ISSN 1673-9418 CODEN JKYTA8
Journal of Frontiers of Computer Science and Technology
1673-9418/2018/12(04)-0559-08
doi: 10.3778/j.issn.1673-9418.1611087

E-mail: fcst@vip.163.com
http://www.ceaj.org
Tel: +86-10-89056056

Alltoall通信性能模型研究*

罗红兵+,张晓霞,魏 勇
北京应用物理与计算数学研究所 高性能计算中心,北京 100094

Research on Performance Model for Alltoall Collective Communication*

LUO Hongbing+, ZHANG Xiaoxia, WEI Yong
High Performance Computing Center, Institute of Applied Physics and Computational Mathematics, Beijing 100094, China
+ Corresponding author: E-mail: hbluo@iapcm.ac.cn

LUO Hongbing, ZHANG Xiaoxia, WEI Yong. Research on performance model for Alltoall collective communication. Journal of Frontiers of Computer Science and Technology, 2018, 12(4): 559-566.

Abstract: Alltoall is an important collective operation of MPI (message passing interface), which impacts the parallel efficiency of many parallel numerical computing applications. Since theoretic analysis and evaluation of Alltoall collective communications on the massive parallel computer is still insufficient, improper communication module design and poor communication usage are widespread in many applications. The test of basic MPI communication performance shows that the communication latency is instable due to network contention and the variety increases with the number of MPI processes. In order to reduce the gap between the predicted run-time and the measured run-time, this paper proposes a new performance model to evaluate Alltoall operation. The model considers not only the standard parameters such as bandwidth and latency, but also takes into account network communication contention. Results on the BXJ supercomputer show that the performance prediction model accurately captures the Alltoall communication behavior even for the operations on a large number of processors and manifests network competition cost on the Alltoall communication.

Key words: collective communication; communication performance; Alltoall

摘 要: Alltoall是一种重要的MPI(message passing interface)集合通信类别,是影响许多并行程序并行效率的

* The National High Technology Research and Development Program of China under Grant No. 2014AA01A302 (国家高技术研究发展计划(863计划)).
Received 2016-11, Accepted 2017-01.
CNKI网络出版: 2017-01-16, http://www.cnki.net/kcms/detail/11.5602.TP.20170116.1702.014.html

重要因素。但对于大规模并行计算机上Alltoall集合通信的评测和理论分析仍较为缺乏,导致许多应用程序的通信模块设计和使用不合理。首先,开展了MPI基本通信性能的测试和分析,发现随着MPI进程数的增加,其性能波动也增加,而这种波动源自网络竞争。为此,在传统的Alltoall性能评估模型中引入了网络竞争因素,新模型不仅考虑传统的通信带宽和通信延迟参数,还考虑了通信竞争因素。某国产并行机平台上的测试结果显示:引入网络竞争模型的新Alltoall性能评估模型可以较为准确地预估Alltoall性能,体现出网络竞争开销对Alltoall性能的影响。

关键词:集合通信;通信性能;Alltoall

文献标志码:A **中图分类号:**TP311

1 引言

MPI(message passing interface)通信性能是影响并行应用程序性能的关键,特别是MPI集合通信性能对于应用的可扩展性往往具有决定性的作用。在MPI集合通信中,Alltoall是让所有参与通讯的进程彼此进行数据交换的集合通信操作,对于采用该通信模式的应用,例如三维快速傅里叶变换[1]和量子力学分子动力学模拟CPMD(Car-Parrinello molecular dynamics)[2],Alltoall性能对应用软件性能的影响非常大。为此,Alltoall相应的评估和优化研究一直是并行计算领域的研究热点,包括对Alltoall等集合通信的详细分析[3-5],针对当前多核CPU的Alltoall的优化[6-7],在通信算法层[8]针对特定高性能计算机[9]对Alltoall进行的优化等。

