着色 Petri 网对高性能集群的建模与性能评估①

黄伟华, 马 中, 戴新发, 徐明迪, 高 毅

(武汉数字工程研究所, 武汉 430205)

摘 要: 针对形式化建模方法导致的状态空间爆炸问题, 提出了一种基于 Petri 网的高性能集群建模与性能评估方法. 首先分析了高性能集群的系统架构, 构建了模型的总体结构; 然后针对集群系统建立了相应的任务产生子模型和调度子模型, 并通过对 Petri 网进行着色, 根据不同种类任务的执行特点设计了相应的任务处理模型. 仿真结果表明, 利用所建立的模型能够有效评估关键参数对集群性能的影响.

关键词: 高性能集群; 着色 Petri 网络; 性能评估; 模型; 仿真

Modeling and Performance Evaluation for High Performance Cluster Based on Colored Petri Net

HUANG Wei-Hua, MA Zhong, DAI Xin-Fa, XU Ming-Di, GAO Yi

(Wuhan Digital Engineering Institute, Wuhan 430205, China)

Abstract: To solve the problem of state space explosion caused by formal modeling methods, a simulation model and performance evaluation method for high performance cluster based on Petri net are proposed. Firstly, the system structure of high performance cluster is analyzed and a general model is constructed. Then the sub-models for task generation and scheduling are built respectively for cluster system. Finally, according to the characteristics of different kinds of tasks, the corresponding task processing models of different tasks are designed by coloring the Petri net. The simulation results show that the proposed model could effectively evaluate the effects of key parameters on cluster performance.

Key words: high performance cluster; colored Petri net; performance evaluation; model; simulation

随着高性能计算技术的不断发展, 集群的规模与结构复杂度不断增加[1,2]. 为了指导高性能集群的系统设计, 有必要在设计过程中对系统进行建模与仿真, 以对其任务性能指标和系统性能指标预先进行评估[3]. 近年来, 基于形式化方法的仿真分析技术取得了长足发展, 大多数研究的焦点集中在利用复杂方法建立模型进而分析建模对象的性能, 如随机过程代数[4], 排队网络[5], 谓词变换逻辑[6]等. 此类研究侧重于系统模型的精确建立, 但是由于"状态空间爆炸问题[7]", 上述方法并不适用于高性能集群的建模与性能评估, 将复杂的集群结构转变化为精确的分析模型有时非常困难甚至是无法实现的[8].

着色 Petri 网(Colored Petri Net, CPN)[9,10]不仅可以构建层次化模型, 允许建模者利用多个彼此联系的 CPN 子模型逐步建立大型复杂系统的层次模型, 且图形化建模方式和用户自定义函数的融合使得在没有降低 CPN 建模能力的前提下, 增强了抽象功能, 降低了模型复杂度, 从而有效地抑制了"状态空间爆炸"的发生[11]. 本文按照任务在集群系统中的处理过程, 为任务的生成与分发、调度和执行分别建立相应的子模型, 从而基于着色 Petri 网的层次化建模方式实现了对高性能集群的建模, 以此对系统性能进行评估.

1 高性能集群系统架构

集群系统利用高速互连网络, 将计算节点按照一定的结构互连, 在集群管理软件的统一调度下实现对任务的高效处理. 与传统的单一高性能计算机相比, 集群系统具有扩展能力强, 可靠性高和性价比突出等

① 基金项目:国家自然科学基金(61502438)

收稿时间:2016-08-18;收到修改稿时间:2016-09-23 [doi:10.15888/j.cnki.csa.005735]

优点.

根据现代集群系统的结构[12], 可将集群中的节点分为两类: 管理节点和计算节点. 其中, 管理节点主要负责任务的接收, 调度, 分发, 计算结果的反馈等功能; 计算节点则主要负责对管理节点分发的任务进行计算与处理.

基于上述特点, 文中将集群抽象为如图 1 所示的系统. 图 1 所示的集群系统抽象模型由若干计算节点和一台任务调度节点组成. 客户端提交的任务首先进入任务队列排队, 随后由调度节点对任务进行分发. 因单机任务和并行计算任务在集群上的处理方式存在较大差别[13], 调度器分别为单机任务和并行计算任务建立了独立的任务队列. 为了与建模语言统一, 便于理解, 后文在建模过程中分别用 Single 型任务和 MPI 型(Message Passing Interface)任务指代上述两种类型的任务. 位于队列中的任务由计算节点依次进行处理. 如图 1 所示, 处理完成的结果首先反馈至调度节点并最终返回给客户.

