高性能服务器集群部署系统传输模型研究

薛正华[1)] 董小社[1)] 胡雷钧[2),3)] 伍卫国[1)]

[1)](西安交通大学计算机科学与技术系 西安 710049)

[2)](高效能服务器和存储技术国家重点实验室 济南 250013)

[3)](浪潮电子信息产业股份有限公司 北京 100085)

摘 要 对基于映像的集群部署系统传输模型进行了研究,提出了基于多叉树的 TFTP 传输模型以解决 TFTP 服务器性能瓶颈问题.依据集群规模增大、系统平均带宽下降的特点,提出了基于带宽受损的动态流水线模型,模型给出了节点到达率和部署系统性能的关系.通过数学解析法、数值模拟法和实际测试对模型进行了验证.为使所提模型具有较好的可扩展性,以交换机为单位对系统进行分域,各域并行工作.作者对文中所提模型与其它 3 种映像传输模型——组播、可靠组播和 BT 进行了测试比较,结果表明,组播和可靠组播的性能较优,但可靠性难以保障,不适合传输映像文件,基于带宽受损的动态流水线模型有较高的可靠性且性能优于 BT,其部署 596MB 的映像到 48 个服务器的时间仅为 17.2s.

关键词 服务器集群;部署;动态流水线

中图法分类号 TP391 **DOI 号**: 10.3724/SP.J.1016.2008.01956

Study on Transfer Model of Deployment System for High Performance Server Cluster

XUE Zheng-Hua[1)] DONG Xiao-She[1)] HU Lei-Jun[2),3)] WU Wei-Guo[1)]

[1)](*Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an* 710049)

[2)](*State Key Laboratory of High-end Server & Storage Technology, Jinan* 250013)

[3)](*Langchao Electronic Information Industry Co., Ltd, Beijing* 100085)

Abstract This paper proposes a multi-branch tree based TFTP transfer model to relieve the performance bottleneck of TFTP server. According to the fact that average bandwidth of cluster system decreases with the increase of the size of a cluster, this paper proposes bandwidth loss-based dynamic pipeline image transfer model which reveals the relation between node arrival and deployment performance. The proposed model is validated by mathematical analysis, simulation and real measurement. To improve the scalability of the proposed model, the cluster system is divided into partitions, and each partition has only one switch. All partitions concurrently work. A comparative measurement is performed among multicast, dependable multicast, BT and the proposed model. The results show that multicast and reliable multicast have better performance, however with worse reliability. The performance of the proposed model outperforms that of BT, and it just takes 17.2 seconds to deploy a 596MB system image to 48 servers.

Keywords server cluster; deployment; dynamic pipeline

收稿日期:2008-06-10;最终修改稿收到日期:2008-09-11.本课题得到国家"八六三"高技术研究发展计划项目基金(2004AA111110,2006AA01A109)、国家自然科学基金项目(60773118)资助.**薛正华**,男,1978 年生,博士研究生,主要研究方向为高性能集群计算和网格计算.E-mail: zhenghuaxue@gmail.com. **董小社**(通信作者),男,1963 年生,教授,博士生导师,主要研究领域为高性能计算、网格计算和片上系统. E-mail: xsdong@mail.xjtu.edu.cn. **胡雷钧**,男,1971 年生,高级工程师,主要研究方向为高性能计算机体系结构和系统软件. **伍卫国**,男,1963 年生,教授,博士生导师,主要研究领域为高性能计算和嵌入式系统.

1 引　言

高性能服务器集群由于其良好的可扩展性、容错性和高性价比,被广泛应用于科学计算和事务处理,它是目前最重要的计算基础设施.在最近的超级计算机世界五百强中①,基于集群架构的系统占81.2%.随着应用需求的不断增长,集群规模扩展到上千上万台[1],部署大规模服务器集群成为挑战性的工作.具体表现在如下几方面:

高效性.部署系统能够在尽量短的时间内完成大规模集群部署.

可扩展性.部署系统不会随着集群规模的增长而产生瓶颈.

平台无关性.部署系统能够部署不同操作系统和应用;能够尽可能适应不同硬件平台.

可靠性和容错性.部署结果准确可靠;系统具有错误处理能力.

现有的部署系统大多采用基于映像[2]的部署方式,通过抓取模版服务器的系统快照将其保存为映像文件,然后将映像部署到系统中的各服务器.该方式的优点在于可部署任意操作系统和应用,具有较好的软件平台无关性.

除平台无关性外,部署系统的其它几个挑战主要依赖于映像传输方式,它是部署系统中最为重要的部分.基于实测数据,本文分析了基于映像的集群部署系统文件传输特征,为了解决 TFTP 服务器性能瓶颈问题,提出了基于多叉树的 TFTP 传输模式.针对集群规模增大,系统平均带宽下降的特点,提出了带宽受损的动态流水线部署模型,通过数学解析法、数值模拟法和实测对模型进行了验证,该模型部署效率高,并具有较好的扩展性、可靠性和容错性.

2 相关工作

2.1 映像传输模式

映像传输模式主要包括 3 种:C/S(Client/Server)、组播和 P2P(Peer to Peer)模式.

C/S 实现较为简单,但当部署一个大规模集群时,Server 端会成为性能瓶颈.

组播[3-5]采用 UDP 传输映像,其性能优于 C/S 和 P2P 模式,然而,其可靠性难以保障.一些系统采用可靠组播技术,通过增加冗余码方式提高可靠性,但增加的冗余数据会对性能带来影响.尤其是当网络数据量较大时,其丢包率增多,性能下降明显.此外,其对硬件环境的依赖性较强,不同环境下文件传输性能和可靠性差别较大,平台适应性较差.