已有的研究结果[5]显示:Alltoall的理论预估值与实际测试值的差别往往较大,尤其在超大规模情况下,实测值甚至是理论值的数倍,反映出对Alltoall集合通信性能的理论建模仍然是值得深入研究的问题。如何利用理论模型解释Alltoall的性能,是MPI通信算法设计、评估和优化,乃至高性能计算机优化中必须要面对的问题。当前,对于Alltoall集合通讯的性能建模[10-11]大都基于基本的通讯模型进行,其中被广泛使用的通信性能模型是LogP(latency,overhead,gap,and processor)模型[12],该模型是一个针对分布式存储的多处理器模型,处理器间采用点对点通信。LogGP模型[13]在LogP模型的基础上增加了一个参数G,该参数可以描述在传递长消息时获得的带宽。从现有的研究和实验结果看,某些因素未在模型中准确地体现,导致Alltoall性能理论预测在大规模情况下的失真。

针对超大规模情况下Alltoall的理论性能模型存在的不足,本文从MPI通信的基本特征和Alltoall实现算法和模型两方面予以分析,希望刻画出实际互连网络系统中的某些特征,以期建立更为精确的Alltoall性能模型。

2 Alltoall实现算法分析

MPI的开源实现版本mpich中对Alltoall的实现涉及4个算法,分别面向不同的消息长度和进程数规模,具体为:

(1)对于短消息(缺省是不大于256 B)且MPI进程数大于等于8,采用存储前进算法,以多传输数据来减少通信延迟的影响,算法需执行$\mathrm{lb}p$步,单进程的数据传输量增加到原传输量的$\mathrm{lb}p/2$倍。

(2)对于中等规模的消息(缺省为不大于32 KB)且MPI进程数小于8,以同时进行irevs和isends,再进行一次waitall的方式实现,其中需避免所有进程在同一时刻向同一进程进行irevs和isends。

(3)对于长消息且进程数为2的幂,使用配对交换算法,需$p-1$个传输步。

(4)对于长消息且进程数不为2的幂,以第i步,每个进程从$rank-1$收消息,向$rank+1$发消息的流程进行,需$p-1$个传输步。

对于大规模的Alltoall通信,分别在短消息时用算法(1),在其余长度的消息时使用算法(3)和算法(4)。用k表示消息块的大小,p表示进程数,α表示通信延迟,β表示通信带宽的倒数,Alltoall时间开销分别可以表示为:

$$T_{\text{bruck}} = \text{lb}(p)\times\alpha + \frac{1}{2}\times\text{lb}(p)\times(p-1)\times k\times\beta \quad (1)$$

$$T_{\text{long}} = (p-1)(\alpha + k\times\beta) \quad (2)$$

从以上算法体现的Alltoall时间开销看,Alltoall通信性能建模依赖于通信延迟和通信带宽的准确刻画。由于通信延迟和通信带宽一方面依赖于高性能计算机互连网络的实现技术,一方面依赖于系统负载情况,其性能的准确刻画并非易事。Alltoall涉及到 p 个进程同时进行通信,当 p 的数量达到一定规模时,其通信性能不可避免地有所差别,这也是在Aalltoall性能建模时需要考虑的。

3 Alltoall通信性能模型

Alltoall通信性能模型依赖于互连网络通信性能模型,考虑到大规模互连通信网络中通信性能模型的复杂性,本文首先选择一个实际系统进行评测,以期总结其性能特征。在此基础上,结合Alltoall的特点,建立一个较为合理的性能模型,然后在此基础上设计Alltoall通信性能模型。

3.1 测试平台

测试平台选择某国产并行机(简称BXJ),该系统的每个计算节点包含2颗英特尔微处理器,每颗微处理器包含6个计算核心;互连系统采用自主设计的高阶路由芯片(network route chip,NRC)和高速网络接口芯片(network interface chip,NIC),实现光电混合的二层胖树结构高阶路由网络互连。NRC采用了16×16高阶网络交换部件,计算节点最大跳转次数为3,工作主频为312.5 MHz,时钟周期为3.2 ns,基本传输单位为256 bit。BXJ并行机通信系统的性能参数详见表1,NRC路由交换芯片的基本参数详见表2。

