·开发研究与工程应用· 文章编号：1000-3428(2016)04-0295-06 文献标识码：A 中图分类号：TP311

基于全级 C 阶矩模型并行流数预测的广域大数据吞吐量优化

李 芝，龙 敏

（长沙理工大学计算机与通信工程学院，长沙 410114）

摘 要： 针对传统大数据密集型的可扩展计算系统在数据源利用和数据传输方面效率不高的问题，提出基于并行流数预测的应用层吞吐量优化模型。为提高并行流数预测精度，以提高瓶颈链路的利用效率为目的，设计等效并行流数选取方式。借鉴部分 C 阶矩模型和完全二阶矩模型，构建全级 C 阶矩模型，并且设计低采样吞吐量优化框架，降低计算复杂度。在不同大小数据集上的实验结果表明，全级 C 阶矩并行流数的预测模型更适合大数据传输，且效率更高。

关键词： C 阶矩模型；二阶矩模型；大数据；并行流数预测；吞吐量

中文引用格式： 李 芝，龙 敏. 基于全级 C 阶矩模型并行流数预测的广域大数据吞吐量优化[J]. 计算机工程，2016，42(4)：295-300，306.

英文引用格式： Li Zhi，Long Min. Wide Area Big Data Throughput Optimization Based on Full C-order Moment Model with Parallel Flow Prediction[J]. Computer Engineering，2016，42(4)：295-300，306.

Wide Area Big Data Throughput Optimization Based on Full C-order Moment Model with Parallel Flow Prediction

LI Zhi，LONG Min

（School of Computer and Communication Engineering，Changsha University of Science and Technology，Changsha 410114，China）

【Abstract】 In order to solve the problem of low efficiency in the use of data sources and data transmission of the traditional data intensive scalable computing systems，this paper proposes a big data throughput optimization algorithm based on full C-order moment model with parallel flow prediction. To improve the prediction accuracy of parallel flow，it takes the utilization efficiency improvement of the bottleneck link for the purpose，and designs the method of equivalent parallel flow algorithm. According to the partial C-order moment model and full second-order moment model，it constructs the full C-order moment model，and designs the low sample throughput optimization framework，which can reduce the computational complexity. Experimental results on the data set with different size show that the full C-order moment model with parallel flow is more suitable for the transmission of big data，and more efficient.

【Key words】 C-order moment model；second-order moment model；big data；parallel flow prediction；throughput

DOI： 10.3969/j.issn.1000-3428.2016.04.050

1 概述

近年来，随着各领域科技水平的日新月异，产生了越来越复杂和庞大的存储数据。当前大数据领域的发展趋势，主要围绕实用性、高性能以及可恢复性为目的[1-2]。在数据传输方面，采用光纤网络，比如 XSEDE，Esnet，以及 Internet2 等方式[3]，为用户提供高速链接，以降低数据传输瓶颈的影响。特别是，随着网络技术的发展，科学家设计提供了容量达到 100 Gb/s的高速光纤宽带。

即便如此，由于各种限制因素仍然存在，比如次优协议优化、低效的端到端路由、发送/接收端磁盘性能瓶颈以及服务器处理器性能局限等因素，制约了用户获取理论上的高速网络。例如，文献[4]指出从波士顿寄送 1 TB 的取证数据，亚马逊的 S3 存储系统需要花几个星期的时间。基于该原因，许多用户宁可使用快递数据邮寄来取代网络传输。在此背景下，研究人员开发了如 Pandora 等合作互联协议[5]，这些系统收集互联网和运输环节可用信息，并利用这些数据作为算法输入，求解最优的数据传输

基金项目： 湖南省自然科学基金资助项目(2015JJ2007)；湖南省研究生科研创新基金资助项目(CX2013B376)。

作者简介： 李 芝(1990－)，女，硕士研究生，主研方向为信息安全、云存储安全；龙 敏，教授、博士。

收稿日期： 2015-04-07 **修回日期：** 2015-05-03 **E-mail：** lgdlls@163.com

方案。以上所述表明,高速网络非常重要,但并非充分条件,如何更为有效地使用高速网络处理数据密集型可扩展计算系统和动态数据驱动的应用程序共享等问题,变得越来越重要和紧迫。

在大数据传输中,并行 TCP 数据流具有更高效的数据传输性能,平行数据流通过模仿单个数据流行为,进而获得更高的可用带宽份额[6]。不过由于一些参数在时空域上的独立性,导致网络拥塞点无法预测。并行流数选取因此变得非常困难,并且依赖于可用带宽、丢包率、瓶颈链路容量、数据大小等网络参数。

