国 防 科 技 大 学 学 报
第 23 卷 第 1 期 JOURNAL OF NATIONAL UNIVERSITY OF DEFENSE TECHNOLOGY Vol.23 No.1 2001

文章编号:1001-2486(2001)01-0055-04

并行系统可扩性分析研究*

吴建平[1],王正华[1],李晓梅[2]

(1.国防科技大学并行与分布处理国家重点实验室,湖南 长沙 410073;2.总装指挥技术学院,北京 101416)

摘 要:分析了几种已有的可扩性分析模型,并对传统的时间受限与存储受限加速比定律作了新的解释。在此基础上,概括出了可扩性分析的本质,定义了一类一般意义下同构机器与并行算法组成的并行系统的可扩性模型,并由此出发,提出了三种新的可扩性模型:等平均 I/O 需求模型,等平均通信需求模型和等利用率模型。最后探讨了工作站机群与并行算法组成的并行系统的可扩性分析。

关键词:可扩性分析;同构机器;工作站机群;并行算法

中图分类号:TP301 文献标识码:A

Study of the Scalability Analysis for the Parallel Systems

WU Jian-ping[1], WANG Zheng-hua[1], LI Xiao-mei[2]

(1. National Lab of Parallel and Distributed Processing, National Univ. of Defense Technology, Changsha 410073, China;
2. Institute of Command and Technology, General Armament Department, Beijing 101416, China)

Abstract: Several existed models for scalability analysis are introduced. The time-constraint and memory-constraint speedup laws are analyzed from a new viewpoint. Based on these models, we extract the essential of this metric and give a generalized definition for it when the parallel system is composed of a parallel algorithm and an isomorphic architecture. From this definition, present three other models for scalability analysis: equal-average-I/O-requirements model, equal-average-communication-requirements model and equal-utilization model. Finally, We discuss the extension of these models to cluster of workstation systems.

Key words: scalability analysis; isomorphic architecture; cluster of workstations; parallel algorithm

可扩性描述能用增加的处理器来计算更大问题的能力,同时也能描述增大了规模的问题对增加的处理机资源的利用程度。可扩性是并行机和并行算法的组合体即并行系统的一个度量参数,离开机器来谈并行算法的可扩性或者离开算法来讨论并行机的可扩性的作法都不太妥当。

可扩性分析很重要,算法设计者能用其分析算法,及选最优并行算法以充分利用增加的处理器。同时可用其估计取得最佳加速比或其它性能参数,如 T_P 与 $P(T_P)^r$ 等[1]的最佳处理器数。并行机的制造者可用其研究硬件技术对性能的影响。此外,可扩性在性能评价和预测中具有重要作用。

1 同构系统可扩性分析回顾与分析

此处同构系统是指由并行算法与同构的并行机(如 MPP)组成的系统。1967 年,Amdahl 提出了固定问题规模的 Amdahl 定律[2],该定律说明串行部分比例越小,就能用更多机器有效求解,故可扩性越好,但问题规模不一定不变。1988 年 Gustafson 提出了时间受限的 Gustafson 定律[2,3]。如保持运行时间不变,则增加机器规模时,问题规模增长的程度可作为可扩性的度量,但该模型主要衡量的是执行时间。Gustafson 还考虑了存储受限模型[3],该模型考虑机器规模增长时,让问题规模在保持平均存储需求不变的条件下增长。问题规模增加幅度越大,可认为可扩性越好,但衡量的主要是存储资源的利用率。

1987 年 Kumar 与 Rao 提出了等效率模型[4],该模型考虑当机器规模 P 增加时,让问题规模 W 在使并行效率 E 不变的条件下增长。如 W 与 P 在增长时满足函数关系 $W=f(P)$,则称 f 是等效率函数且该并行系统可扩,可扩性为:

* 收稿日期:2000-11-10
基金项目:国家自然科学基金资助项目(69933030);国家 863-306 主题资助项目
作者简介:吴建平(1974-),男,博士生。

$$\mathrm{Scale}(E,P,P') = (P'W)/(PW') = P'f(P)/(Pf(P'))$$

Yong, Xiaodong 与 Qian 认为用平均负载增长来描述可扩性很难作实际分析,从而引入平均延迟来度量可扩性[5]。当问题规模 W 和机器台数 P 都给定时,定义系统的平均延迟为:

$$L(W,P) = \sum_{i=1}^{P}(T_{\mathrm{para}} - T_i + L_i)/P$$

其中 T_{para}是并行执行时间,T_i 是处理机 i 上的执行时间,L_i 是 T_i 中的额外开销时间,T_{para}、T_i 和 L_i 都可测出,从而计算 $L_E(W,P)$时很方便。设用 P 台处理机算规模为 W 的问题时并行效率为 E,平均时间延迟为 $L_E(W,P)$。设处理机数增加到 P' 时,保持 E 不变,要将问题规模增加到 W',记此时平均延迟为 $L_E(W',P')$,则可扩性定义为:

$$\mathrm{Scale}(E,P,P') = L_E(W,P)/L_E(W',P')$$

与等效率模型类似的是由 Xian-He 于 1996 提出的等速度模型[6]。该模型认为如果问题规模 W 与机器规模 P 同时增长时存在一定关系(表现在定理 2 中[6]),使得平均速度不变,则称该并行系统可扩。平均速度是指算法取得的执行速度与处理机数的比值。设在等平均速度 V 条件下,处理机数和问题规模分别增大到 P' 与 W',则该系统的可扩性定义为:

$$\mathrm{Scale}(V,P,P') = (P'W)/(PW')$$

等效率模型和等速度模型分别建立在效率和速度参数上,衡量的分别是效率和速度。

1989 年 Zorbas 等人提出用开销函数度量并行系统的可扩性[4]。设问题规模为 W 时,只能串行执行的部分的运行时间为 $T_{\mathrm{seq}}(W)$,可并行执行的部分的串行执行时间为 $T_{\mathrm{para}}(W)$,则在含 P 台处理机的机器上执行时,理想时间为 $T'(P,W) = T_{\mathrm{seq}}(W) + T_{\mathrm{para}}(W)/P$,设实际时间为 $T(P,W)$。如果对所有 P 都有 $T(P,W) <= T'(P,W)O(P,W)$,则称满足上式的最小 $O(P,W)$为该系统的开销函数。如果当 P 增加时,只要 W 按 P 的一定函数关系增长,就能保持 $O(P,W)$始终为常数不变,则可认为该系统可扩。如果 $O(P,W)$随 P 增长,则 $O(P,W)$决定了系统的不可扩程度。

1996 年,吴幸福等人提出了等平均开销函数模型。设计算规模为 W 的问题时,在 P 台处理机上的时间为 T_P,而在单机上的时间为 T_1,则并行开销为 $T_O(P,W) = PT_P - T_1$,定义系统的平均开销为:

$$T_O^P(P,W) = T_O(P,W)/P = T_P(1 - E_P)$$

如果当 P 增加,W 以 P 的函数形式 $W = f(P)$增长时,恰保持系统的平均开销恒定,则可称 f 是等平均开销函数,且系统可扩。f 的变化率越大,说明要维持等平均开销所要增加的问题规模越大,从而可扩性越好。开销函数模型和等平均开销函数模型都建立在开销函数基础上,主要描述系统开销随机器规模的变化状况。

Kumar 与 Gupta 于 1994 年提出了性能/价格模型[4]。在给定开销下,为达到某个性能指标时处理机的变化与问题规模的变化之间的关系可作为可扩性度量。事实上,可以同时增大机器数和问题规模,使系统取得的性能/价格不变,这时就可以把平均每台处理机上的负载增长率作为可扩性的定义。这个模型反映了为解决某类问题,购买什么样的机器最划算。

2 对时间受限与存储受限模型的另一种解释

对时间受限模型,设 $W = W_{\mathrm{seq}} + W_{\mathrm{para}}$,则 $T = W_{\mathrm{seq}} + W_{\mathrm{para}}/P + T_O(P,W)$。同样设 $W' = W'_{\mathrm{seq}} + W'_{\mathrm{para}}$,则 $T' = W'_{\mathrm{seq}} + W'_{\mathrm{para}}/P' + T_O(P',W')$。设 $W_{\mathrm{seq}} = W'_{\mathrm{seq}}$,则 $T = T'$ 时,有:

$$T_O(P,W) + W_{\mathrm{para}}/P = T_O(P',W') + W'_{\mathrm{para}}/P'$$

问题规模很大时,可认为 W 近似等于 W_{para},W' 近似等于 W'_{para},从而:

$$T_O(P,W) + W/P = T_O(P',W') + W'/P' \tag{1}$$

一般 $T_O(P,W) < T_O(P',W')$,从而 $W/P > W'/P'$。如果存在函数 f 使得当问题规模以 W 增长时,处理机数 P 以 $f(W)$增长恰并行执行时间不变,则称该机器对算法可扩,且可扩性为:

$$\mathrm{Scale}(T,W,W') = (W'P)/(P'W) = (Pf(P'))/(P'f(P)) \tag{2}$$

上式说明 f 变化率越大,为保持执行时间不变,所要增加的处理器数越多,所以 f 反映了可扩程度,称为等执行时间函数。另外由式(1)知:

$$\mathrm{Scale}(T,W,W')=1-(T_O(P',W')-T_O(P,W))P/W \tag{3}$$

上式说明如果随问题规模的增加,在保持执行时间恒定条件下增加系统规模时开销增长越快,可扩性越差。同时问题规模增加时,因为通信等开销的影响,为了在给定时间内求解,必须使机器的增长率大于问题的增长率才能保持执行时间不变。该可扩性的好坏反映了额外开销的影响。

对存储受限模型,设 M,M' 分别是问题规模为 W,W' 时内存消耗量,当机器规模从 P 增到 P' 时,如果存在函数 f,问题规模 W 以 $f(P)$ 增长时,平均内存消耗量 m 恒等,即 $M/P=M'/P'=m$,则称算法对机器可扩,且可扩性定义为

$$\mathrm{Scale}(m,P,P')=(WP')/(PW')=(P'f(P))/(Pf(P'))$$

可以看出 f 的变化率越大,为了保持平均内存消耗量不变,计算的问题要求越大,所以 f 反映了可扩性程度,称为等平均内存消耗函数。该模型衡量了问题求解时对内存资源的利用潜力。如果可扩性差,说明机器设计不合理,大量的内存将被浪费,所以应该把重点放在通信带宽与处理器速度等方面,而不应在本来利用潜力低的情况下再去增加存储量。

3 可扩性的一般定义与三种新模型

如果要主要衡量某个参数(非 P 非 W),可在保持该参数不变的条件下,同时增大 P 和 W,如果 W 的增长率大于 P 的增长率,并且在增长时满足关系 $W=f(P)$,则应该用

$$\mathrm{Scale}(P,P')=(P'W)/(PW')=(P'f(P))/(Pf(P')) \tag{4}$$

衡量并行系统的可扩性,实际反映了算法对机器规模的适应程度。

如 P 和 W 的增长满足 $P=g(W)$ 且 P 的增长率大于 W 的增长率,应该用

$$\mathrm{Scale}(W,W')=(PW')/(P'W)=(W'g(W))/(Wg(W')) \tag{5}$$

衡量并行系统的可扩性,实际反映了机器对问题规模的适应程度。

据衡量参数不同,可把可扩性模型分为两类,即性能分析型和资源分析型。等效率、等速度、等运行时间和等平均开销函数等模型属性能分析型,这类模型最能反映系统的实际性能指标的变化规律。内存受限模型等属于资源分析型,这类模型反映了对资源的利用潜力,资源是否会成为性能瓶颈及程度。对后一类模型,如可扩性差,就应在机器设计时,把这些因素放在次要位置,而把重点放在其它方面。

由此可知,可扩性模型本身并无优劣之分,应看要衡量什么参数。如要衡量并行效率,等效率模型会最好。但如要衡量存储利用率,用等效率模型将不如用存储受限模型。当然,系统性能的总体提高也将在一定程度上取决于待求解问题对资源的需求以及系统所能提供的资源水平之间的关系,从而资源分析模型也将从侧面反映系统性能的变化规律。

3.1 等平均 I/O 需求模型

如果要并行处理的问题的 I/O 频繁,应该衡量算法中的 I/O 需求与机器具体 I/O 能力的匹配程度。一般要保持平均 I/O 能力不变(设每台机器均有 I/O 能力),问题规模变化率会大于机器规模的变化率。如果机器规模 P 增长时,存在函数 f,问题规模 W 以 $f(P)$ 增长恰保持平均 I/O 需求不变,则称该并行系统可扩,可扩性由(4)式给出。f 变化率越大,可扩性越差,f 可作为可扩性的度量,称 f 是等平均 I/O 函数。f 变化率越大,说明问题求解时 I/O 的利用潜力越低,同时也说明性能受 I/O 的影响越小。

3.2 等系统利用率模型

当在机器上求解单一问题时,重点考虑的性能指标是加速比等,但有多个用户程序要求增加处理机以求解更大规模问题时,一种分配方案是尽量把空闲处理器分配给利用率高的程序[2],此时等系统利用率模型十分重要。