Table 1 Basic parameters for communication system of BXJ parallel computer

表1 BXJ并行机通信系统基本参数

结构	类别	参数值
设备层	GLEX通讯点对点最小延迟/μs	1.58
	GLEX通讯单向最大带宽/(MB/s)	6 343
	GLEX通讯双向最大带宽/(MB/s)	9 710
软件层	MPI通讯延迟/μs	2.37
	MPI单向通讯带宽/(MB/s)	6 343
	MPI双向通讯带宽/(MB/s)	9 236

Table 2 Basic parameters for NRC interconnection

表2 NRC互连基本参数

类别	参数值
最大跳转次数	3
工作主频/MHz	312.5
报文大小/bit	256
交叉开关	16×16

3.2 基本通信性能分析

选用Intel IMB测试程序,测试BXJ上16至8 192个MPI进程执行Sendrecv操作的通信延迟和通信带宽情况。与Alltoall类似,测试程序中的每个MPI进程同时执行Sendrecv操作,都参与数据通信。测试含单计算节点启动8个MPI进程和12个MPI进程2组测试。表3和表4是有关通信延迟的部分测试结果,图1是测试中出现的通信延迟抖动(通信延迟的最大波动幅度与通信延迟的平均值之比)与进程数间的关系。其中的趋势线显示:通信延迟的抖动幅度随着进程数的增多明显呈增大的趋势。

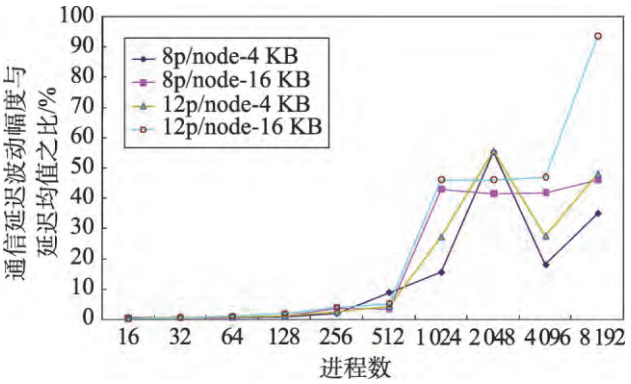

Fig.1 Relationship between latency and the number of processes

图1 通信延迟抖动与进程数间的关系

表3和表4是16~8 192进程时,通信延迟的具体结果,其中单MPI进程的消息块分别是4 KB和16 KB。表3和表4中的数据显示:无论是单计算节点启动8个MPI进程,还是单计算节点启动12个MPI进程,这种通信延迟的抖动都在一定程度上存在。计算节点启动的MPI进程较多时,抖动的幅度更大。

图2和图3是BXJ上16进程至8 192进程下的通

Table 3 Relationship between latency and the number of processes (8 processes per node)

表3 通信延迟与进程数的关系(单节点8个MPI进程)

进程数	4 KB数据块				16 KB数据块			
	最大延迟/μs	最小延迟/μs	延迟差/μs	提升比例/%	最大延迟/μs	最小延迟/μs	延迟差/μs	提升比例/%
16	18.30	18.23	0.07	0.38	40.82	40.75	0.07	0.17
32	17.99	17.92	0.07	0.39	37.39	37.24	0.15	0.40
64	18.12	18.03	0.09	0.50	37.49	37.31	0.18	0.48
128	18.09	17.98	0.11	0.61	37.71	37.31	0.40	1.07
256	18.35	18.04	0.31	1.72	39.68	38.27	1.41	3.68
512	19.62	18.05	1.57	8.70	38.72	37.49	1.23	3.28
1 024	20.69	17.93	2.76	15.39	54.01	37.80	16.21	42.88
2 048	27.84	17.92	9.92	55.35	53.39	37.72	15.67	41.54
4 096	21.06	17.86	3.20	17.92	53.32	37.63	15.69	41.69
8 192	24.24	17.96	6.28	34.97	54.87	37.56	17.31	46.09

Table 4 Relationship between latency and the number of processes (12 processes per node)