P2P 是一种较新的映像传输模式,其特点是系统中的每个节点既是映像下载者又是提供者,该方式消除了单点性能瓶颈,同时具有较好的容错性.一些研究[6-11]对 P2P 的流量特征和传输特性进行了研究和建模,但这些研究是基于 Internet 环境,将 P2P 技术用于集群环境的研究工作较少.

2.2 现有系统

一些开源项目致力于部署系统的研究和开发,较有代表性的工作包括以下几个:

Patagonia CloneSys② 采用分区模式[2]制作映像,映像被存储在映像服务器,并通过 NFS(Network File System)方式将映像导入集群系统中的节点上.NFS 方法本质上属于 C/S 模式,当集群规模较大时,映像服务器会成为性能瓶颈.

Dolly③ 采用分区模式制作映像,并采用了一种"TCP Ring"的方式传输映像,效率较高,其本质属于 P2P 模式.然而,当环中的某一节点出现故障时,其后续节点将不得不重新部署,容错性差.

SystemImager④ 采用文件模式[2]制作映像,其性能优于分区模式.它提供了 3 种映像传输方式:Rsync、组播和 Bittorrent(BT)⑤[12-14].Rsync 是一种简单高效的文件传输方式,但其采用 C/S 模式可扩展性差.组播对硬件环境有一定的要求,容错性较差且可靠性难以保障.BT 是目前广为使用的一种 P2P 文件传输协议,是 SystemImager 新提供的一种映像传输方式,然而,BT 是一种基于 Internet 环境的文件传输协议,其内在许多保障公平性的算法并不适合集群环境,到目前为止,SystemImager 并未对此作任何改进.

3 部署系统结构

我们开发了一款基于映像的集群部署系统,其系统结构如图 1 所示.部署服务器抓取模版服务器的映像,将映像文件保存在映像服务器,并开始部署集群系统,其流程如图 2 所示.

---

① Home Top500 Supercomputing Sites. http://www.top500.org/, 2008

② Patagonia Cluster Project, Partition and Disk Cloning, Multi-Boot Installation, http://www.cs.inf.ethz.ch/CoPs/patagonia, 2008

③ Dolly — A program to clone disks /partitions, http://www.cs.inf.ethz.ch/CoPsgonia/pata/dolly.html, 2008

④ Main page — SystemImage, http://wili.systemimager.org/index.php/Main_Page, 2008

⑤ BitTorrent.org, http://www.bittorrent.org/, 2008

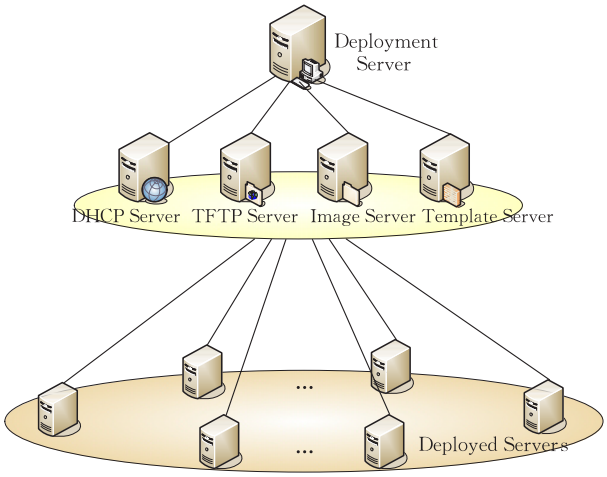

图 1 基于映像的部署系统 ESIR 结构

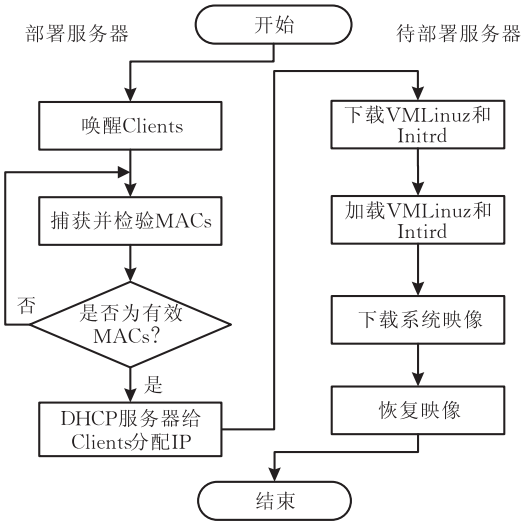

图 2 基于映像的部署流程

部署服务器通过 WOL(Wake up On LAN)远程唤醒待部署节点(Client). Client 广播 MAC 地址,DHCP 服务器捕获 MAC 地址并验证其是否是待部署节点的 MAC,以防对集群系统中其它节点进行误操作. 若是,则分配 IP 给 Client. Client 发出 TFTP 文件传输请求,TFTP 服务器响应其请求,分发 VMLinuz(一个基于 Linux 的微内核)和 Initrd(用于加载驱动和文件系统的小映像文件)文件给 Client. 获得文件后,Client 完成内核初始化工作,并请求传输系统映像. 映像传输结束后,Client 在本地磁盘上恢复映像,重新启动后完成部署过程. 值得注意的是,图 1 为系统的逻辑结构,具体实施时,可将 DHCP、TFTP 和映像服务器归并于部署服务器. 由部署流程知,整个部署过程包括两次文件传输:TFTP 文件传输和系统映像文件传输,我们分别对两次文件传输进行讨论.

4 基于多叉树的 TFTP 传输模型

TFTP 服务器传输 VMLinuz 和 Initrd 文件到 Client,其传输模式为 C/S. 当系统规模增大时,TFTP 服务器将成为系统性能瓶颈. 我们在浪潮天梭 TS10000-2 集群(具体环境详见第 6 节表 3)上测试了 TFTP 服务器网络流量和节点规模的关系,结果如图 3 所示.