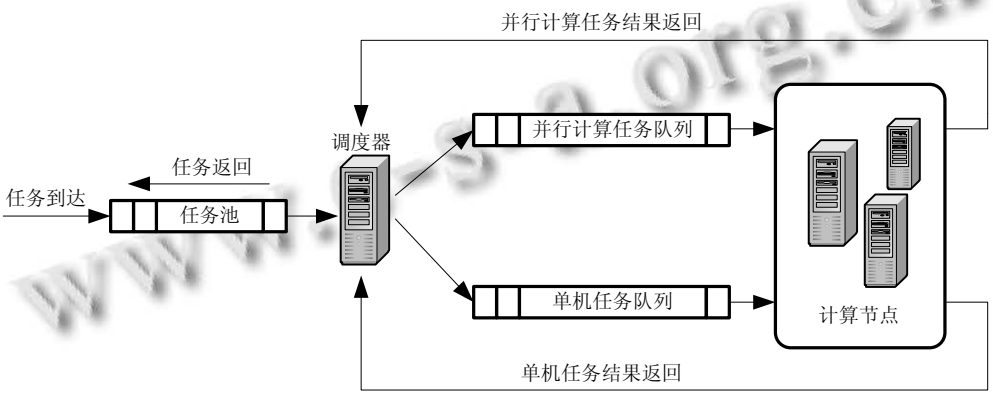

图 1　集群系统架构

为了对集群性能进行评估, 文中后续章节通过 CPN 对上述集群进行建模. 根据任务的处理流程, 文中按照任务在集群系统中的不同生命周期进行建模: 任务的到达与分发, 任务调度, Single 型任务的执行和 MPI 型任务的执行. 为了利用所建立的模型对系统性能进行评估, 建模过程中对系统性能有关键影响的参数均设定为动态可调.

2　高性能集群的CPN建模

CPN 的基本组成包含库所, 变迁和有向弧. 库所表示系统的状态, 变迁表示资源的消耗、使用及系统状态的变化; 变迁的发生受到系统状态的控制, 即变迁发生的前置条件必须满足. 故 CPN 可用如下的形式化三元组[14]来表示:

$$PN=(P,T,F) \tag{1}$$

式(1)中, $P=\{p_1,p_2,\cdots,p_m\}$ 是库所的有限非空集; $T=\{t_1,t_2,\cdots,t_m\}$ 是变迁的有限非空集; $F=(P\times T)\cup(T\times P)$ 是有向弧的集合, P 和 T 还满足 $P\cap T=\Phi$ 且 $P\cup T\neq\Phi$.

在图形化建模方式中, CPN 用圆形表示库所, 用矩形表示变迁. 在本文中, 使用库所表示集群系统的状态, 利用变迁表示导致系统状态发生变化的操作与事件.

在系统建模时, 使用库所和变迁模拟了任务从到达到执行完毕反馈给用户的整个过程, 通过有向弧规定了任务在仿真模型中的流动方向. CPN 允许一个库所中包含多个托肯[15], 每个托肯代表一个任务或任务队列. 文中将其着色, 以区分 MPI 型任务和 Single 型任务这两种不同类型的任务对象. 图 2 是高性能集群的CPN模型总体结构. 因当前的主流CPN建模工具仅支持英文描述, 为忠实于实际模型, 在后文中, CPN 模型图使用默认语言进行表达.

在所建立的模型中, 新任务的产生与到达由变迁子网 Job Arrival 模拟. 变迁子网 Job Arrival 生成 MPI 型和 Single 型两种任务, 以模拟高性能集群中对应的并行计算类任务和非并行计算类任务. 模拟生成的任务随后被放入任务池中, 由调度节点对任务进行统一分发, 此过程的执行实体对应图 1 中的任务调度与分

发服务器. 在使用 CPN 建模时, 任务池由库所 Job Pool 进行模拟. 模型中, 库所 Job Pool 中的任务经过变迁子网 Job Scheduling 调度后分别加入对应的任务队列. 模型中分别用库所 Single Job Queue 和 MPI Job Queue 模拟 Single 型与 MPI 型任务队列缓存.