表4 通信延迟与进程数的关系(单节点12个MPI进程)

进程数	4 KB数据块				16 KB数据块			
	最大延迟/μs	最小延迟/μs	延迟差/μs	提升比例/%	最大延迟/μs	最小延迟/μs	延迟差/μs	提升比例/%
16	23.86	18.23	0.04	0.17	50.03	49.94	0.09	0.18
32	24.08	23.82	0.16	0.67	54.26	53.97	0.29	0.54
64	25.60	23.92	0.20	0.79	54.62	54.10	0.52	0.96
128	25.04	25.40	0.28	1.13	54.71	53.74	0.97	1.80
256	24.82	24.76	0.56	2.31	56.01	53.96	2.05	3.80
512	25.63	24.26	1.05	4.27	56.30	53.55	2.75	5.14
1 024	30.80	24.58	6.57	27.12	78.41	53.67	24.74	46.10
2 048	37.67	24.23	13.47	55.66	78.39	53.68	24.71	46.03
4 096	30.86	24.20	6.66	27.52	78.31	53.30	25.01	46.92
8 192	35.65	24.20	11.57	48.05	102.83	53.09	49.74	93.69

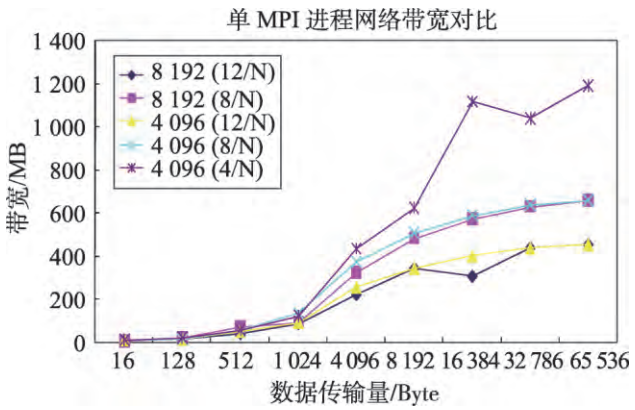

Fig.2 Relationship between communication bandwidth of single MPI process and the size of messages

图2 单进程时MPI通信带宽与数据传输量间的关系

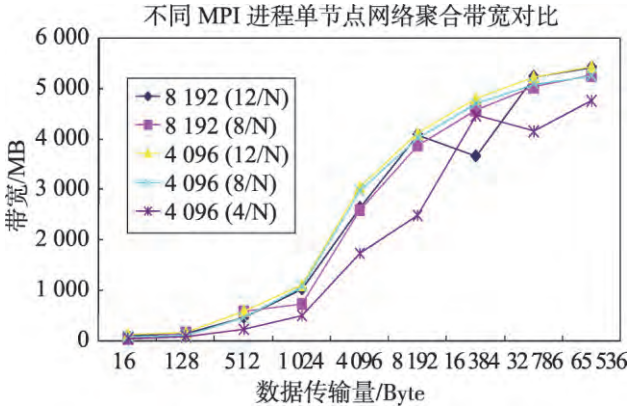

Fig.3 Relationship between cumulative communication bandwidth of single node and the size of messages

图3 单计算节点通信带宽与数据传输量间的关系

信带宽情况。图2和图3的数据显示:计算节点的通信带宽随消息块的增大而增加,直到达到最大值,不同进程数下节点的增长趋势基本一致;单节点上所有MPI进程分享通信带宽,启动的MPI进程数越多,分享的带宽越少。

基于以上对大规模并行情况下通信延迟和通信带宽情况的分析,可以得出以下基本结论:

(1)通信延迟的准确刻画并非易事,随着进程数和数据传输量的增加,网络传输会存在竞争,导致通信延迟的变化和性能抖动。

(2)对于通信带宽,利用MPI进程实测通信带宽基本可以反映其特征。

3.3 Alltoall性能模型

已有的研究[14]显示,通信性能与负载有关。评估互连网络性能需要定义负载模型,涉及目的分布、注入速率和消息长度等。

对于Alltoall集合通信而言,目的分布是均匀的,数据注入规律简单,消息长度固定,因而评估其通信延迟时可以在已有通信性能模型[15]上简化。考虑到互连网络的多样性,本文仅仅针对多级互连网络(multistage interconnection networks, MINs)进行建模,这是当前使用最为普遍的网络类别。