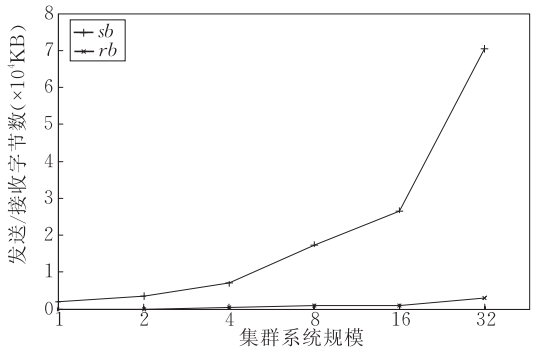

图 3 TFTP 服务器网络流量

TFTP 服务器端的 sb(发送字节数)随部署规模的增大而不断增长. 根据所测数据,采用最小二乘法[15]对所测数据进行多项式曲线拟合,分析 TFTP 网络负载增长和部署规模的关系,结果如式(1)所示.

$$y=25.4x^2+1339.7x+1355.9 \quad (1)$$

x 表示部署规模,y 为 TFTP 服务器发送字节数. 在我们的测试环境中,网络带宽上限为 128MB/s(1Gbps),令 $y=128$,代入式(1),可得 $\lfloor x \rfloor=48$,即 TFTP 服务器最多能同时支持 48 个节点.

为解决 TFTP 服务器瓶颈问题,提出了基于多叉树的 TFTP 传输模型. 其基本思想为:将 TFTP 服务文件封装在 Initrd 映像文件的相应目录中,当 Client 下载完 VMLinuz 和 Initrd 文件并完成初始化后,自动启动其上的 TFTP 服务,将其转换成一个能够提供 VMLinuz 和 Initrd 下载的 TFTP 服务器. 为描述基于多叉树的传输模型,引入如下参数:

N 表示部署系统规模;R 表示剩余的部署节点数目;K 表示一个 TFTP 服务器一次最多能部署的节点数;τ 表示一个 TFTP 服务器完成一次部署所需时间;i 表示部署次数.

基于多叉树的 TFTP 传输模型的算法描述如图 4.

根据图 4 所述算法,可计算出基于多叉树的 TFTP 传输模型的部署能力,表 1 比较了 C/S 模型与多叉树模型的部署能力. 随着时间的增长,前者的部署能力呈线性增长,而后者呈指数级增长.

```
Input: N=R, i=1,τ
Output: T
Begin:
  If (R≤(K+1)^(i-1)K){
     部署 R 个节点;
     T=τ;
    }
  Else{
       While (R>(K+1)^(i-1)K){
          部署(K+1)^(i-1)K 个 Client;
          R=R-(K+1)^(i-1)K;
          If (R>(K+1)^(i-1)K){
             TFTP 服务器将 VMLinuz 和 Initrd 文件发送给
             Client;启动(K+1)^(i-1)K 个部署后的节点的
             TFTP 服务,将其转换为 TFTP 服务器;
            }
          i=i+1;
       }
  If (R! =0){
     部署 R 个节点;
     i=i+1;
    }
  T=iτ;
  }
End
```

图 4　基于多叉树的 TFTP 传输算法

表 1　部署能力比较

时间	C/S 模型	基于多叉树模型
τ	K	K
3τ	$3K$	$(K+1)^2-1$
5τ	$5K$	$(K+1)^3-1$
…	…	…
$(2n-1)\tau$	$(2n-1)K$	$(K+1)^n-1$

5　系统映像传输模型

为了便于表述,本节称映像传输为部署,部署节点(包括 DHCP、TFTP 和 Image 服务器)为 Seed,待部署节点为 Peer. 定义如下参数:

b_p表示 Peer 的上传/下载带宽;

b_s表示 Seed 的上传/下载带宽;

$d_i(t)$表示 Peer i 在 t 时刻的下载速率;

N 表示 Peer 数目,即部署系统规模;

M 表示文件片段数目;

s 表示每个片段的大小;

T 表示部署整个系统所需时间.

集群系统中节点具有对称带宽,即上传带宽等于下载带宽,用 b_p 表示. 设 Peer 为同构节点,采用 P2P 模型部署整个集群系统的时间为

$$T=\frac{M\cdot s}{\min\{E(d_1),\cdots,E(d_i),\cdots,E(d_L)\}} \tag{2}$$

$E(d_i)$为 Peer i 在整个部署过程中的下载速率期望值. 部署系统的重要目标之一是最小化集群系统部署时间. 由式(2)知,系统部署时间取决于系统中具有最小平均下载速率的节点的部署时间,最大化该 Peer 的平均下载速率将得到最优的系统部署时间. 理想状态下,系统的最小部署时间为

$$T_{\min}=\frac{M\cdot s}{\min(b_p,b_s)} \tag{3}$$

式(3)表明,最小部署时间与 Peer 规模无关. 当 $b_p\leq b_s$时,系统中所有 Peer 在整个部署过程中都以带宽速率下载,可得到最小部署时间;当 $b_p>b_s$时,最小部署时间取决于 Seed 带宽. 然而,在实际系统中,受硬件设备能力及 P2P 部署机制约束,最小部署时间与 Peer 规模是相关的,Peer 在部署过程中难以达到带宽速率. 基于实际测试和理论分析,我们分别讨论两种制约因素.