在并行数据流数预测方面,有学者提出预测模型,如文献[7]提出全级二阶矩网络吞吐量预测模型,并结合 GridFTP 数据传输系统进行验证。文献[8]在此基础上提出部分 C 阶矩网络吞吐量预测模型,同样在 GridFTP 数据传输系统上进行了验证。文献[7]指出全级二阶矩模型比部分二阶矩模型预测精度要高。基于该思想,本文对部分 C 阶矩网络吞吐量预测模型进行研究,并推导出全级 C 阶矩网络吞吐量预测模型。但在具体实验测试中发现,全级 C 阶矩网络吞吐量预测模型由于增加了网络预测参数,导致算法的实时性不好,针对该问题本文同步设计了低采样并行流数预测方案。同时,为验证所提算法的有效性,在仿真实验环节,参照有关文献做法,在 GridFTP 数据传输系统中,对相关模型的预测性能进行仿真验证。

2 相关研究

2.1 问题描述

以往文献中对于该问题的研究多基于近似理论模型,通常会事先给定具体的假设和限制条件,脱离实际条件,因此这些研究仅可在实验条件下进行验证,实际中较难应用。文献[9]指出大数据流量预测类似于河流预测,可采用并行化预测方式,如下式所示。

$$Th_n \leqslant Mss \times c \times n/(RTT \times \sqrt{p}) \tag{1}$$

其中,Mss 为最大分割段大小;RTT 为往返时间;n 为并行数据流数;p 为丢包率;c 为固定值。但是该模型设计初衷主要是针对拥堵网络,应用到普通网络中对网络性能提升有限。

文献[10]采用与文献[9]相似的理论,但是设计了平等分流协议。并在此基础上指出,随数据流数目增加,总吞吐量在数据连接能力上始终显示相同的特性,3 条数据流足以获得 90% 以上的资源利用率,如下式所示。

$$Th_n = C\left(1 - \frac{1-B}{(1-B)+(1+B)n}\right) \tag{2}$$

其中,B 为 TCP 连接数量的 1/2;C 为网络链路容量;n 是并行流数目。

上述模型均无法较好地平衡预测精度提高与采样开销增大的矛盾,为此本文工作的主要目的是,实现低采样的同时提高数据的预测精度。在文献[7-8]所述的部分二阶矩、完全二阶矩和部分 C 级模型中,会用到 1 个 ~3 个采样点,但如何选取最佳的采样点,上述文献并未考虑。若采取随机选取采样点的方式,会导致预测结果失常。为此可通过随机选取多个采样点,然后从中选取 1 个 ~3 个最佳采样点的方式进行改进,但增加了采样开销。本文设计全级 C 阶模型(Full C-order Model,FCOM)并行流数预测实现大数据吞吐量的优化。

2.2 全级二阶矩模型

文献[7]提出部分二阶矩模型,根据式(1)可获得式(3),形式如下:

$$\begin{cases} p'_n = p_n \cdot RTT_n^2/(c^2 \cdot Mss^2) = a'n^2 + b' \\ Th_n = n/\sqrt{p'_n} = n/\sqrt{a'n^2 + b'} \end{cases} \tag{3}$$

其中,n 定义同前;Th_n 为网络吞吐量。需要计算 a' 和 b'对曲线进行拟合,在此情形下,需要 2 条不同的并行流数进行曲线拟合;p'_n 为丢包率。

文献[7]对上述部分模型进行了改进,通过增加参数,提出全级二阶矩模型,形式如下式所示。

$$\begin{cases} p'_n = p_n \cdot RTT_n^2/(c^2 \cdot Mss^2) = a'n^2 + b'n + c' \\ Th_n = n/\sqrt{p'_n} = n/\sqrt{a'n^2 + b'n + c'} \end{cases} \tag{4}$$

其中,需要拟合参数增加为 3 个:a',b'及 c'。式中,参数 Th_n 和 n 已知,但拟合参数未知,其迭代计算形式如下:

$$\begin{aligned} a' &= \frac{1}{Th_{n_1}^2(n_3 - n_2)} \cdot \left(\frac{n_3^2 Th_{n_1}^2 - n_1^2 Th_{n_3}^2}{Th_{n_3}^2(n_3 - n_1)} - \frac{n_2^2 Th_{n_1}^2 - n_1^2 Th_{n_2}^2}{Th_{n_2}^2(n_2 - n_1)}\right) \\ b' &= \frac{n_2^2 Th_{n_1}^2 - n_1^2 Th_{n_2}^2}{Th_{n_1}^2 Th_{n_2}^2(n_2 - n_1)} - (n_2 + n_1)a' \\ c' &= \frac{n_1^2}{Th_{n_1}^2} - n_1^2 a' - n_1 b' \end{aligned} \tag{5}$$

2.3 部分 C 阶矩模型

文献[8]提出一种改进模型:部分 C 阶矩模型,即模型中的阶矩参数 c'是未知的。模型形式如下:

$$\begin{cases} p'_n = = a'n^{c'} + b' \\ Th_n = n/\sqrt{a'n^{c'} + b'} \end{cases} \tag{6}$$