考虑到实际系统仅具有有限的计算能力, 为了满足任务的 QoS 要求, 调度器会监测任务队列的长度. 如果队列中的任务数已经达到设定的上限值, 调度器可以选择丢弃部分任务以保证 QoS, 此过程在模型中通过变迁子网 Job Scheduling 后续的选择关系[16]进行模拟. 被丢弃任务分别被放入库所 Dropped Single、Dropped MPI 中. 被集群调度器接收的任务将获得执行机会, 具体地, 两种不同任务的执行分别由子网 Single Job Exe 和 MPI Job Exe 进行模拟. 任务执行过程中需要从资源池中获取资源, 使用 CPN 建模时, 系统资源通过库所 Resource Pool 进行模拟. 执行完毕的任务最终分别到达库所 Completed Single 和 Completed MPI 中.

在使用模拟软件 CPN Tools 对高性能集群进行具体建模时, 模型中利用库所 Job Pool、Single Job Queue、MPI Job Queue、Dropped Single、Dropped MPI、Completed Single、Completed MPI 中的托肯代表任务, 颜色是复合型变量 Job, 由任务代号 n、任务类型 jobtype 和时间标记 t 组成.

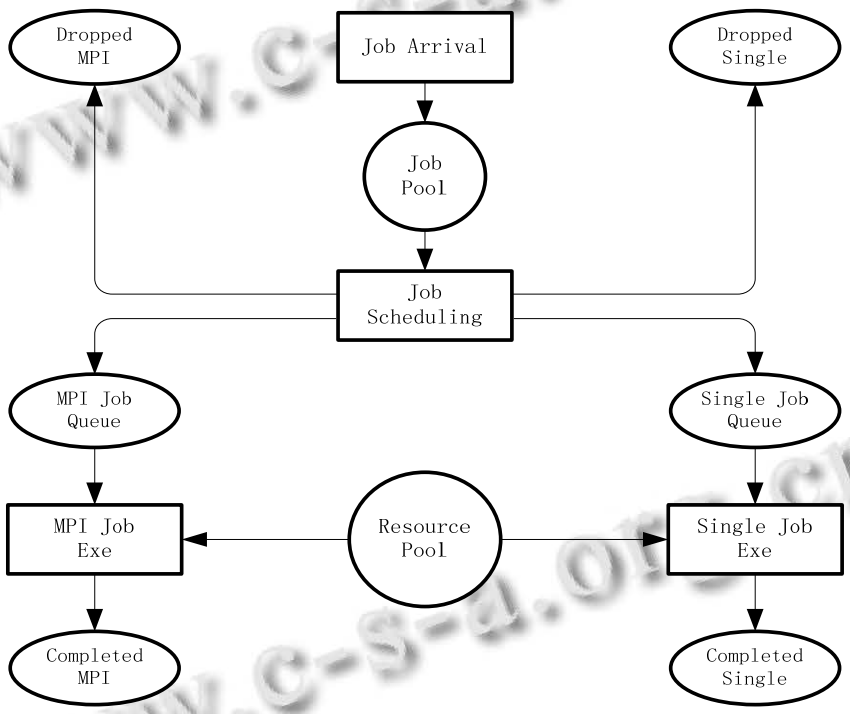

图 2 高性能集群 CPN 模型架构

图 3 是子网 Job Arrival 的结构, 用以模拟任务的产生过程. 变迁 Single Job Generation 到库所 Single Job 的弧函数[17]n+1@+expTime(Next_Single)指定了下一个任务的到达时间. 为了更加贴近实际的集群应用场景, 模型中设定任务到达时间间隔可调, 以模拟不同的任务负载; 与 Single 型任务类似, 下一个 MPI 型任务的到达时间由弧函数 n+1@+expTime(Next_MPI) 指定. 产生的任务最终以队列的方式进行组织并存放进任务池库所 Job Pool 中.

图 4 显示了变迁子网 Job Scheduling 的结构. 任务池库所 Job Pool 中的任务经过两种不同的调度方式 Schedule MPI Job 和 Schedule Single Job 进行调度. 任务随后分流到对应的任务队列缓存 MPI Job Queue、Single Job Queue 中. 为了保证服务的 QoS, 当队列中的任务数量达到上限值时, 允许丢弃后续到达的任务, 模型中对此种场景进行了模拟. 以 Single 型任务为例, 模型中设定其任务队列的上限值为 MaxQueueS, 变迁 Drop Single 通过守卫函数[18]进行条件判断 [QueueLength(S_queue)>MaxQueueS]以决策是否继续接收后续到达的任务. 模型中, 为了维持任务队列的

实时更新, 无论任务是否会被接收都需要向 Job Queue　　反馈信息以更新任务队列缓存.