5.1　带宽受损模型

Chariot 是一款网络性能测试工具,可对网络设备进行强度测试. 我们利用其测试了部署规模对带宽的影响. 测试环境为表 3 所示的浪潮天梭 TS10000-2 集群,分别测试了不同规模下点对点 TCP 传输的带宽能力,结果如图 5 所示.

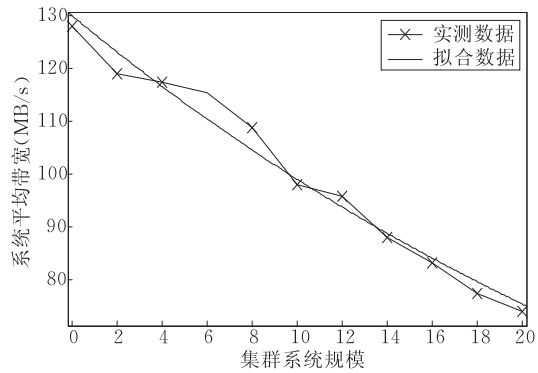

图 5　集群系统规模和系统平均带宽关系

随着系统规模的不断增大,集群系统中的数据发送和接收量增大,受网络设备处理能力限制,Peer 所能达到的最大带宽呈衰减趋势. 采用 trust-region[16] 算法对所测数据进行曲线拟合,得到 Peer 数目和系统平均带宽的关系式:

$$f(x)=130\mathrm{e}^{-0.027x} \tag{4}$$

x 表示 Peer 数目. 两者呈负指数关系.

规定 Peer 下载完映像后离开系统,设 $x(t)$为 t 时刻系统中的 Peer 数,则

$$\int_0^T x(t)f(x(t))\mathrm{d}t=NMs \tag{5}$$

为了得到最优的部署性能,等价于求解最小的集群部署时间 T. 通过上述分析,可对该问题建立如下非

线性规划数学模型，我们称该模型为带宽受损模型.

$$\begin{aligned} &\min \quad T \\ &\text{s.t.} \quad \int_0^T x(t)f(x(t))\mathrm{d}t = NMs \end{aligned} \tag{6}$$

定理 1. T 由 Peer 到达率(到达分布)决定.

证明. 定理 1 所述 Peer 到达率是指 Peer 加载完 VMLinuz 和 Initrd 文件后，下载系统映像的开始时刻所服从的分布.

设 Peer 在[0，T]时间的到达率服从随机分布 $r(t)$，离开率服从 $l(t)$，则

$$x(t) = \int_0^t (r(t) - l(t))\mathrm{d}t \tag{7}$$

设第一个/组到达的 Peer 在 t_1 时刻离开，则

$$\int_0^{t_1} f(x(t))\mathrm{d}t = Ms \tag{8}$$

$$x(t) = \int_0^t r(t)\mathrm{d}t \tag{9}$$

故第一个/组 Peer 的离开时刻 t_1 由到达率决定. 第二个/组 Peer 的离开时刻 t_2 取决于离开前 Peer 的达到率和第一个/组 Peer 的离开时刻，而第一个/组离开时刻由 Peer 到达率决定，故第二个/组 Peer 的离开时刻取决于其离开前 Peer 到达率，第三个/组 Peer 的离开率由其离开前的 Peer 到达率、t_1 和 t_2 共同决定. 由前述论证可知，t_1 和 t_2 由 Peer 到达率决定，故第三个/组 Peer 的离开率由其离开前的 Peer 到达率决定. 以此类推，第 N 个 Peer 离开时刻，即集群部署时间 T，由 Peer 到达率决定. 故式(6)所述模型实质为求解最优的 Peer 到达率.

由于 NMs 为常数，故式(6)等价于求解 $\max E(xf(x))$，即最大化系统总带宽的期望值可得到最小部署时间. 引入如下启发式规则.

规则 1. 为得到最好的部署性能，总是使系统维护 p 个 Peer 同时下载，其中，$pf(p) = \max xf(x)$.

图 6 为由式(4)和(5)得到的系统规模同系统总带宽关系图.

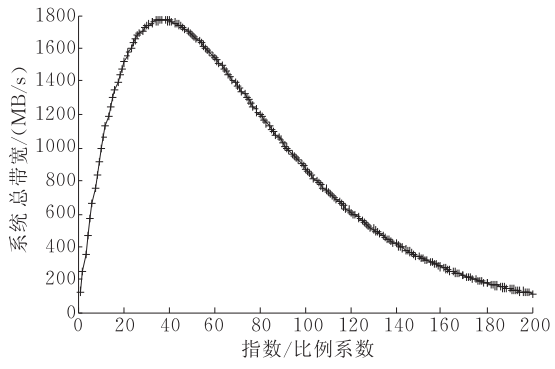

图 6 系统规模和系统总带宽关系

由 $\dfrac{\mathrm{d}xf(x)}{\mathrm{d}x} = 0$，可得 $x = 37$，即 $p = 37$ 时，系统带宽最大. 由图 6 可得式(6)的解析解满足如下结论：

(1) 当 $N \leq p$ 时，令所有 Peer 同时到达，$\max E(xf(x)) = Nf(N)$，最小部署时间 $T = \dfrac{NMs}{Nf(N)} = \dfrac{Ms}{f(N)}$.

(2) 当 $N = mp$(m 为正整数)时，分组到达，每次到达 p 个 Peer，完成部署后，下一组到达，直到 m 组完成部署. 可得到 $\max E(xf(x)) = pf(p)$，系统总带宽总是处于最大值. 故最小部署时间 $T = \dfrac{NMs}{pf(p)}$.