设机器数从 P 增到 P' 时,存在函数 f,问题规模 W 以 $f(P)$ 增长恰保持利用率 U 不变,则称 f 是等利用率函数,该并行系统可扩,可扩性由(4)式给出。

该模型反映了算法对系统资源的综合利用潜力,f 变化率越小,表明算法对系统资源的综合利用潜

力大,所以当有多个用户程序同时要求使用空闲处理器时,可优先把其分给用该模型估计到的具有最好可扩性的程序。

3.3 等平均通信需求模型

对基于消息传递的系统,通信问题至关重要,故应考虑算法对通信的需求与机器能提供的通信能力之间的匹配程度。

设机器规模从 P 增到 P' 时,存在函数 f,问题规模 W 以 $f(P)$ 增大恰保持平均通信需求不变,则称 f 是等平均通信需求函数,该并行系统可扩,且可扩性由(4)式给出。

该模型反映了算法对机器提供的通信能力的利用潜力,同时反映了机器的通信能力对算法性能的潜在影响。f 的变化率越小,说明对机器提供的通信能力的利用潜力越大,同时说明问题中的通信需求增长迅速,系统额外开销增长很快,此时并行效率必然受到很大限制。

4 并行系统可扩性模型在 COW 上的探讨

前述可扩性模型以同构机器为基础,但 COW 中任一工作站的计算能力和通信能力均可能不同,故用平均负载的变化率不便于分析 COW 系统。

对 COW 系统,可用计算能力需求与具体计算能力的比值的变化率来刻划其可扩性。COW 系统的计算能力可用各工作站的计算能力之和来评估,如此就可把同构系统的可扩性模型推广到 COW 系统。对于一个开放的 COW 系统,有许多工作站相互连接,运行于其上的应用程序也很多,此时衡量系统性能的最恰当指标是利用率,可把同构系统的等利用率模型推广以分析 COW 系统。

设问题规模为 W,机器计算能力为 C(如 MFLOPS 等)。如机器计算能力 C 增加时,存在函数 f,W 以 $f(C)$ 增长恰保持系统利用率不变,则称 f 是等利用率函数,该系统可扩,且可扩性为:

$$\mathrm{Scale}(C,C') = (C'W)/(CW') = (C'f(C))/(Cf(C'))$$

设用 P 台处理机计算时,第 $i=1,2,\cdots,P$ 台机器的计算能力为 C_i,程序执行时间为 T,在 T 内,第 i 台处理机上的有用操作的时间为 T_i(非等待时间),则系统利用率可定义为:

$$U = (\sum_{i=1}^{P} C_iT_i)/(T\sum_{i=1}^{P} C_i) = (\sum_{i=1}^{P} C_iT_i)/(CT)$$

其它模型也可类似扩展,COW 系统可扩性的深入分析以后将继续研究。

5 小结

本文介绍了几种现有可扩性分析模型,并对传统的时间受限与存储受限加速比定律作了另一解释。在分析这些模型基础上,定义了一般意义下同构机器与并行算法组成的并行系统可扩性分析的一类抽象模型,形成了可扩性分析的一般方法,并由此,提出三种新的可扩性模型:等平均 I/O 需求模型,等平均通信需求模型和等利用率模型。最后探讨了工作站机群与并行算法组成的并行系统可扩性分析。

参考文献:

[1] Alan H K, Horace P F. Measuring parallel processor performance[J]. Communications of ACM, 1990, 33(5): 539-543.

[2] Huang K 著. 王鼎兴,沈美明,郑伟民,温冬婵译. 高等计算机系统结构·并行性·可扩展性·可编程性[M]. 清华大学出版社 & 广西科学技术出版社,1997.

[3] Gustafson J L, Montry G R, Brenner R E. Development of parallel methods for a 1024-processor hypercube[J]. SIAM Journal of Scientific and Statistical Computing, 1988, 9(4): 522-533.

[4] Kumar V, Gupta A. Analyzing Scalability of parallel algorithms and architectures[J]. Journal of Parallel and Distributed Computing, 1994, 22(3): 379-391.

[5] Zhang Y, Yang X D, Ma Q. Software support for multiprocessor latency measurement and evaluation[J]. IEEE Transactions on software engineering, 1997, 23(2): 4-16.

[6] Sun X H, Diane T R. Scalability of parallel algorithm – machine combinations[J]. IEEE Transactions on Parallel and Distributed systems, 1994, 5(6): 599-613.