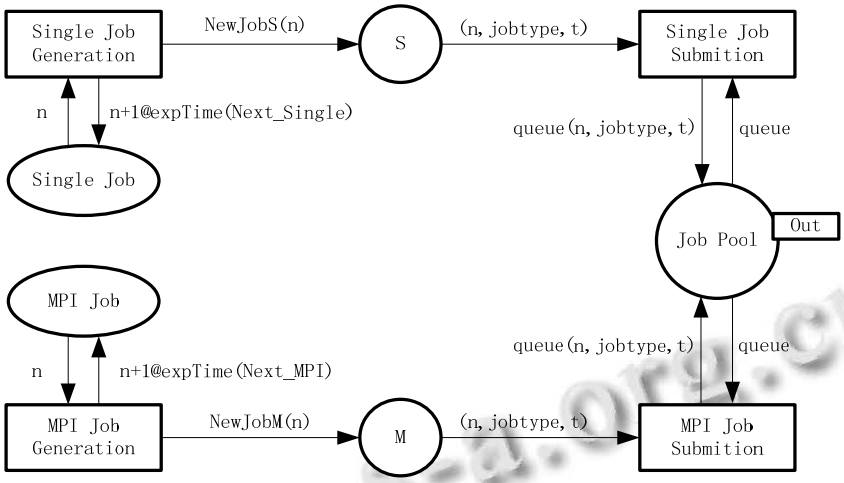

图 3　Job Arrival 子网结构

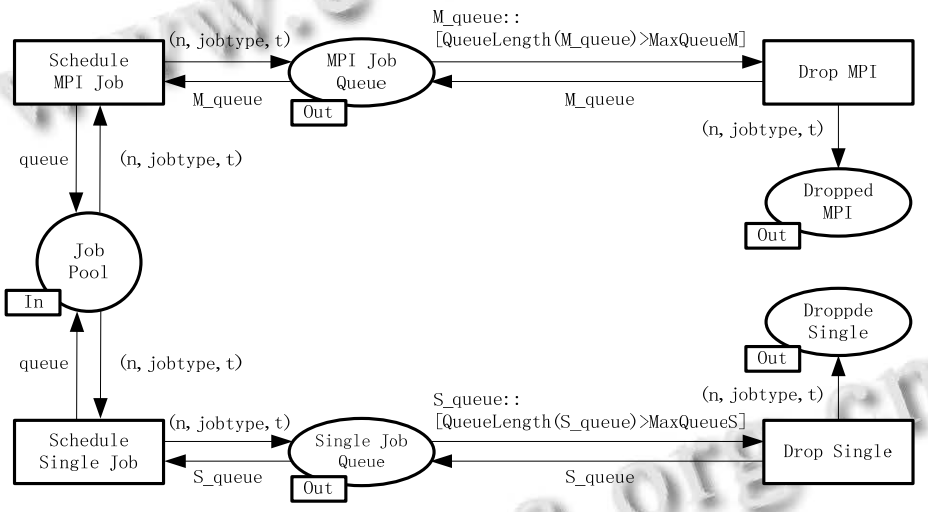

图 4　Job Scheduling 子网结构

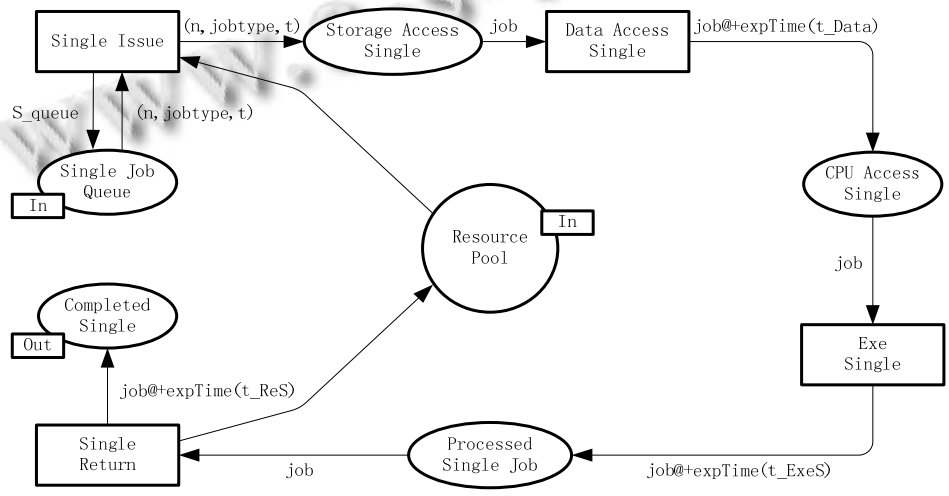

图 5　Single Job Exe 子网结构

如图 5 所示的变迁子网 Single Job Exe 描述了 Single 型任务的数据访问, 任务执行和返回这三个步骤. 建模过程中充分利用了 CPN 的特性, 通过赋时变迁 Data Access Single 的延时函数@+expTime(t_Data)模拟任务访问存储设备所耗费的时间, 通过赋时变迁 Exe Single 的延时函数@+expTime(t_ExeS)模拟任务在处理器上的执行时间, Single Return 的变迁延时 t_ReS 则用来模拟任务返回所需时间.

图 6 所示的变迁子网 MPI Job Exe 建立了 MPI 型任务的执行过程模型. 与 Single 型任务不同的是, MPI 型任务在执行时对计算节点数量的需求由任务规模决定[19]. 