(3) 当 $N = mp + r(0 < r < p)$时，若允许 Peer 未下载完映像暂时离开系统，适当时刻再进入系统，则通过控制 Peer 的到达和离开使得系统在任何时刻都有 p 个 Peer 在传输映像，则可得到最小部署时间 $T_0 = \dfrac{NMs}{pf(p)}$. 然而这一控制机制的实现较为复杂. 为简化求解，以组到达方式求式(6)的近似最优解，即组大小为 p(最后一组除外)，下一组在前一组获得映像后，才能开始下载映像. 则

$$T = \frac{mpMs}{pf(p)} + \frac{rMs}{rf(r)} \tag{10}$$

其与最优解的误差为

$$\delta = T - T_0 = \frac{Ms}{f(r)} - \frac{rMs}{pf(p)} \tag{11}$$

采用数值模拟方法验证模型最优解，分别求解节点按照指数方式到达($y = \mathrm{e}^{ax}$)、线性方式到达($y = bx$)和组方式到达下的最优解. a 和 b 分别为指数系数和比例系数，映像大小为 1GB，Peer 数为 100. 结果如图 7 和图 8 所示. 结果表明，当 Peer 达到率服从组分布且每组 Peer 接近 p 时，得到的部署时间最短，此结果验证了上述关于带宽受损模型解析解结论的正确性.

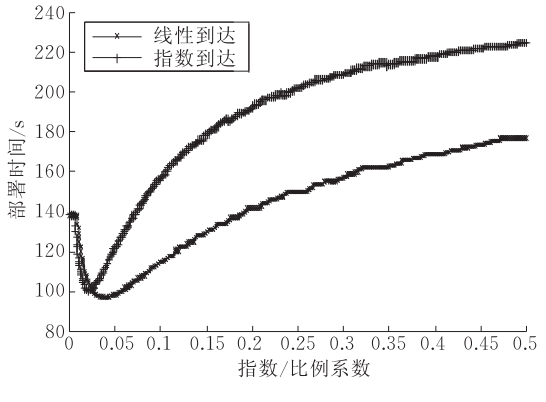

图 7 线性到达和指数到达的部署时间

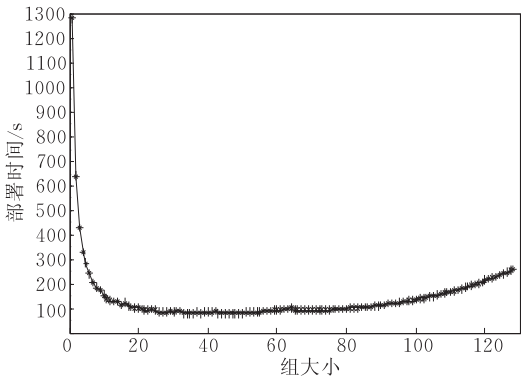

图 8　组到达的部署时间

表 2 比较了不同系统规模下,采用数学分析法得到的最优解(即 T_0)、组到达方式(组大小为 p)得到的近似最优解以及采用数值模拟法得到的最优解.由结果知,采用解析法得到的近似最优解非常接近最优解,表明结论(3)中的组到达方式可以作为一种高效的集群部署方式.采用数值模拟法得到的最优解与解析法的最优解也较接近.表 2 的末行表示模拟法取得最优解时的组大小,当系统规模为 32(小于 p),所有 Peer 同时部署可得到最短部署时间,这一结果验证了结论(1).当系统规模大于 p 时,其取得最优解的组大小在 p 附近,这一结果表明结论(3)的组到达方式可以近似为式(6)的最优解.

表 2　模型的分析解和数值模拟解结果比较

系统规模	最优分析解	近似最优分析解	最优仿真解	最优仿真解的组大小
32	18.5	18.5	21.0	32
64	37.0	37.7	40.0	37
128	74.0	76.7	81.0	37

实际系统中,Peer 数目和系统平均带宽的关系式 $f(x)$会因系统硬件环境的不同而有所不同.但带宽受损模型及模型的数学分析解法具有一般性,可应用于不同集群环境的系统部署.

5.2　基于带宽受损的动态流水线模型

5.1 节分析了设备能力对部署性能的影响,本节讨论 P2P 传输机制对文件传输性能的影响.

大部分基于 P2P 模型的文件传输协议采用消息通知机制更新 Peer 拥有的数据片断信息,即每个 Peer 每下载一个 Piece 都要向其它 Peer 通知,通过通知机制 Peer 可以即时获悉其它各 Peer 拥有的 Piece 状况.较为流行的 P2P 协议 BT 由于发送通知消息而引入的额外开销,在 Piece 大小恒定的情况下,开销与映像大小成正比,与 Peer 数目成平方比,在具有高带宽特征的集群环境中,这种通知消息为系统带来了较高的额外开销.

基于 P2P 模型的传输协议对部署性能的影响还包括其采用的随机 Peer 选择机制,该机制会导致 Peer“饥饿”现象.

定义 1.　Peer“饥饿”现象是指 Peer 由于被其它 Peer 阻塞或其它 Peer 无其需要的数据片断,而导致该 Peer 下载速率不能达到系统平均带宽的现象.

定义 2.　饥饿因子 $\eta_i(t)$表示 t 时刻 Peer i 被“怠慢”的程度,$\eta_i(t)=1-\dfrac{d_i(t)}{B(t)}(d_i(t)\leq B(t))$,$B(t)$为 t 时刻系统平均带宽.

设一个通知消息大小为 b 个字节,则

$$\int_0^t B(t)(1-\eta(t))\mathrm{d}t=Ms+(N-1)Mb \qquad (12)$$

其中,$(N-1)Mb$ 为通知机制引入的额外开销.

由式(12)知,最小化额外开销和饥饿率,可得到较优的部署时间.以此为目的,结合带宽受损模型,我们提出了基于带宽受损的动态流水线传输模型.其工作流程如下:

1. 映像分片.对映像数据进行分片(Piece),每个片指定一个片号.