通常来说,实现 N 个计算节点互连的 $N\times N$ MIN互连网络由 $L=\log_k N$ 级 $k\times k$ 交换单元构成。为便于描述,假定网络完全由 $k\times k$ 交换部件构成, $k\times k$ 交换部件含 k 个输入端口和 k 个输出端口,每个输出端口在单时钟周期内分别可以接受一个报文。为防止阻塞,每个输出端口的buffer实现为FIFO(first input first output)队列。到达的报文直接进入到与目的输出端口对应的buffer,不同的buffer之间不会有冲突。

对于以上理想的交换单元,令其时钟周期为 t_c , t_T 为从交换单元到下一交换单元的传输时间。假定在每个时钟周期,报文到达每个输入端口的可能性为 ρ ,令 v_n 表示在时刻 n 加入到一个输出队列的报文的数目,那么 $v_1,v_2,\cdots,v_n$ 为独立的符合伯努利分布的随机变量。到达报文数量的数学期望 $E=k\times\frac{\rho}{k}=\rho$, 其方差 $V=k\times\frac{\rho}{k}\times\left(1-\frac{\rho}{k}\right)=\rho\left(1-\frac{\rho}{k}\right)$。令 q_n 为时刻 n 在队列中的报文数目, q_n 和 v_n 有如下关系式:

$$q_{n+1}=q_n+v_{n+1}-1,\ \text{if}\ q_n>0$$

$$q_{n+1}=v_n+1,\qquad \text{if}\ q_n=0$$

上面排队关系可以用M/G/1队列系统描述[16-17],相应地,到达输出端口报文数的数学期望为:

$$\bar{q}=\frac{E}{2}+\frac{V}{2(1-E)}$$

报文通过交换部件的时间的数学期望为:

$$\bar{s}=\frac{\bar{q}}{E}=\frac{1}{2}+\frac{V}{2E(1-E)}$$

报文通过交换部件的等待时间的数学期望为:

$$\bar{w}=\bar{s}-1=\frac{V}{2E(1-E)}-\frac{1}{2}$$

求出 E 和 V 代入上式,可以得到:

$$\bar{w}=\frac{(1-1/k)p}{2(1-p)} \tag{3}$$

将式(3)引入到式(2),可以得到增加了网络竞争因素的Alltoall性能模型:

$$T_{\text{contention}}=(p-1)(\alpha+k\times\beta+\frac{k}{k_p}\times n_{\text{hop}}\times\bar{w}) \tag{4}$$

其中, k_p 是网络最小传输单位(报文)的大小; n_{hop} 是报文需要经过的交换单元数目。

4 模型验证和评估

由于BXJ测试平台处于生产性运行状态,实际测试时没有机会占用全系统,以下相关测试的最大并行规模为8 192个MPI进程,Alltoall的实测采用Intel IMB测试程序获得。

4.1 传统Alltoall性能模型评估

首先,评估实测值与采用传统模型时理论估值的对比情况,图4和图5是对比结果图。其中,理论值按照实际实现算法估算,在128 B短消息时使用Bruck算法,在16 KB消息时下使用Long算法,涉及的通信延迟 α 和通信带宽 β 分别使用系统标称的理论值和实测值。图中,Alltoall的实测值用Real标注,另外标注中的8和12表示单节点启动的MPI进程数;理论值用“算法名 + 数字 + 字母”标注,例如:Long8B表示理论值按Long算法估算,单节点启动8个MPI进程,字母B表示通信延迟 α 和通信带宽 β 采用理论值;Bruck12A表示理论值按Bruck算法估算,通信延迟 α 和通信带宽 β 采用实测值。

