

基于时间序列的服务器负载预测

杨伟^{1,2}, 朱巧明^{1,2}, 李培峰^{1,2}, 钱培德^{1,2}

(1. 苏州大学计算机科学与技术学院, 苏州 215006; 2. 江苏省计算机信息处理技术重点实验室, 苏州 215006)

摘要: 在负载均衡集群中, 为了实现高效的动态负载均衡, 需要对服务器的负载进行有效的预测。在介绍服务器负载特性和时间序列的3种模型的基础上, 根据服务器负载动态变化的规律, 提出了基于时间序列的服务器负载预测方法, 并利用这一方法对IP网关服务器的负载进行了预测, 实验结果证明该方法具有较好的实用效果。

关键词: 负载特性; 时间序列; 负载预测; ARMA

Server Load Prediction Based on Time Series

YANG Wei^{1,2}, ZHU Qiaoming^{1,2}, LI Peifeng^{1,2}, QIAN Peide^{1,2}

(1. School of Computer Science & Technology, Soochow University, Suzhou 215006;

2. Key Lab of Computer Information Processing of Jiangsu Province, Suzhou 215006)

【Abstract】 In order to realize an effectively dynamic load balance in load balance cluster, it is necessary to predict server load accurately. This article introduces the properties of server load and three models of time series, then proposes the method of server load prediction, time series based on the regularity of load's dynamic varieties, and applies this method to predict load of IP control gateway server. It is proved by the experiment that the predicting method has highly effectiveness in practice.

【Key words】 Load properties; Time series; Load prediction; ARMA

随着计算机网络的迅速发展, 各种服务器越来越难以满足用户的需求。负载均衡集群的出现, 在一定程度上解决了这个问题。负载均衡算法作为影响负载均衡的三大要素之一, 在很大程度上决定了负载均衡集群的性能。但是目前负载均衡集群大多采用的是静态的负载均衡算法, 由于这种算法没有考虑到服务器的当前负载情况, 很难做到真正的负载均衡; 当然也有一些负载均衡集群采用的是动态负载均衡算法, 这种算法一方面使各个服务器之间的负载更加均衡, 另一方面又造成了服务器的资源的巨大浪费, 因为各个服务器要在很小的时间间隔内(甚至是实时的)就要向前端调度器发送自身的负载信息。

基于以上原因, 本文提出了基于时间序列的服务器的负载预测方法。目前, 国内外利用时间序列等其他方法进行预测的研究主要集中在网络流量的预测上, 对服务器负载预测研究的相对较少。二者的不同主要在于: 网络流量预测关心的是TCP/IP流量大小和延迟, 而服务器负载预测关心的是服务器的平均负载量, 这是一个服务器消耗资源的综合体, 包括CPU利用率、网络利用率、内存使用情况以及硬盘可使用空间。文献[1]提出了基于ARMA模型的网络流量预测的方法, 但是它并没有为实际序列寻找拟合最好的时间序列, 而是采用了ARMA(2, 1)序列。这样虽然减少了计算量, 但同时也降低了预测的准确性。本文根据服务器负载的特性以及负载动态变化的规律, 提出了基于时间序列的负载预测方法。此方法不仅为实现高效的负载均衡打下了基础, 而且提高了服务器的资源利用率。

1 负载特性

为了能够进行准确的负载预测, 就要知道如何准确的衡量负载, 即了解负载的特性。美国卡内基梅隆大学的 Peter A.