类似前述二阶矩模型,参数 Th^{n_1},Th^{n_2},Th^{n_3},n_1,n_2 及 n_3 均已知,而模型系数 a',b'及 c'均未知。基

于上述公式,可得到未知模型系数的曲线:

$$(n_2^{c'}-n_1^{c'})\left(\frac{n_3^2}{Th_{n_3}^2}-\frac{n_1^2}{Th_{n_1}^2}\right)=(n_3^{c'}-n_1^{c'})\left(\frac{n_2^2}{Th_{n_2}^2}-\frac{n_1^2}{Th_{n_1}^2}\right) \tag{7}$$

$$a'=\frac{Th_{n_1}^2n_2^2-Th_{n_2}^2n_1^2}{Th_{n_1}^2Th_{n_2}^2(n_2^{c'}-n_1^{c'})} \tag{8}$$

$$b'=n_1^2/Th_{n_1}^2-a'n_1^{c'} \tag{9}$$

对于上述模型求解,无法像二阶矩模型求解方式通过线性方式推导出 c'切线,需要借助牛顿迭代法对 c'进行近似求解。

$$c'_{x+1}=c'_x-f(c'_x)/f'(c'_x) \tag{10}$$

3　低采样高精度吞吐量预测

3.1　全级 C 阶矩模型

如前所述,第 2 节模型在选取最佳采样点方面存在无法与采样开销平衡的问题。若简单地采取随机采样方式,采样点很可能不能准确反映出曲线特性,导致预测精度过低;相反如果顾及最佳采样点问题,从多个随机采样点中选取最佳采样点,则会导致过多的采样开销。

在采样点选取结束后,根据式(1)利用最大分割段大小 Mss、往返时间 RTT、丢包率 p,计算网络吞吐量。并基于全级二阶矩模型和部分 C 阶矩模型,构建新的全级 C 阶矩模型形式如下:

$$\begin{cases}p'_n=p_n\dfrac{RTT^2}{c^2MSS^2}=a'n^{c'}+d'n+b'\\ Th_n=\dfrac{n}{\sqrt{p'_n}}=\dfrac{n}{\sqrt{a'n^{c'}+d'n+b'}}\end{cases} \tag{11}$$

其中,出现 4 个未知参数:a',b',c'和 d',已知参数有 8 个:Th_{n_1},Th_{n_2},Th_{n_3},Th_{n_4}以及 $n_1\sim n_4$。

为得到未知参数 a',b',c'和 d'的值,需要知道 4 个并行流等级的吞吐量,可由网络测量工具或过去的数据传输数据进行预测,在已知参数中,$Th_{n_1}\sim Th_{n_4}$与 $n_1\sim n_4$ 的关系表达式如下:

$$\begin{cases}Th_{n_1}=\dfrac{n_1}{\sqrt{a'n_1^2+b'n+c'}}\\ Th_{n_2}=\dfrac{n_2}{\sqrt{a'n_2^2+b'n+c'}}\\ Th_{n_3}=\dfrac{n_3}{\sqrt{a'n_3^2+b'n+c'}}\\ Th_{n_4}=\dfrac{n_4}{\sqrt{a'n_4^2+b'n+c'}}\end{cases} \tag{12}$$

联合上述方程,可得模型系数 a',b'和 d'表达式如下:

$$a'=\frac{\dfrac{n_3^2(n_2-n_1)}{Th_{n_3}^2}+\dfrac{n_2^2(n_3-n_1)}{Th_{n_2}^2}+\dfrac{n_1^2(n_3-n_2)}{Th_{n_1}^2}}{n_3^{c'}(n_2-n_1)-n_2^{c'}(n_3-n_1)-n_1^{c'}(n_3-n_2)} \tag{13}$$

$$d'=\frac{\dfrac{n_2^2}{Th_{n_2}^2}-\dfrac{n_1^2}{Th_{n_1}^2}-a'n_2^{c'}+a'n_1^{c'}}{n_2-n_1} \tag{14}$$

$$b'=\frac{n_1^2}{Th_{n_1}^2}-d'n_1-a'n_1^{c'} \tag{15}$$

因篇幅原因,上述系数求解过程略。在求取模型系数 a',b'和 d'后,借鉴部分 C 阶矩模型方式,采用牛顿迭代法对 c'进行近似求解,如式(10)所示。

3.2　并行流数选取

瓶颈链路的利用效率是制约大数据传输效率的关键因素,这里提出一种新的瓶颈链路并行流数选取机制,实现更为公平的分流方式。目标积压可从当前测量瓶颈进行计算获取,在此计算过程中需要用到如下参数:瓶颈链路的容量,平均最小的往返时间和丢包率。基于上述参数,可获得共享流量瓶颈链路的平均数,计算步骤如下:

步骤 1　单流数据积压计算,$b=S\times d$,其中,S为发送速率;$d=RTT_{\text{avg}}-RTT_{\min}$。

步骤 2　总数据积压计算,$B=d\times 12/C$,其中,C为瓶颈链路容量。

步骤 3　计算瓶颈链路利用率 U:

$$U\approx 2B/(2B+1) \tag{16}$$

步骤 4　计算瓶颈链路等效流量共享均值数 n:

$$n=B/b=(C\times U)/12S \tag{17}$$

在计算得到上述瓶颈链路等效流量共享均值数 n 后,可结合全级 C 阶矩模型,根据需要的瓶颈链路利用率计算需要额外开启的数据流分链路:

$$n_{\text{add}}=n_{\text{opt}}-n \tag{18}$$

然而,对于某些计算信息只能通过低层协议获取,比如发送速率等。

3.3　计算复杂度及模型系数分析

3.3.1　计算复杂度分析

上述全级 C 阶矩模型及其算法框架的提出,主要目的是平衡算法的时效性和高精度间的关系。在每个采样迭代过程中,选取并行数据流组成三元组,当达到终止条件后,算法停止并输出结果。

本文设计的全级 C 阶矩模型,选择的采样数据点为 3 的幂次$(1,3^1,\cdots,3^k)$,这与前述模型选取的 2 的幂次数据点不同,如此选取的原因有:(1)可加速模型收敛;(2)可有效避免终端系统过载,以及过多的平行流网络,如 4 的幂次。

在经过 $k+1$ 次迭代后,会得到 $k+1$ 个并行流数吞吐量数据,应用最优系数拟合算法获得最佳的并行流数。在此过程中,计算复杂度为 $O(\log_3 k)$,相

比于传统 2 的幂次数据点采样的计算复杂度 $O(\mathrm{lb}k)$,算法更为简化,且后续实验证明,这种全级 C 阶矩模型的低采样方式并未降低算法预测精度。

3.3.2 模型系数理论分析

为得到先增大后减小的弧形曲线,需要保证网络吞吐量函数的一阶导数,在曲线增大部分为正值,在顶点处为 0,在曲线下降部分为负值。

$$Th'_{\mathrm{ful}} = \begin{cases} \dfrac{d'n/2 + b'}{(a'n^{c'} + d'n + b')^{3/2}} > 0 & n \text{ 小于顶点位置} \\ \dfrac{d'n/2 + b'}{(a'n^{c'} + d'n + b')^{3/2}} = 0 & n \text{ 等于顶点位置} \\ \dfrac{d'n/2 + b'}{(a'n^{c'} + d'n + b')^{3/2}} < 0 & n \text{ 大于顶点位置} \end{cases} \tag{19}$$

由此可得相关模型系数的取值范围为:

$$\begin{cases} a' > 0, \quad b' < 0 \\ c' > 0, \quad d' < 0 \\ 2c' + b' > 1 \end{cases} \tag{20}$$

因此,并行流数预测下限可由以下极限方式进行求解:

$$\lim_{n\to\infty} \frac{n}{\sqrt{a'n^{c'} + d'n + b'}} = \lim_{n\to\infty} \frac{2\sqrt[c']{a'n^{c'} + d'n + b'}}{2a'n + d'} = \lim_{n\to\infty} \frac{\sqrt{a' + \dfrac{d'n}{n^{c'}} + \dfrac{n^{c'}}{}}}{a' + \dfrac{2n^{c'}}{}} = \frac{\sqrt[c']{a'}}{a'} \tag{21}$$

通过数据流获得的模型系数,可根据上述分析进行判断,是否在合理范围内。若不在合理范围内,可节省不必要的计算时间,有助于算法计算复杂度的降低。

3.4 并行流数吞吐量优化

对于上述模型,在选取合适的并行化流数后,均可获得较为理想的算法性能,但是所有模型都需要 2 个 ~3 个不同数据吞吐量水平的并行数据流。因此,若有超过 3 对 (n, Th_n) 数据组合,如何选取历史或者预测数据与 n 并行流模型计算数据最接近,是算法需要解决的关键性问题。以伪代码形式给出这种最佳组合的选取方式,并且该算法具有很好的可扩展性。

定义伪代码中有关参量如下:(n_l, Th_l) 中 n_l 表示并行流数,Th_l 为与之对应的网络吞吐量,(a', b', c', d') 为全级 C 阶矩模型系数,min(*err*) 为多项式系数的最小平均误差,*CollecData* 表示 $m \times 2$ 数据集,*Getefficient*(·) 以 3 组数据作为输入参数。进行多项式系数预测,该函数中包含 3.3 节中的参数合理性判断,*Errvalue*(·) 基于均方误差计算误差值。