2. Peer 排序.根据 Peer 的 IP 信息对其进行排序,构成一个链表,并将链表保存在 Seed.

3. 建立连接.Seed 根据带宽受损模型得到的最优到达率控制 Peer 进入部署系统(即控制 Peer 下载系统映像的时刻),各 Peer 根据排序链表信息建立 TCP 连接,形成文件传输流水线.

4. 推送 Piece.Seed 按照 Piece 序号依次推送 Piece 至 Peer 流水线.

5. 更新流水源.若第一组到达的 Peer 完成下载,该组 Peer 中,位于链表上的最后一个 Peer 比较其带宽和 Seed 带宽大小,若其带宽大于 Seed 带宽,转入步 5.1;否则,转入步 5.2.

5.1. 该 Peer 取代 Seed 作为新的流水源,该组 Peer 中的其余 Peer 离开流水线开始恢复映像.

5.2. 该组 Peer 离开流水线,新一组 Peer 与 Seed 建立连接,Seed 推送 Piece 给新一组 Peer.

6. 稳态传输.若某组 Peer 完成下载,即离开流水线.其后继 Peer 与流水源建立连接,流水源推送 Piece 给该组 Peer.当第 N 个 Peer 完成下载后,部署结束.

步 5.1 使得由于 Seed 的带宽限制而导致系统部署时间延缓的损失降为最小.动态流水线模型采用推送方式传输数据,额外开销项 $(N-1)Mb$ 为零.Peer 的下载速率取决于其前序 Peer 上传速率,而集群系统中 Peer 具有对称带宽,故 Peer 能够得到与前序 Peer 相同的下载速率,系统中的各 Peer 不会出现饥饿现象,该模型中系统各 Peer 下载速率逼近系统平均带宽.

上述部署流程中,各 Peer 加入和离开系统(即开始下载系统映像和完成下载)均通知 Seed,Seed 动态更新链表信息,其余 Peer 从 Seed 获得动态

Peer 链信息. 该模型的缺点在于流水线中的任一 Peer 出现故障,整个流水线将从故障 Peer 处断流. 基于此,我们提出了动态流水线的容错机制,其工作流程如下:

1. 失效探测. 若 Peer i 在一段时间内未收到其前序 Peer 的数据,将重连前序 Peer,若重连成功,请求从断点 Piece 处续传数据;否则,通知 Seed 隔离失效节点,转入步 2.

2. 故障点隔离. Peer i 请求 Seed 从链表信息中查询故障 Peer 的前序 Peer,并与前序 Peer 建立连接. 若连接失败,认定该 Peer 也为故障 Peer,重复步 2;否则,转入步 3.

3. 断点续传. Peer i 请求从断点 Piece 处续传数据,前序 Peer 响应请求,推送数据.

4. 失效点恢复. 失效 Peer 经修复后,若其还保存有已下载的数据片段,可从断点 Piece 处续传数据.

5.3 模型的可扩展性方案

产生带宽受损现象的主要原因是由于交换机的处理能力有限,不能同时使所有端口的流量达到峰值,且随着转发数据包的端口数的增多,交换机整体的数据包转发能力反而下降得更为明显,一个极端例子就是网络拥塞现象. 规则 1 所描述的最优值 p 即为使交换机达到最优性能的工作端口数.

为了使本文所提模型具有较好的可扩展性,我们采用分域方案,其基本思想是,以交换机为单位划分域,使得各域相对独立地并行工作,并根据带宽受损模型的结论控制 Peer 到达,使得各域内的交换机性能在系统部署过程中始终达到或接近最优. 为便于表述,设各域内节点数目相同且等于交换机的端口数,该方案的具体实施步骤如下:

1. 将集群系统以交换机为单位划分域,每个域有且仅有一台交换机.

2. 对映像分片,同 5.2 节中的步 1.

3. 从每个域内各选取 p 个 Peer. 将这些 Peer 按照所在域的先后顺序排序,形成一条链表,表头为 Seed. 重复上述步骤,直到各域内剩余 Peer 数目小于 p,将这些剩余 Peer 按照所在域的先后顺序排序,形成最后一条链表. 此时,共生成多条链表,将链表信息保存在 Seed. Peer 根据链表信息建立 TCP 连接,形成多条文件传输流水线.

4. Seed 按照 Piece 序号依次推送 Piece 至第一条流水线,直至推送完所有 Piece,结束后,比较 Seed 带宽和第一个域内的 Peer 带宽,若前者小于后者,则任选一个该域内的已完成部署的 Peer 替换 Seed 作为新的流水源. 新的流水源按照 Piece 序号依次推送 Piece 至其它几条流水线上,系统部署结束.

上述方案的主要特点在于,以交换机为单位划分域,各域内除了首或尾 Peer 与其它域相连外,其余 Peer 都是域内 Peer,因此,各域能够相对独立地并行工作,部署一个域和多个域所需的时间也几乎是一致的,这使得本文的模型具备较好的可扩展性. 此外,流水线上每个域的 Peer 数为 p,这能保证各域均以较优性能传输映像. 流水线上的各域的可靠性机制与 5.2 节中的可靠性机制类似,不予赘述.

6 测 试

在国家高性能计算中心(西安)的浪潮天梭 TS10000-1 和 TS10000-2 集群上,我们对组播、可靠组播、BT 及动态流水线 4 种传输模型进行了测试比较. 具体测试环境如表 3. 我们采用文件模式的方式制作了一个安装有 Fedora Core 3 的系统映像,其大小为 596MB. 本节所有测试均采用此映像.

