallel and Distributed Processing Symposium, (IPDPS' 2000), USA: IEEE Computer Society Press, 2000. 431.

[8] 胡道元,著. 计算机网络[M]. 北京: 清华大学出版社, 1999. 87.

[9] 林闯. Web 服务器请求分配和选择的性能分析[J]. 计算机学报. 2000, 23(5): 500- 508.
Lin Chuang. Performance analysis of request dispatching and selectiong in web server cluster[J]. Chinese J Computer, 2000, 23(5): 500- 508. (in Chinese)

[10] Ciardo G, Muppala J K, Trivedi K S. SPNP: stochastic Petri net package [A]. Proc of the Petri nets and performance models[C]. USA: IEEE Computer Society Press, 1989. 142- 151.

作者简介:

吴乃星 男,1974 年出生于江西石城,2005 年毕业于北京邮电大学,获博士学位,现任职于中国联通深圳分公司. 主要研究方向: 下一代网络技术,电信网综合运营支撑系统.
E-mail: wunaixing_p@163.com.

廖建新 男,1965 年出生于四川省宜宾市,北京邮电大学教授,博士生导师,主要研究方向为移动智能网,宽带 IP 智能网,下一代网络技术.

杨孟辉 男,1970 年生于湖南桃江,2005 年毕业于北京邮电大学,获博士学位,主要研究方向: 下一代网络技术.

朱晓民 男,1974 年出生于浙江义乌,北京邮电大学副研究员,博士,主要研究方向为智能网、下一代业务网络.