令 ε 表示历史或网络工具采集到的网络吞吐量 $(Th^a_{n_i})$ 与模型预测吞吐量 $(Th(n_i))$ 的距离,可表示为:

$$\varepsilon = Th^a_{n_i} - Th(n_i) \tag{22}$$

则误差值计算公式如下:

$$Err = \sqrt{(\varepsilon_0^2 + \varepsilon_1^2 + \cdots + \varepsilon_{m-1}^2)/m} = \sqrt{\sum_{i=0}^{m-1} (Th^a_{n_i} - Th(n_i))^2/m} \tag{23}$$

算法伪代码如下:

输入 $CollecData[m][2] = \begin{pmatrix} n_0 & n_1 & \cdots & n_i & \cdots \\ Th_0 & Th_1 & \cdots & Th_i & \cdots \end{pmatrix}^{\mathrm{T}}$

输出 $(n_r, Th_r)(n_s, Th_s)(n_t, Th_t)(a', b', c', d')\, err_{\min}$

```
1.  Begin
2.   for i = 0:m - 2
3.    for j = i + 1:m - 1
4.     for k = j + 1:m
5.      a,b,d←Geteffic(CollecData[i],CollecData[j],
CollecData[k])
6.      err←Errvalue(a,b,d,CollecData)
7.      if min(err) is not initialized then
8.      min(err)←err,  r,s,t←i,j,k,  a',b',d'←a,
b,d
9.      elseif err < min(err) then
10.      min(err)←err,  r,s,t←i,j,k,  a',b',c'←a,
b,c
11.     endif
12.     k = k + 1
13.    endfor
14.    j = j + 1
15.   endfor
16.   i = i + 1
17.  endfor
18.  n_r←CollecData[r][0]
19.  n_s←CollecData[s][0]
20.  n_t←CollecData[t][0]
21.  Th_n_r←CollecData[r][1]
22.  Th_n_s←CollecData[s][1]
23.  Th_n_t←CollecData[t][1]
24. end
25. Implement newton iterationc'_x+1 = c'_x - f(c'_x)/f'(c'_x)
```

4 实验结果与分析

本节将基于 GridFTP[11] 和 Stork[12] 2 个数据传输系统进行实验验证,其中,GridFTP 系统服务器位于路易斯安那州立大学和特伦托大学 SuSE 集群,传输数据大小为 512 MB,并行流数设为 $n = 1 \sim 40$。

4.1 GridFTP 数据传输系统

对比算法选取部分二阶矩模型(Partial Second

Order Model,PSOM)、全级二阶矩模型(Full Second Order Model,FSOM)及部分 C 阶矩模型(Partial C Order Model,PCOM)。对于每个样本数据集,计算得出其模型系数,并进行网络吞吐量预测。

4.2　评价指标

定义模型可预见性指标,若该指标较小,则称之为可预见性差。使用 $P_{pre[m\_i]}$ 表示来自数据集 m_i 的特定模型可预见性,其中,m 表示数据数量;i 表示第 i 个数据集。则 $P_{pre[m\_i]}$ 可定义为:

$$P_{pre[m\_i]} = \frac{N_{\text{ind}}}{N_{\text{outd}} + N_{\text{ind}}} \tag{24}$$

其中,N_{ind} 表示预测正确数据量;N_{outd} 为预测不正确数据量。若模型对于数据的可预见性较差,则称其为对数据敏感。从样本数据集的灵敏度分析,令 N_{eff} 为 $P_{pre[m\_i]}=1$ 的样本数量,样本总数量为 N_{sam},则灵敏度指标可定义为:

$$P_{sen[m\_i]} = \frac{N_{\text{eff}}}{N_{\text{sam}}} \tag{25}$$

对于模型精度预测评价,基于距离度量定义如下误差指标:

$$Err_{[m\_i]} = \sqrt{\frac{\sum_{k=1}^{N_{\text{ind}}} \varepsilon_k^2}{N_{\text{ind}}}} \tag{26}$$

则不同并行流数组合的最佳误差率值可定义如下:

$$Err_{best[m\_i]} = \min_{\text{Neff}}(Err_{[m\_i]}) \tag{27}$$

以所有随机选取数据集上实验结果的平均值作为模型性能评价的最终指标,定义如下[13]:

$$\begin{cases} P_{sen[m\_i]ave} = \sum_{i=0}^{k_m-1} P_{sen[m\_i]}/k_m \\ P_{pre[m\_i]ave} = \sum_{i=0}^{k_m-1} P_{pre[m\_i]}/k_m \\ Err_{best[m\_i]ave} = \sum_{i=0}^{k_m-1} Err_{best[m\_i]}/k_m \end{cases} \tag{28}$$