针对 MPI 型任务的资源需求特点, 模型中专门为 MPI 型任务设计了资源管理函数 MPI_Resource_Alloc; 根据 MPI 型任务的执行方式[20], 任务以多进程的方式在不同的物理计算节点上并行执行, 执行过程中通过消息传递进行同步, 最后通过汇合操作统一整个执行过程. 图 6 中的模型模拟了 MPI 型任务执行的主要步骤, 且为了回收 MPI 型任务在执行过程中占用的系统资源, 在任务执行结束之前通过 MPI_Resource_Join 函数对所占用的资源进行释放.

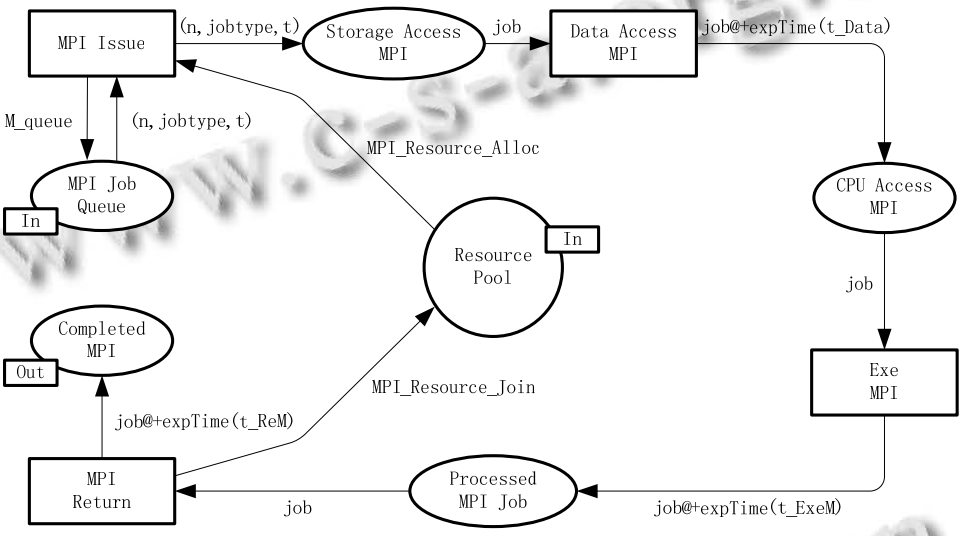

图 6 MPI Job Exe 子网结构

3 仿真实验与结果分析

3.1 评价指标与参数设置

系统 CPN 模型建立之后, 即可通过改变模型参数评估其对系统性能的影响. 为了通过上述模型评估集群系统在不同任务负载与系统配置下的若干关键性能指标, 文中利用不同的模型配置参数进行了仿真.

为表述方便, 此处对后文用到的相关术语进行说明:

任务队列长度: 集群模型的任务队列缓存中, Single 型任务和 MPI 型任务的缓存长度;

资源利用率: 文中主要针对集群中的计算节点计算资源利用率, 定义为一定时间段内被占用的计算节点数量和模型资源池中计算节点总数量的比值;

任务响应时间: 定义为任务被提交至集群系统中的时刻与反馈计算结果时刻之差;

任务丢失率: 定义为被丢弃的任务数与总任务数的比值.