表 3 测试环境

Cluster	CPU	Memory	Disk	Network
TS10000-1	2-way Intel® Xeon™ 2.80GHz	1GB	32GB SCSI disk	Gigabit Ether Net
TS10000-2	2-way Intel® Xeon™ 2.33GHz, 4 cores	4GB	150GB SAS disk	Gigabit Ether Net

6.1 可靠性测试

我们定义如下两个参数来衡量可靠性.

定义 3. 丢包率. 从发送端发出的包中,没有能够到达接收端的包所占的比率.

定义 4. 失败率. N 次映像传输过程中,发生丢包的次数所占的比率.

组播采用 UDP 传输,其性能较优,但可靠性难以保障. 我们利用组播工具 UDPCast① 在 TS10000-1 集群上对组播的丢包率进行了 10 次测试. 结果如图 9.

尽管最大丢包率不超过 0.1%,但考虑到映像的不完整性会为系统带来潜在的隐患,组播不适用于集群部署.

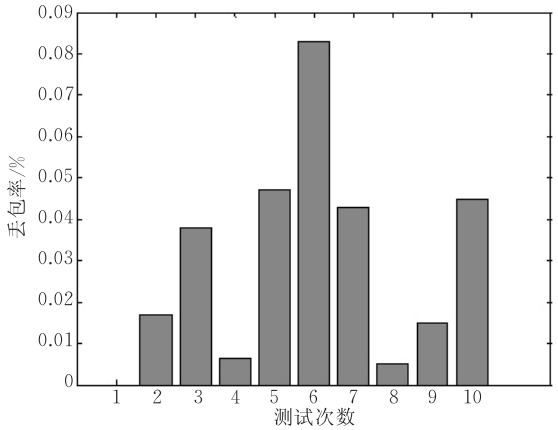

图 9 组播丢包率

① UDP Cast, http://udpcast.linux.lu/, 2008

可靠组播技术用于改善组播的可靠性,FEC(Forward Error Correction)是应用较为广泛的可靠组播技术之一,我们利用带 FEC 的 UDPCast 工具对失败率进行了测试.除了在同一集群内对失败率进行测试外,我们将 TS10000-1 中的部分服务器放置到 TS10000-2 中,以测试发送端和接收端性能不一致时的可靠组播失败率,结果如表 4.

表 4　组播可靠率测试结果

	可靠率/%		
	0	8	16
1-1	70	10	6
1-2	2	0	0
2-1	80	16	8
2-2	2	0	0

表 4 的第二行表示 FEC 纠错码长,首列表示发送端和接收端的机型.如 1-2 表示发送端为 TS10000-1 中的服务器,接收端为 TS10000-2 中的服务器.我们对每一组进行了 50 次测试.结果表明,失败率随着纠错码长的增加而减小,这是因为纠错码越长,其恢复的失败的能力越强,但其传输性能会越弱(如图 10).测试结果还表明,失败率与发送端和接收端的能力有关,接收端能力越强失败率越低.这一结论表明,可靠组播对硬件平台的适应性较差,不同的平台需要设计不同的参数.而其它两个传输模型由于采用 TCP 传输,其部署结果是可靠的,对硬件平台没有特殊要求.

6.2　性能测试

在浪潮天梭 TS10000-2 集群上,我们比较了 4 种传输模型的性能,其结果如图 10 所示,其中,FEC-0 表示组播.

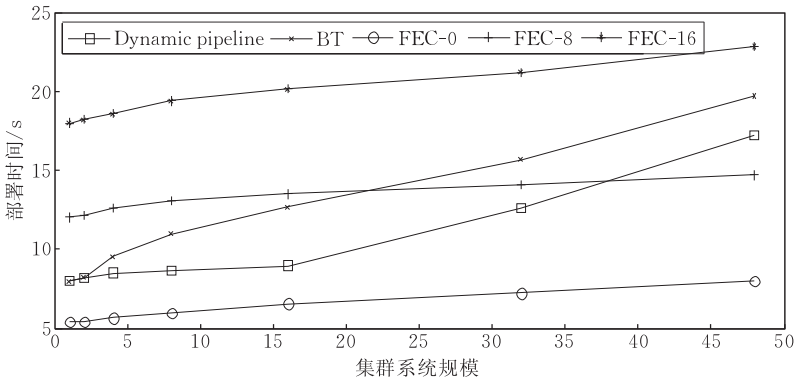

图 10　传输模型性能比较

组播性能最优,但可靠性最差.当集群系统规模较小时,基于带宽受损的动态流水线模型的性能优于 BT 和可靠组播(FEC-8 和 FEC-16).随着集群规模的增大,可靠组播的性能逐渐优于动态流水线模型和 BT.这是由于,随着集群规模增大,基于 TCP 协议的集群部署系统其传输总开销高于基于 UDP 协议的传输,故组播和可靠组播的性能较优.然而,当集群规模增大到一定程度后,系统网络流量较大,网络传输的可靠性变差,可靠组播的丢包率和失败率增长,恢复丢包数据所产生的时延会增加系统的部署时间.此外,由于节点无法恢复数据(所发送的一组数据中丢包数大于纠错码长时,导致无法恢复)而不得不重新部署节点,使得整个系统部署时间明显增长.文献[17]比较了组播和 BT 的传输性能,给出了部署某映像时的测试数据,结果表明当集群规模大于 100 时,采用组播部署的失败率异常高,其部署时间呈线性增长,且增长率远高于 BT,部署性能明显劣于 BT.而图 11 的结果表明,基于带宽受损的动态流水线模型的部署性能始终优于 BT.