Dinda 在 1997 年和 1998 年分两次对 38 台不同的机器, 包括集群服务器、计算服务器和桌面主机等进行长期的跟踪抽样, 获得了大量的负载图样。通过对这些图样进行统计分析, Dinda 总结出了负载的特性, 这些研究成果收录于文献[2]。从这篇文献中, 可以得出负载的特性主要有以下 7 个方面:

(1) 负载的变化是一种随机过程。

(2) 负载一般处于较低的水平, 但具有很强的波动性。

(3) 平均负载量高的其负载的绝对波动量也大, 即其最大负载量与最小负载量间的差值大; 但从相对波动量来说, 平均负载量高的要低于平均负载量低的。这种最大负载量与平均负载量的关联性表明对重负载机器进行负载预测有更高的价值。

(4) 负载值的分布是比较复杂的, 尤其是平均负载量高的, 其负载值呈现复合多样性。这种分布复杂性表明对负载进行分析、预测应该是图样驱动, 而不能简单地依据分布曲线。

(5) 负载随时间变化有很强的关联性, 即过去的负载值对将来的负载值有很大的影响。这表明依据过去的负载值对将来的负载值有很大的影响, 同样依据过去的负载值进行负载预测是可行的, 而且利用时间序列模型进行负载预测是合适的。

(6) 负载的变化又有高度的自相似性, 即在所有的时间尺度下负载的变化既复杂, 又具有长期的依赖性, 因此负载

作者简介: 杨伟(1981-), 男, 硕士生, 主研方向: 信息处理, 网络安全; 朱巧明, 教授; 李培峰, 博士生、副教授; 钱培德, 教授、博导

收稿日期: 2005-12-14 **E-mail:** qmzhu@suda.edu.cn

的模型化和预测是困难的。

(7) 负载的变化又具有突变性。负载值在某一个时刻会有比较大的波动,然后又趋于平稳。负载的这种突变性表明负载预测应具有修正和重适应机制。

2 时间序列

由上一节可知,服务器负载的变化是一种时间序列,而且具有高度的自相似性,因此利用时间序列对服务器的负载进行预测是可行的。

时间序列是按时间次序排列的随机变量序列。任何时间序列经过合理的函数变换后都可以被认为是由趋势项部分、周期项部分和随机噪声项部分等 3 部分叠加而成的。它主要包括自回归模型 AR(p)、滑动平均模型 MA(q)、自回归滑动平均模型 ARMA(p, q) 3 种简单的模型,其中,AR(p)和 MA(q)都是 ARMA(p, q)的特殊情况。

定义 设 $\{\varepsilon_t\}$ 是白噪声 $WN(0, \sigma^2)$, 实系数多项式 $A(z)$ 和 $B(z)$ 没有公共根, 满足 $b_0 = 1, a_p b_q \neq 0$ 和

$$A(z) = 1 - \sum_{j=1}^p a_j z^j \neq 0, |z| \leq 1,$$

$$B(z) = \sum_{j=0}^q b_j z^j \neq 0, |z| < 1$$

称差分方程 $X_t = \sum_{j=1}^p a_j X_{t-j} + \sum_{j=0}^q b_j \varepsilon_{t-j}, t \in Z$ 是一个自回

归滑动平均模型,简称 ARMA(p, q)模型。当 $q=0$ 时,它为 AR(p)模型;当 $p=0$ 时,它为 MA(q)模型。与 AR、MA 模型相比,ARMA 模型具有更小的预测误差方差;与 AR 模型,MA 模型相比具有相对稳定的特点;另外 ARMA、AR、MA 模型都可以进行短期预测,并且预测步数越少,误差越小,这一点将会在下面涉及到。

3 实验

(1) 数据采集

下面以 IP 网关服务器为例对其负载进行预测,IP 网关服务器运行在 Red Hat Linux AS 4 平台上,主要实现控制校园网(或园区网)用户访问的站点、记录用户访问的站点信息、采集用户流量等功能。在 IP 网关服务器中连续采集 5 周的负载数据,每周采集 7 个工作日,采样的时间间隔为 5min,通过 /proc 文件系统收集 IP 网关服务器的负载值(用百分数表示)作为观测值。此处的负载是一个百分数,它表示系统的 CPU、网络带宽、内存、硬盘等的综合使用情况。尽管校园网(或园区网)中单个用户的网络行为是不确定的,但是所有的校园网(或园区网)用户的网络行为促使 IP 网关的负载值变化呈现一定的周期性和时间关联性。