CPN 模型的全局配置参数如表 1 所示.

表 1 仿真参数配置

参数名称	描述	参数配置	单位
Queuelength_S	Single 型任务队列长度	[5,40]	任务
Queuelength_M	MPI 型任务队列长度	[5,40]	任务
Inter_S	Single 型任务到达时间间隔	[10,1000]	秒
Inter_M	MPI 型任务到达时间间隔	[10,1000]	秒
NodeNum	计算节点数量	[10,70]	个

3.2 仿真

为了评估不同任务负载时的系统性能, 文中通过调整任务到达时间间隔对不同轻重的任务负载进行模拟, 并通过仿真对任务到达时间间隔与系统性能的关系进行研究; 考虑到高性能集群的任务队列长度和计

算节点数量对系统性能有较大影响, 仿真过程中通过CPN 模型对相关参数进行动态调整以评估其对系统性能的影响.

基于上述目标, 首先通过所建立的模型对不同任务负载下的任务响应时间和资源利用率进行了仿真; 然后在恒定任务负载下通过对不同任务队列长度的仿真以评估其对任务响应时间和丢失率等关键性能指标的影响; 最后对集群中包含不同计算节点数量下的系统性能进行了仿真, 以评估计算节点数量对系统性能的影响.

3.2.1 任务负载对系统性能的影响

为了评估不同任务负载对系统性能的影响, 此节针对不同的任务到达时间间隔进行仿真. 通过改变任务到达时间间隔的长度, 以仿真不同负载对任务响应时间和系统资源利用率的影响. 仿真时设定任务到达的时间间隔区间为[10,1000], 时间单位设定为秒. 仿真结果如图 7 所示.

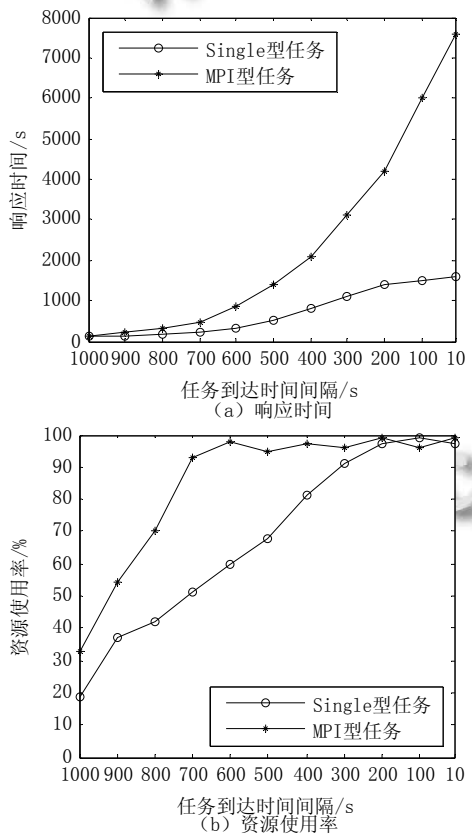

图 7　不同任务负载时的系统性能

由图 7(a)和图 7(b)可知, 在任务到达时间间隔较长时, 此时系统的任务负载相对较低, 任务的响应时间较短, 且资源利用率较低, 显示系统此时具有良好的动态性能; 随着任务到达时间间隔缩短, 集群系统的任务负载加重, 无论是 Single 型任务还是 MPI 型任务, 响应时间均明显上升, 且 MPI 型任务响应时间的上升程度远高于 Single 型任务, 这是因为 MPI 型任务在执行时需要消耗更多资源. 图 7(b)验证了随着任务到达时间缩短, MPI 型任务的资源利用率更容易达到饱和.

3.2.2 任务队列长度对系统性能的影响

在高性能集群中, 任务队列用以实现对任务调度节点分发的任务进行缓存. 队列长度作为高性能集群设计时的关键考虑因素之一, 对系统性能有重要影响. 此节通过改变 CPN 模型中任务队列的长度研究评估其对响应时间和任务丢失率的影响.

在 CPN 模型中固定任务到达时间为恒定值, 分别针对不同的任务队列长度进行仿真. 仿真中设定任务队列长度区间为[5,40], 对此区间内不同任务队列长度对系统性能的影响进行评估. 仿真结果如图 8 所示.