对比算法在上述 3 个评价指标上的仿真对比结果,如图 1(a)~图 1(c)所示。图 1 为 4 种对比算法在所定义的 3 个评价指标上的实验对比情况。图 1(a)为模型灵敏度指标对比结果,该指标定义的主要目的是测试模型对于数据的灵敏程度,即模型的稳定性。从图中可以看出,在同等采样数据数量下,PSOM 和 PCOM 模型的灵敏度较高,说明算法对于数据的依赖性较强,稳定性差,而 FSOM 和 FCOM 模型灵敏度较低,算法的稳定性较好,并且采样数据越多,所有模型的稳定性均增加。图 1(b)为模型可预见性指标,该指标主要评价模型的预测性能,从图中可以看出,FCOM 模型在该指标上的性能最好,所有算法随采样数据数量增加,均出现该指标增大的现象,说明采样数据增大,有助于提高算法预测性能。图 1(c)为模型预测误差结果对比,该指标直接反映出模型的预测精度,从图中可以看出,FCOM 模型在该指标上的数值最小,说明其预测精度最高。

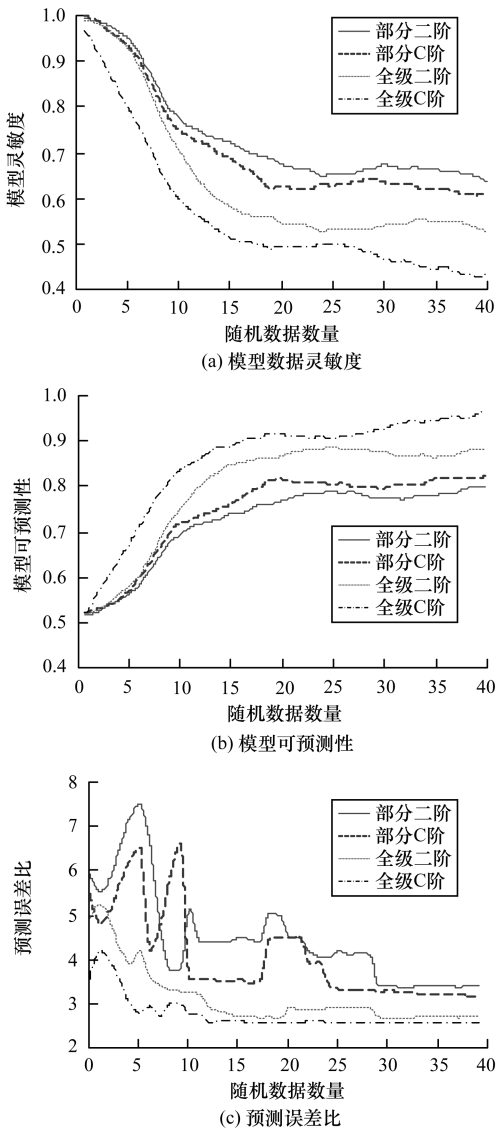

图 1　模型性能测试

从该指标收敛曲线可看出,随着采样数据集的增大,所有模型的预测精度均增加,但会达到饱和,对比几种模型的饱和数据点,可看出 FCOM 模型饱和点在 10 左右,FSOM 模型饱和点在 15 左右,PCOM 模型饱和点在 25 左右,PSOM 模型饱和点在 30 左右。饱和点小说明模型可在较少采样数据数量前提下,取得更好的预测精度。因此,PCOM 模型更适合于稀疏采样点下网络吞吐量预测。

4.3　Stork 系统网络吞吐量实测

本节主要验证模型的实际应用效果,将 PSOM,PCOM,FSOM 及 FCOM 4 种模型在 Stork 数据调度系统中进行实际网络条件下的实验测试。图 2 给出上述

4 种模型算法在 Stork 2.0.1 系统中的数据载入过程。

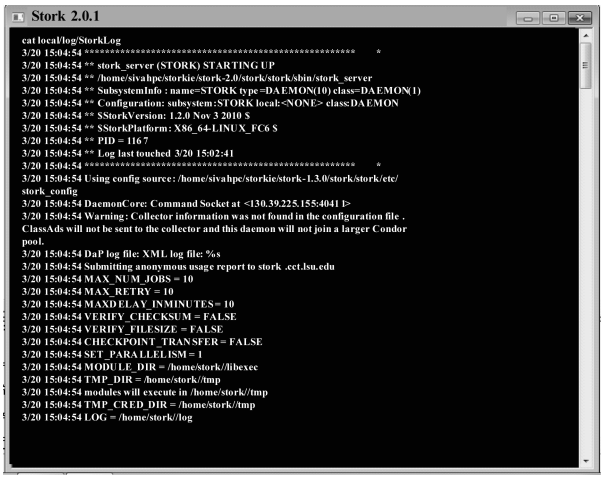

图 2 Stork 2.0.1 系统数据载入界面

表 2 给出上述模型在 Stork 数据调度系统中的实验对比结果,该结果为在 LONI, XSEDE, DIDCLab, CRON 及 EMULAB 5 种网络条件下,网络数据吞吐量的情况对比。