(2) 数据的预处理

在采集的负载数据中,肯定会存在一些含有粗大误差的数据,这些数据会扭曲最终结果。因此,要把这些含有粗大误差的数据剔除。在 5 周的相同工作日的相同时刻有 5 个不同的数据 x_i , 其中 $i=1,2,3,4,5$ 。对这 5 个数据计算平均值 \bar{x} , 然后求出它们的标准差 v 。即

$$\bar{x} = \frac{1}{5} \sum_{i=1}^5 x_i$$

$$v = \sqrt{\frac{1}{5} \sum_{i=1}^5 (x_i - \bar{x})^2}$$

根据格拉布斯准则,如果 x_i 满足 $|x_i - \bar{x}| > kv$, 则 x_i 为坏值,应该剔除不用,其中 k 为格拉布斯准则系数,与置信区间 95%

相对应的 $k=2.03$ 。这样就得到正常的 IP 网关负载值变化曲线,如图 1 所示。

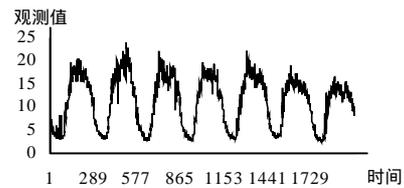


图 1 预处理后的负载变化

(3) 平稳化数据序列

首先要判定经过预处理后的观测值序列是否平稳,这里利用自相关函数法进行判定。当然,通过观测图 1 也可以初步判定观测值序列具有明显的周期性,不是一个稳定的时间序列。图 2 是观测值序列的自相关函数。可以看出,该序列的自相关函数呈现出周期性缓慢衰减的现象,由此可以判定该序列存在某种趋势或周期,所以不是稳定的。

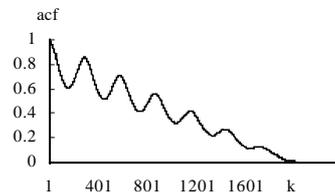


图 2 样本自相关函数

平稳化的目的就是去除时间序列中的趋势项和周期项,为建立时间序列模型做准备。本文将采用方差分析的方法进行平稳化处理。具体来讲,用 V_{ij} 表示一周当中第 i 天第 j 个 5 分钟的观测值,其中 $i=1,2,3 \dots 7; j=1, 2, \dots, 288$ 。首先对 V_{ij} 取对数,这样可以有效地减少标准差,然后把 $\ln V_{ij}$ 划分为趋势项 T_j 、周期项 S_i (季节项)、随机噪声项 3 部分,即 $\ln V_{ij} = T_j + S_i + R_{ij}$, 其中, T_j 表示在第 j 个时刻的平均值, S_i 表示第 i 天的平均值, R_{ij} 是观测值的残余。这样,就把一周当中不同的工作日以及每天的不同时刻这两个影响因素都从原观测值 V_{ij} 中剔除掉了。如果服务器的负载在周与周之间的变化比较大的话,也可以把不同周的影响剔除掉。

经过平稳化处理以后,就得到了如图 3 所示的一个平稳的时间序列 R_{ij} 。从图中可以看出,序列 R_{ij} 没有明显的趋势性和周期性,是一个平稳的时间序列。

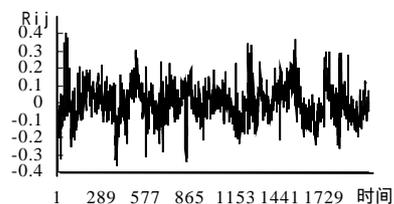


图 3 平稳化的时间序列

(4) 模型识别

对经过平稳化处理得到的时间序列 R_{ij} 进行模型识别,即确定该时间序列拟合 AR、MA、ARMA 中的哪一个模型最好。本文采用 Box-Jenkins 的模型识别方法,即根据自相关函数、偏自相关函数的截尾性和拖尾性判定时间序列 R_{ij} 所拟合的模型类型。时间序列 R_{ij} 的自相关函数、偏自相关函数分别见