从图 10 的测试结果可知,动态流水线模型部署 596MB 映像到 32 个节点时间为 11.8s,系统的平均下载率为 50.5MB/s,而第 5 节表 2 中,部署 1GB 映像到 32 个节点的最优部署时间为 18.2s,系统的平均下载率为 56.2MB/s.实测值与理论分析值较接近.这是由于本文的理论分析是建立在实测数据基础上,能较为准确地反映实际部署结果,具有一定的实践指导意义.测试结果中,部署 48 个节点的时间为 17.2s,平均下载率为 34.7MB/s.根据式(2),部署 48 个节点的系统平均带宽为 35.6MB/s,基于带宽受损的动态流水线模型的部署速率接近系统平均带宽.这一结果表明该模型是高效的,其引入的额外开销几乎可以忽略.

7　结　论

本文研究了基于映像的集群部署系统传输模型.为解决 TFTP 服务器性能瓶颈问题,提出了基于多叉树的 TFTP 传输模型.针对集群规模增大、

系统平均带宽下降的现象,提出了带宽受损映像传输模型,对模型的最优解进行了分析,并通过数学解析法、数值模拟法和实际测试对模型进行了验证.针对部署大规模集群系统对性能、可扩展性、可靠性和容错性的要求,提出了基于带宽受损的动态流水线模型,该模型是一种特殊的 P2P 传输模型,其映像传输速率逼近系统平均带宽,并具有较好的可扩展性、可靠性和容错性.对本文所提模型与几种传输模型——组播、可靠组播和 BT 进行了测试比较,结果表明,组播和可靠组播的性能较优,但可靠性难以保障,不适合传输映像文件,基于带宽受损的动态流水线模型的性能优于 BT.

本文的传输模型是基于映像的系统传输模型,将操作系统和安装配置好的应用捆绑为映像进行传输,在未来工作中,我们将进一步研究操作系统和应用分离的部署技术,实现独立于操作系统的应用的自动部署、安装和配置.

参 考 文 献

[1] Davis K, Hoisie Z, Johnson G et al. A performance and scalability analysis of the BlueGene//Proceedings of the IEEE/ACM SC2004 Conference — Bridging Communities. Pittsburgh, PA, United States, 2004: 711-719

[2] Dailey L, Zhang J, Landau R et al. A modeling perspective of image-based installation. Dell Power Solutions, Austin, TX, USA: Dell Incorporation, 2002: 1-13

[3] Tsutzbach D, Rejaie R, Sen S. Characterizing unstructured overlay topologies in modern P2P file-sharing systems. IEEE/ACM Transactions on Networking, 2008, 16(2): 267-280

[4] Chen C, Tsai K. The server reassignment problem for load balancing in structured P2P systems. IEEE Transactions on Parallel and Distributed Systems, 2008, 19(2): 234-246

[5] Naor M, Wieder U. Novel architectures for P2P applications: The continuous-discrete approach. ACM Transactions on Algorithms, 2007, 3(3): 1273350

[6] Hu Y, Zhu Y. Efficient proximity-aware load balancing for DHT-Based P2P systems. IEEE Transactions on Parallel and Distributed Systems, 2005, 16(4): 349-361

[7] Sen S, Wang J. Analyzing peer-to-peer traffic across large networks. IEEE/ACM Transactions on Networking, 2004, 12(2): 219-232

[8] Wu C, Li B, Zhao S. Characterizing peer-to-peer streaming flows. IEEE Journal on Selected Areas in Communications, 2007, 25(9): 1612-1626

[9] Papadopoulos C, Parulkar G, Varghese G. Light-weight multicast services (LMS): A router-assisted scheme for reliable multicast. IEEE/ACM Transactions on Networking, 2004, 12(3): 456-468

[10] Gau R, Haas Z, Krishnamachari B. On multicast flow control for heterogeneous receivers. IEEE/ACM Transactions on Networking, 2002, 10(1): 86-101

[11] Defago X, Schiper A, Urban P. Total order broadcast and multicast algorithms: Taxonomy and survey. ACM Computing Surveys, 2004, 36(4): 372-421

[12] Yue Y, Lin C, Tan Z. Analyzing the performance and fairness of BitTorrent-like networks using a general fluid model. Journal of Commputer Communication, 2006, 29(18): 3946-3956

[13] Qiu D, Srikant R. Modeling and performance analysis of BitTorrent-like peer-to-peer networks. ACM SIGCOMM Computer Communication Review, 2004, 34(4): 367-378

[14] Liogkas N, Nelson R, Kohler E, Zhang L. Exploring the robustness of BitTorrent peer-to-peer content distribution systems: Research Articles. Concurrency and Computation: Practice & Experience, 2008, 20(2): 179-189

[15] Markovsky I, Huffel S. Overview of total least-squares methods. Journal of Signal Processing, 2007, 87(10): 2283-2302

[16] Helfrich H, Zwick D, Trust region algorithm for parametric curve and surface fitting. Journal of Computational and Applied Mathematics, 1996, 73(1-2): 119-134

[17] Focht V, Righi A, Finley B et al. SystemImager and BitTorrent: A peer-to-peer approach to large scale OS deployment//Proceedings of the 2007 LinuxTag Conference. Berlin, Germany, 2007

XUE Zheng-Hua, born in 1978, Ph. D. candidate. His research interests include high performance cluster computing and autonomic computing.

DONG Xiao-She, born in 1963, Ph. D., professor, Ph. D. supervisor. His research interests include high performance cluster computing, grid computing, and system on chip.

HU Lei-Jun, born in 1971, senior engineer. His research interests include high performance cluster computing and grid computing.

WU Wei-Guo, born in 1963, Ph. D., professor, Ph. D. supervisor. His research interests include high performance computing and embedded system.