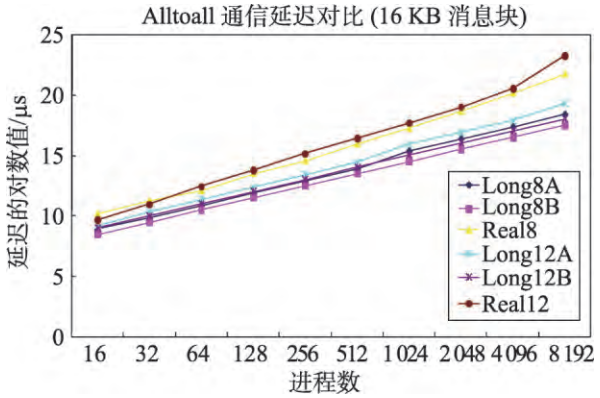

Fig.4 Comparison of actual value and predicted value by different Alltoall models on BXJ (16 KB message)

图4 BXJ上Alltoall传统模型估值与实测值对比(16 KB消息)

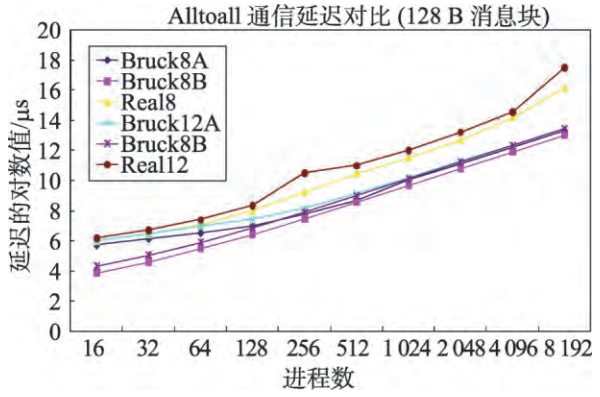

Fig.5 Comparison of actual value and predicted value by different Alltoall models on BXJ (128 B message)

图5 BXJ上Alltoall传统模型估值与实测值对比(128 B消息)

考虑到同一消息块不同进程数下Alltoall的实测值与传统模型理论估值的差别太大,为方便比较,图4和图5中延迟值是实际数的对数值(2为幂)。图4和图5中的结果显示:(1)相比使用通信延迟和通信带宽的理论值,以实测值为参数,Alltoall理论值更接近于实测值;(2)即便以实测值为参数,Alltoall理论值的准确性有所提高,但仅在MPI进程数小于128时有效,超过128进程后实测值基本上是理论值的数倍,显示出传统的Alltoall模型在大规模并行时对于Alltoall的性能评估存在明显缺陷。

4.2 新Alltoall性能模型评估

表5和表6分别是4 KB消息块和16 KB消息块时Alltoall实测性能(延迟值)与理论预估的对比,分为单计算节点启动8个MPI进程和12个MPI进程两组,MPI并行规模从512进程至最大8 192进程。表中"原模型"为利用式(2)的估算结果,"新模型"为式(4)的估算结果。

Table 5 Comparison of actual value and predicted value by different Alltoall models (4 KB message)

表5 Alltoall实测性能与理论值对比(4 KB消息块)

进程数	单节点8个MPI进程延迟/μs			单节点12个MPI进程延迟/μs		
	实测值	原模型	新模型	实测值	原模型	新模型
512	17 411	11 941	17 435	24 610	17 911	23 406
1 024	45 186	23 905	43 393	62 408	35 858	65 090
2 048	3 230 697	47 834	117 007	160 906	71 752	158 666
4 096	330 610	95 693	346 391	435 567	143 540	429 731
8 192	983 752	191 410	908 548	2 512 881	287 115	2 117 705

Table 6 Comparison of actual value and predicted value by different Alltoall models (16 KB message)

表6 Alltoall实测性能与理论值对比(16 KB消息块)