图 4 和图 5。

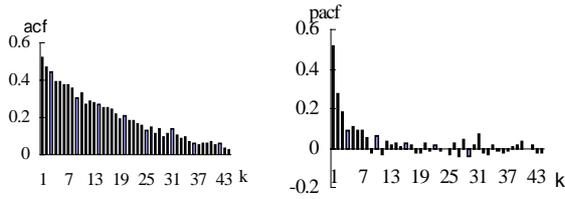


图 4 自相关函数

图 5 偏自相关函数

由图 4 和图 5 可以看出, 时间序列 R_{ij} 的自相关函数和偏自相关函数都不截尾, 而且都被收敛到 0, 因此可以判定 R_{ij} 很有可能是 ARMA 序列。

(5) 模型定阶

时间序列 R_{ij} 被识别为 ARMA(p,q) 序列, 下面就要确定 ARMA(p, q) 中的 p、q 的值, 也就是模型定阶。本文采用 AIC 定阶方法, 首先设 P 为 p 的上界, Q 为 q 的上界, 对每一对 (k, j), 其中, $0 \leq k \leq P, 0 \leq j \leq Q$, 计算 AIC 函数 $AIC(k, j) = \ln(\sigma^2(k, j)) + 2(k + j)/N$, 其中, N 为样本总数, σ^2 是残差方差, P、Q 为 30(因为较大的 p、q 会导致较多的待估参数, 其结果会使建立的模型关于数据过于敏感, 降低模型的稳健性)。求得 AIC(k, j) 的最小值, 其中的 k、j 就是 p、q 的值。根据 Pandit-Wu 方法, 对于 ARMA(p, q) 的定阶可以按 ARMA(2n, 2n-1) 的方式进行, 其中 $n \geq 1$ 。图 6 是时间序列 R_{ij} 的 AIC 定阶图。

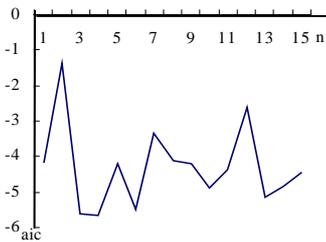


图 6 AIC 定阶图

从图 6 可以看出, 在 $n=4$ 时, AIC(2n, 2n-1) 最小, 因此 $p=8, q=7$, 即时间序列 R_{ij} 为 ARMA(8, 7) 序列。

(6) 参数估计

时间序列 R_{ij} 确定为 ARMA(8, 7) 序列后, 接着就要确定在第 3 节提到的 a_j, b_j 的值, 即 $X_t - \sum_{j=1}^8 a_j X_{t-j} = \varepsilon_t - \sum_{j=1}^7 b_j \varepsilon_{t-j}$; 然后用矩估计方法, 求得参数:

$$\begin{aligned} a_1 &= 0.220230 & a_2 &= 0.089872 & a_3 &= 0.165754 & a_4 &= -0.268652 \\ a_5 &= 0.724933 & a_6 &= -0.761566 & a_7 &= 0.493517 & a_8 &= 0.160131 \\ b_1 &= 0.0496127 & b_2 &= 0.098379 & b_3 &= 0.043374 & b_4 &= 0.027579 \\ b_5 &= 0.209916 & b_6 &= -0.134451 & b_7 &= -0.071675 & \sigma^2 &= 0.047347 \end{aligned}$$

接下来就要对参数进行平稳性判定, 也就是求解 ARMA(p, q) 模型定义中的 $A(z)=0$ 和 $B(z)=0$ 的根, 并判断这些根是否在单位圆外, 如果在单位圆外, 则 ARMA(8, 7) 是平稳的时间序列; 否则就是不稳定的。经过计算可以得到 $A(z)=0$ 和 $B(z)=0$ 的根都在单位圆外, 因此 ARMA(8, 7) 是平稳的时间序列。