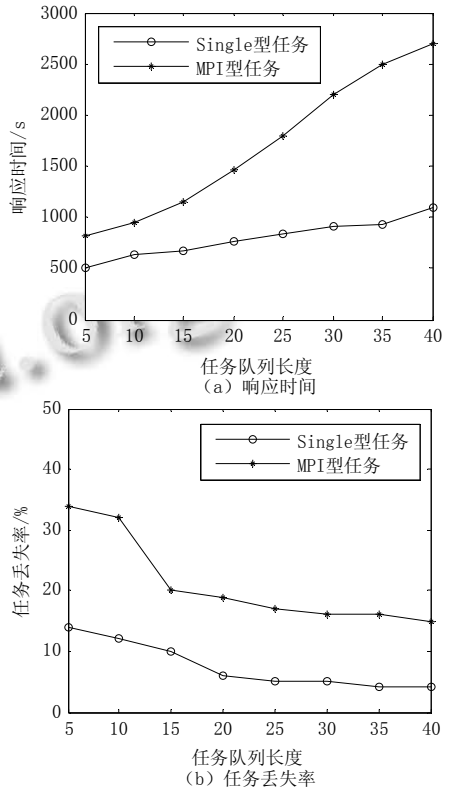

图 8　不同队列长度对系统性能的影响

由图 8(a)可以看出, Single 型任务和 MPI 型任务的响应时间均随着任务队列长度的增加而增长. 其中,

MPI 型任务的响应时间增长更为显著, 这是因为 MPI 型任务的执行需要多个计算节点并行进行, 其响应时间对由任务队列长度增加而导致的任务累积更为敏感. 观察图 8(b)可知, 当任务队列增加到一定长度时, 集群的任务丢失率并未进一步降低, 主要原因在于此时系统的瓶颈在于计算资源, 资源利用率已经接近饱和. 文中后续章节针对集群系统中计算节点数量对任务丢失率的影响进行了仿真研究.

3.2.3 计算节点数对任务丢失率的影响

此节在 CPN 模型中固定任务到达时间间隔为恒定值, 并保持任务队列长度不变, 改变模型中计算节点的数量进行仿真. 仿真中动态调整集群所包含的计算节点数量, 设定计算节点数量区间为[10,70], 以评估不同规模的计算节点对任务丢失率的影响.

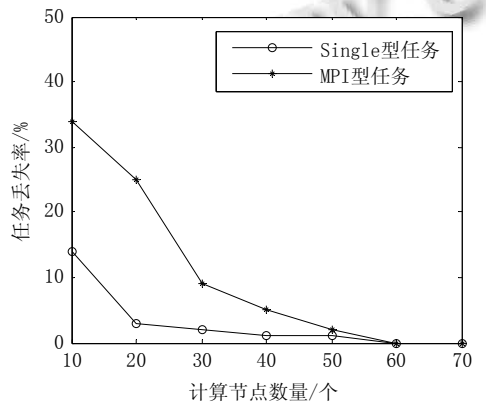

图 9 计算节点数量对任务丢失率的影响

观察图 9 可知, 维持系统的任务负载不变, 并保持相同的队列长度, 集群系统的任务丢失率随着计算节点数的增加迅速下降. 可见, 在因资源使用饱和而导致任务丢失率较高的情况下, 无论是 Single 型任务还是 MPI 型任务, 增加计算节点数量均能够有效降低任务丢失率.

基于所建立的模型对高性能集群的仿真, 可以得出如下基本结论: ⅰ)任务负载的改变引起集群资源使用率的变化, 当高性能集群的资源使用率上升到一定程度时, 系统对任务的响应时间显著增加; MPI 型任务因对资源需求敏感, 响应时间上升更为明显; ⅱ)增加任务队列长度能够在一定程度上降低任务丢失率, 但任务响应时间随之增加, 实质是系统中存在大量任务拥塞; ⅲ)增加集群系统中计算节点的数量能够降低任务丢失率. 提高集群系统的处理能力是降低任务丢失率的最有效方式.

仿真过程表明, 使用着色 Petri 网构建的层次化模型能够对高性能集群的性能进行有效评估, 与随机过程代数等复杂方法相比, 基于着色 Petri 网的建模方法层次明晰, 建模过程也相对简洁高效.