进程数	单节点8个MPI进程延迟/μs			单节点12个MPI进程延迟/μs		
	实测值	原模型	新模型	实测值	原模型	新模型
512	62 157	19 925	57 600	87 675	29 888	86 400
1 024	152 595	39 890	137 446	226 292	59 835	206 170
2 048	401 802	79 819	419 392	520 184	119 729	534 762
4 096	1 150 017	159 677	1 162 755	1 479 123	239 516	1 384 282
8 192	3 387 261	319 394	3 187 948	9 891 631	479 091	7 801 452

在理论估算中,通信延迟 α,使用表1中的MPI通信延迟值和MPI单向通信带宽值,计算通信数据量时考虑单计算节点启动8个MPI进程和12个MPI进程对应到单个通信端口数据量的差别。在使用新模型时,依照数量传输量换算公式(3)的 p 值,其余参数选择表2中的数据。

表5和表6中的数据显示:(1)引入网络竞争后的Alltoall性能预估值与实测值非常接近,体现出网络竞争是可以预测的;(2)从数值上看,影响大规模Alltoall性能的主要因素是网络竞争开销,而网络的基本传输延迟和传输带宽的占比很小;(3)Alltoall性能实测时有时会有很大的波动,如表5中2 048个进

程(单节点启动8个MPI进程)时Alltoall实测值存在明显的跳跃,这种现象是由于突发的网络拥塞造成的。

5 小结

综合以上测试和分析,不难看出:

(1)MPI通信性能对于底层互连通信系统性能的依赖性很强,并且与负载有关。尤其是对于Alltoall这种让所有参与通讯的进程进行彼此数据交换的集合通信操作,其性能对于底层互连通信系统的要求最高,最难实现非常好的可扩展性。

(2)预估Alltoall通信的理论值时,需要考虑网络竞争的影响,否则,无论是采用MPI的通信延迟和通信带宽的理论,还是采用实测值,都不一定能够反映出Alltoall的真实特性,尤其是面对大规模Alltoall操作。

(3)在大规模并行时,主导Alltoall性能的主要因素是网络竞争开销,而不是网络的基本传输延迟和传输带宽。

References:

[1] Luszczek P, Dongarra J, KoesterD, et al. Introduction to the HPC challenge benchmark suite[R]. Springfield: Lawrence Berkeley National Laboratory, 2005.

[2] The CPMD Consortium. CPMD: Car-Parrinello molecular dynamics, Version3.15.3[EB/OL]. (2015)[2016-07-30]. http://cpmd.org/downloadable-files-authentication/manual.pdf.

[3] Rao Li, Zhang Yunquan, Li Yucheng. Performance test and analysis of Alltoall collective communication on domestic hundred trillion times cluster system[J]. Computer Science, 2010, 37(8): 186-188.

[4] Liu Yang, Cao Jianwen, Li Yucheng. Testing and analyzing of collective communication models[J]. Computer Engineering and Applications, 2006, 42(9): 30-33.

[5] Luo Hongbing, Zhang Xiaoxia. Analysis of scalability for MPI collective communication[J]. Journal of Frontiers of Computer Science and Technology, 2017, 11(2): 252-261.

[6] Xu Cong, Venkata M G, Graham R L, et al. SLOAVx: scalable logarithmic AlltoallV algorithm for hierarchical multicore systems[C]//Proceedings of the 13th International Symposium on Cluster, Cloud, and Grid Computing, Delft, May 13-16, 2013. Washington: IEEE Computer Society, 2013: 369-376.

[7] Li Qiang, Sun Ninghui, Huo Zhigang, et al. Optimizing MPI Alltoall communications in multicore clusters[J]. Journal of Computer Research and Development, 2013, 50(8): 1744-1754.

[8] Bruck J, Ho C T, Kipnis S, et al. Efficient algorithms for all-to-all communications in multiport message-passing systems[J]. IEEE Transactions on Parallel and Distributed Systems, 1997, 8(11): 1143-1156.

[9] Kumar S, Mamidala A, Heidelberger P, et al. Optimization of MPI collective operations on the IBM blue gene/Q supercomputer[J]. International Journal of High Performance Computing Applications, 2014, 28(4): 450-464.