(7) 预测

ARMA(8,7) 模型的预测就是利用 ARMA 的时间相关性对未来某一时点的负载值进行预测, 那么 $t+1$ 步预测值就

等于 ARMA(8,7) 模型中以 a_j 为参数的前 8 步负载值之和加上其以 b_j 为参数的 7 步预测的所有预测误差之和。下面对服务器的负载值进行平稳线性最小方差预测。这里采用模型递推法。

若线性最小均方误差为 μ , l 为向后预测步数, 可得 l 步预测的最小均方误差是 $\mu^2(l) = \sigma^2(G_0^2 + G_1^2 + \dots + G_{l-1}^2)$, 其中 $G_0, G_1, G_2, \dots, G_l$ 是 ARMA(8, 7) 的格林函数。可见, $\mu^2(l)$ 与预测时间无关, 只与预测的步数有关, 预测的步数越多, 即预测的时间越远, 方差越大, 预测的结果与实际值的差别越大, 越不可预测。也正是由于这个原因, 在接下来的预测中, 采用的就是 1 步预测的模型递推法, 来降低预测误差。

根据 ARMA(8,7) 模型 $X_t = \sum_{i=1}^8 X_{t-i} + \varepsilon_t - \sum_{i=1}^7 \varepsilon_{t-i}$, 可得:

$$\varepsilon_t = \sum_{i=0}^8 X_{t-i} - \sum_{i=1}^7 \varepsilon_{t-i}$$

同理可得:

$$\varepsilon_{t-1} = \sum_{i=0}^8 X_{t-1-i} - \sum_{i=1}^7 \varepsilon_{t-1-i}$$

...

$$\varepsilon_{t-7} = \sum_{i=0}^8 X_{t-7-i} - \sum_{i=1}^7 \varepsilon_{t-7-i}$$

令 $\varepsilon_{t-8}, \varepsilon_{t-9}, \dots, \varepsilon_{t-14} = 0.0$, 因此 1 步预测值

$$X_t(1) = \sum_{i=1}^8 X_{t-i} + \varepsilon_t - \sum_{i=1}^7 \varepsilon_{t-i}$$

从以上几个表达式可以看出, 1 步预测的模型递推法只需要预测之前的 15 个观测值。这样, 把得到的下一个时间点的预测值 X_{t+1} (即随机项 $R_{[(t+1)/288][(t+1)\%288]}$)、下一个时间点的趋势项 $T_{(t+1)\%288}$ 和当天的周期项 $S_{(t+1)/288}$ 加在一起, 然后把这个值作为以 e 为底的幂次, 最后求得的就是这一个下一个时间点的服务器负载的预测值。图 7 就是利用 ARMA(8,7) 模型的 1 步预测的模型递推法产生的一周的预测值, 图 7 是 50 个预测值与实际值的比较图, 其中实线是实际值, 虚线为 1 步预测值。

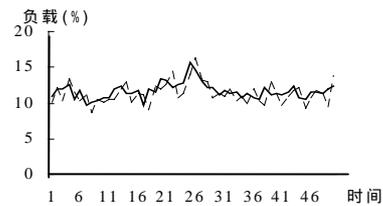


图 7 预测值与实际值比较

(8) 结果分析

利用 ARMA(8,7) 模型对一个周的负载进行了预测。为了衡量预测的效果, 这里引入了绝对误差、相对误差、最大绝对误差、最小绝对误差、平均相对误差等几个概念。

绝对误差=预测值-实际值;

相对误差=绝对误差/实际值;

最大绝对误差=max(abs(绝对误差));

最小绝对误差=max(abs(绝对误差));

平均相对误差=max(abs(相对误差)).

对负载的预测值与实际值的计算可以得出, 平均相对误

(下转第 148 页)