4 结语

当今的高性能集群已变得十分庞大, 结构也越来越复杂, 为系统设计带来了巨大挑战. 文中分析了当前高性能集群的典型架构, 并基于着色 Petri 网建立了集群的仿真与分析模型; 通过不同任务负载和系统配置对高性能集群的性能进行仿真, 预测并评估了关键参数对系统性能的影响, 以期望在设计阶段为系统优化提供参考. 相对于形式化方法, 文中提出的建模方法有效抑制了状态空间爆炸问题. 建立更为精确的系统模型以及提高模型的自动化仿真能力是后续需要完成的工作.

参考文献

1 Niu SC, Zhai JD, Ma XS, Tang X, Chen WG, Zheng WM. Building semi-elastic virtual clusters for cost-effective HPC cloud resource provisionin. IEEE Trans. on Parallel and Distributed Systems, 2016, 27(7):1915–1928.

2 伍康文,柴华.微服务器集群架构的绿色云计算平台.计算机系统应用,2013,22(2):19–25.

3 David B, Felipe O, Jean A, Felipe F, Alberto A, Patricia E, Paulo M, Edward D. Performance evaluation of hypervisors for HPC applications. IEEE International Conference on System, Man and Cybernetics. 2015. 846–851.

4 Zhao GS, Wang HP, Wang J. A novel formal analysis method of network survivability based on stochastic process algebra. Tsinghua Science and Technology, 2007,12(1):175–179.

5 Tribastone M. A fluid model for layered queuing networks. IEEE Trans. on Software Engineering, 2013, 39(6): 744–756.

6 屈婉霞,李暾,郭阳,杨晓东.谓词抽象技术研究.计算机研究与发展,2008,19(1):27–38.

7 Furkan C, Tolga O. Attacking state space explosion problem in model checking embedded TV software. IEEE Trans. on Consumer Electronics, 2015, 61(4): 572–579.

8 Alexander R. Stabilization of a high performance cluster model. International Congress on Ultra Modern Tele-

communications and Control Systems Workshops. 2014. 518–521.

9 Dmitry A. Toward the minimal universal Petri net. IEEE Trans. on Systems, Man, and Cybernetics: Systems, 2014, 44(1): 47–58.

10 Corneli G, Jose M, Giovanni C. CacheSIM: A web cache simulator tool based on colored Petri nets and Java programming. IEEE Latin America Trans., 2015, 13(5): 1511–1519.

11 Juan I, Emilio J, Mercedes P. Simulation-based optimization for the design of discrete event systems modeled by parametric Petri nets. 2011 Fifth UKSim European Symposium on Computer Modeling and Simulation. 2011. 150–155.

12 刘伟,尹行,段玉光,杜薇,王伟,曾国荪.同构 DVS 集群中基于自适应阈值的并行任务节能调度算法.计算机学报, 2013,36(2):393–407.

13 Gao YF, He BS, Zhong J. Network performance aware MPI collective communication operations in the cloud. IEEE Trans. on Parallel and Distributed Systems, 2015, 26(11): 3079–3089.

14 Issam A. Server consolidation using colored Petri nets and CPN tools. IEEE International Conference on Information and Communication Systems. 2015. 32–37.

15 Peng DG, Qian YL, Zhang H, Xia F. Research on fault diagnosis of turbine generation unit based on improved CPN neural network. IEEE International Symposium on System Integration. 2014. 672–677.

16 George F, Mehdi M, Giovanni C. A modular colored stochastic Petri net for modeling and analysis of signalized intersections. IEEE Trans. on Intelligent Transportation Systems, 2016, 17(3): 701–713.

17 Clarimundo M, Rita M, Stephane J. Modeling recursive search algorithms by means of hierarchical colored Petri nets. International Conference on Information Technology. 2015. 788–791.

18 Ding ZH, Zhou Y, Zhou M. Modeling self-adaptive software systems with learning Petri nets. IEEE Trans. on Systems, Man, and Cybernetics Systems, 2016, 46(4): 483–498.

19 Ashwin M, Lokendra S, Feng J, Karthik M, Milind C, Pavan B, Keith R, James D, Feng W, John M, Ma XS, Rajeev T. MPI-ACC: Accelerator-aware MPI for scientific applications. IEEE Trans. on Parallel and Distributed Systems, 2016, 27(5): 1401–1414.

20 刘志强,宋君强,卢风顺,赵娟.基于线程的 MPI 通信加速器技术研究.计算机学报,2011,34(1):154–164.