表 2 ~ 表 4 给出 4 种模型在 Stork 数据调度系统中有关测试平台上的网络吞吐量仿真结果,表中数据源是指数据发送端,目标为数据接收端,往返时间为数据从发送端到接收端的往返时间,网络吞吐量的单位是 Mb/s。从表 2 对比数据可看出,在同等网络条件下,FCOM 模型的网络吞吐量最佳,FSOM 模型其次,PCOM 模型和 PSOM 模型的网络吞吐量最差。网络吞吐量指标直接反映出模型算法对于数据传输网络的利用效率,从对比结果看,FCOM 模型的网络资源利用率最佳。

表 2 内存到内存的网络吞吐量结果

测试平台	数据源	目标	带宽 /(Gb·s^{-1})	往返时间 /ms	PSOM 网络吞吐量 /(Gb·s^{-1})	PCOM 网络吞吐量 /(Gb·s^{-1})	FSOM 网络吞吐量 /(Gb·s^{-1})	FCOM 网络吞吐量 /(Gb·s^{-1})
LONI	Eric	Oliver	10	3.0	326.2	425.7	554.4	802.4
XSEDE	Lonestar	Steele	1	62.0	113.8	146.3	171.8	376.9
DIDCLab	didclab-ws1	didclab-ws6	1	0.3	543.6	658.0	749.8	886.1
CRON	node0	node1	10	12.0	439.0	485.1	524.9	784.3
CRON	node0	node1	10	48.0	379.2	401.2	497.3	687.3
CRON	node0	node1	10	97.0	301.3	357.9	413.8	596.2
EMULAB	node0	node1	1	12.0	68.1	71.3	76.4	114.6
EMULAB	node0	node1	1	48.0	67.3	72.7	75.3	105.7
EMULAB	node0	node1	1	97.0	69.1	73.8	77.9	99.2

表 3 磁盘到磁盘的网络吞吐量结果

测试平台	数据源	目标	带宽 /(Gb·s^{-1})	往返时间 /ms	PSOM 网络吞吐量 /(Gb·s^{-1})	PCOM 网络吞吐量 /(Gb·s^{-1})	FSOM 网络吞吐量 /(Gb·s^{-1})	FCOM 网络吞吐量 /(Gb·s^{-1})
LONI	Eric	Oliver	10	3	365.9	398.1	453.6	743.2
XSEDE	Lonestar	Steele	1	62	129.4	143.7	166.5	216.7

表 4 内存到硬盘的网络吞吐量结果

测试平台	数据源	目标	带宽 /(Gb·s^{-1})	往返时间 /ms	PSOM 网络吞吐量 /(Gb·s^{-1})	PCOM 网络吞吐量 /(Gb·s^{-1})	FSOM 网络吞吐量 /(Gb·s^{-1})	FCOM 网络吞吐量 /(Gb·s^{-1})
ANI	anl-mempt-1	nersc-diskpt-1	100	46	689.0	754.9	887.1	1 326.9

5 结束语

本文从解决网络瓶颈带宽资源利用效率角度出发,提出一种基于并行流数全级 C 阶矩模型预测的应用层吞吐量优化算法,并根据全级 C 阶矩模型对采样数据稳定性较强的特点,设计了低采样的并行流数优化结构,通过在 GridFTP 和 Stork 2 个数据传输系统中的仿真实验验证了所提算法的性能优势。下一步工作需要在实际网络环境下进一步验证算法的性能。

参考文献

[1] Gu Lin, Zeng Deze, Li Peng. Cost Minimization for Big Data Processing in Geo-distributed Data Centers[J]. IEEE Transactions on Emerging Topics in Computing, 2014, 2(3): 314-323.

[2] 金澈清,钱卫宁,周敏奇. 数据管理系统评测基准:从传统数据库到新兴大数据[J]. 计算机学报, 2015, 38(1): 18-34.

(下转第 306 页)

参考文献

[1] Chen F, Liu A X. Privacy and Integrity Preserving Multi-dimensional Range Queries for Cloud Computing[C]// Proceedings of International Federation for Information Processing on Networking. Washington D. C., USA: IEEE Press, 2014: 1-9.

[2] Hua Yu, Xiao Bin, Wang Jianping. BR-tree: A Scalable Prototype for Supporting Multiple Queries of Multidimensional Data [J]. IEEE Transactions on Computers, 2009, 58(12): 1585-1598.

[3] Shen Guobin, Zheng Changxi, Pu Wei, et al. Distributed Segment Tree: A Unified Architecture to Support Range Query and Cover Query[Z]. 2007.