[10] Mamadou H N, Nanri T, Murakami K, et al. Performance analysis and linear optimization modeling of all-to-all collective communication algorithms[C]//Proceedings of the 19th Symposium on Computer Architecture and High Performance Computing, Gramado, Oct 24-27, 2007. Washington: IEEE Computer Society, 2007: 203-210.

[11] Chan E, Heimlich M, Purkayastha A, et al. Collective communication: theory, practice, and experience[J]. Concurrency and Computation: Practice and Experience, 2007, 19(13): 1749-1783.

[12] Culler D E, Karp R M, Patterson D, et al. LogP: a practical model of parallel computation[J]. Communications of the ACM, 1996, 39(11): 78-85.

[13] Alexanddrov A, Ionescu M F, Schauser K E, et al. LogGP: incorporating long messages into the LogP model-one step closer towards a realistic model for parallel computation [C]//Proceedings of the 7th Annual ACM Symposium on Parallel Algorithms and Architectures, Santa Barbara, Jul 17-19, 1995. New York: ACM, 1995: 95-105.

[14] Duato J, Yalamanchili S, Ni L. Interconnection network: an engineering approach[M]. Xie Lunguo, Zhang Minxuan, Dou Qiang, et al. Beijing: Publishing House of Electronics Industry, 2004: 341-345.

[15] Garofalakis J, Stergiou E. An analytical model for the performance evaluation of multistage interconnection networks with two class priorities[J]. Future Generation Computer Systems, 2013, 29(1): 114-129.

[16] Kruskal C P, Snir M. The performance of multistage interconnection networks for multiprocessors[J]. IEEE Transactions on Computers, 1983, 32(12): 1091-1098.

[17] Agarwal A. Limits on interconnection network performance [J]. IEEE Transactions on Parallel and Distributed Systems, 1991, 2(4): 398-412.

附中文参考文献：

[3] 饶立, 张云泉, 李玉成. 国产百万亿次机群系统Alltoall性能测试与分析[J]. 计算机科学, 2010, 37(8): 186-188.

[4] 刘洋, 曹建文, 李玉成. 聚合通信模型的测试与分析[J]. 计算机工程与应用, 2006, 42(9): 30-33.

[5] 罗红兵, 张晓霞. MPI集合通信性能可扩展性研究与分析[J]. 计算机科学与探索, 2017, 11(2): 252-261.

[7] 李强, 孙凝晖, 霍志刚, 等. MPI Alltoall通信在多核机群中的优化[J]. 计算机研究与发展, 2013, 50(8): 1744-1754.

[14] Duato J, YalamanchiliS, Ni L. 并行计算机互连网络技术: 一种工程方法[M]. 谢伦国, 张民选, 窦强, 译. 北京: 电子工业出版社, 2004: 341-345.

LUO Hongbing was born in 1968. He received the M.S. degree in computer software from Huazhong University of Science and Technology in 1992. Now he is a professor at Institute of Applied Physics and Computational Mathematics. His research interests include parallel computing and performance optimization, etc.

罗红兵(1968—),男,江西永新人,1992年于华中科技大学获得硕士学位,现为北京应用物理与计算数学研究所研究员,主要研究领域为高性能计算,性能优化等。发表学术论文20余篇,承担国家863计划等项目。

ZHANG Xiaoxia was born in 1973. She received the M.S. degree in computer system from National University of Defense Technology in 1998. Now she is a senior engineer at Institute of Applied Physics and Computational Mathematics. Her research interests include parallel computing and performance optimization, etc.

张晓霞(1973—),女,河南焦作人,1998年于国防科学技术大学获得硕士学位,现为北京应用物理与计算数学研究所高级工程师,主要研究领域为并行计算,性能优化等。

WEI Yong was born in 1965. He graduated from Beijing University of Science and Technology in 1990. Now he is a senior technician at Institute of Applied Physics and Computational Mathematics. His research interests include parallel computing and performance optimization, etc.

魏勇(1965—),男,山西襄垣人,1990年毕业于北京科技大学,现为北京应用物理与计算数学研究所高级技师,主要研究领域为并行计算,性能优化等。