[4] Desnoyers P, Ganesan D, Shenoy P. TSAR: A Two Tier Sensor Storage Architecture Using Interval Skip Graphs[C]//Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems. New York, USA: ACM Press, 2005: 39-50.

[5] Guttman A. R-trees: A Dynamic Index Structure for Spatial Searching[M]. New York, USA: ACM Press, 1984.

[6] Bloom B H. Space/Time Trade-offs in Hash Coding with Allowable Errors [J]. Communications of the ACM, 1970, 13(7): 422-426.

[7] 谢 鲲,张大方,谢高岗,等. 基于轨迹标签的无结构 P2P 副本一致性维护算法[J]. 软件学报, 2007, 18(1): 105-116.

[8] 陈 伟,何炎祥,彭文灵. 一种轻量级的拒绝服务攻击检测方法[J]. 计算机学报, 2006, 29(8): 1392-1400.

[9] Fan Li, Cao Pei, Almeida J, et al. Summary Cache: A Scalable Wide-area Web Cache Sharing Protocol [J]. IEEE/ACM Transactions on Networking, 2000, 8(3): 281-293.

[10] Mitzenmacher M. Compressed Bloom Filters [J]. IEEE/ACM Transactions on Networking, 2002, 10(5): 604-612.

[11] Qiao Yan, Li Tao, Chen Shigang. Fast Bloom Filters and Their Generalization[J]. IEEE Transactions on Parallel and Distributed Systems, 2014, 25(1): 93-103.

[12] Hua Y, Feng D, Xie T. Multi-dimensional Range Query for Data Management Using Bloom Filters[C]//Proceedings of IEEE International Conference on Cluster Computing. Washington D. C., USA: IEEE Press, 2007: 428-433.

[13] Dharmapurikar S, Krishnamurthy P, Taylor D E. Longest Prefix Matching Using Bloom Filters[C]//Proceedings of Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications. New York, USA: ACM Press, 2003: 201-212.

[14] Liu A X, Chen Fei. Collaborative Enforcement of Firewall Policies in Virtual Private Networks[J]. IEEE Transactions on Parallel & Distributed Systems, 2011, 22(5): 887-895.

编辑 顾逸斐

(上接第 300 页)

[3] 张 滨,乐嘉锦. 基于列存储的 MapReduce 并行连接算法[J]. 计算机工程, 2014, 40(8): 70-75.

[4] Sandryhaila A, Moura J M. Big Data Analysis with Signal Processing on Graphs: Representation and Processing of Massive Data Sets with Irregular Structure[J]. IEEE Signal Processing Magazine, 2014, 31(5): 80-90.

[5] Gunay C, Edgerton J R, Li Su. Database Analysis of Simulated and Recorded Electrophysiological Datasets with PANDORA's Toolbox [J]. Neuroinformatics, 2009, 7(2): 93-111.

[6] Rathinasamy S, Raju R. Sequencing and Scheduling of Nonuniform Flow Pattern in Parallel Hybrid Flow Shop[J]. International Journal of Advanced Manufacturing Technology, 2010, 49(1): 213-225.

[7] Yin Dengpan, Yildirim E, Kosar T. A Data Throughput Prediction and Optimization Service for Widely Distributed Many-task Computing [J]. IEEE Transactions on Parallel & Distributed Systems, 2011, 22(6): 899-909.

[8] Yildirim E, Yin Dengpan, Kosar T. Prediction of Optimal Parallelism Level in Wide Area Data Transfers[J]. IEEE Transactions on Parallel & Distributed Systems, 2011, 22(12): 765-779.

[9] Hacker T J, Athey B D, Noble B. The End-to-End Performance Effects of Parallel TCP Sockets on a Lossy Wide Area Network [C]//Proceedings of the 16th International Parallel and Distributed Processing Symposium. Washington D. C., USA: IEEE Press, 2002: 314-322.

[10] Kola G, Vernon M K. Target Bandwidth Sharing Using Endhost Measures [J]. Performance Evaluation, 2007, 64(9-12): 948-964.

[11] Filippidis C, Cotronis Y, Markou C. IKAROS: An HTTP-based Distributed File System, for Low Consumption & Low Specification Devices[J]. Journal of Grid Computing, 2013, 11(4): 681-698.

[12] Wu Qishi, Zhu Mengxia, Gu Yi. A Distributed Workflow Management System with Case Study of Real-life Scientific Applications on Grids [J]. Journal of Grid Computing, 2012, 10(3): 367-393.

[13] Portmess L, Tower S. Data Barns, Ambient Intelligence and Cloud Computing: The Tacit Epistemology and Linguistic Representation of Big Data [J]. Ethics and Information Technology, 2015, 17(1): 1-9.

编辑 顾逸斐