Linpack标准测试程序及其分析

智能1702 201708010528 莫诗幻

1. Linpack标准测试程序简介：

Linpack是国际上使用最广泛的测试高性能计算机系统浮点性能的基准测试。通过对高性能计算机采用高斯消元法求解一元N次稠密线性代数方程组的测试，评价高性能计算机的浮点计算性能。Linpack的结果按每秒浮点运算次数（flops）表示。

第一个“Linpack基准”报告作为附录在1979年出现在Linpack的用户指南上，其中包括了Linpack软件组件通常的用法途径的数据；针对一个行列大小都为100的矩阵问题，汇集了众多在当时得到广泛使用的计算机系统上(共计有23种)的结果。这样，用户就可以推断出解决他们的矩阵问题所需时间。

Linpack是针对现代并行计算机提出的测试方式，用户在不修改任意测试程序的基础上，可以调节问题规模大小(矩阵大小)、使用CPU数目、使用各种优化方法等等来执行该测试程序，以获取最佳的性能。最初的设计是为了帮助用户在解一个线性方程式时提供执行时间信息。

多人把用Linpack基准测试出的最高性能指标作为衡量机器性能的标准之一。这个数字可以作为对系统峰值性能的一个修正。通过测试求解不同问题规模的实际得分，我们可以得到达到最佳性能的问题规模，而这些数字与理论峰值性能一起列在TOP500列表中。

1. Linpack标准测试程序分类：

Linpack测试包括三类，Linpack100（100×100问题（内部循环优化））、Linpack1000（1000×1000问题(三级循环优化-整个程序)）和HPL：

（1）Linpack100求解规模为100阶的稠密线性代数方程组，它只允许采用编译优化选项进行优化，不得更改代码，甚至代码中的注释也不得修改；

（2）Linpack1000要求求解1000阶的线性代数方程组，达到指定的精度要求，可以在不改变计算量的前提下做算法和代码上做优化；

（3）HPL即HighPerformanceLinpack，也叫高度并行计算基准测试，它对数组大小N没有限制，求解问题的规模可以改变，除基本算法（计算量）不可改变外，可以采用其它任何优化方法。前两种测试运行规模较小，已不是很适合现代计算机的发展。

三、测试原理：

1、HPL采用高斯消元法求解线性方程组。求解问题规模为N时，浮点运算次数为(2/3\*N^3－2\*N^2)。因此，只要给出问题规模N，测得系统计算时间T，峰值=计算量(2/3\*N^3－2\*N^2)/计算时间T，测试结果以浮点运算每秒（Flops）给出。HPL测试结果是TOP500排名的重要依据。

计算机计算峰值：计算机每秒钟能完成的浮点计算最大次数。包括理论浮点峰值和实测浮点峰值。理论浮点峰值是该计算机理论上能达到的每秒钟能完成浮点计算最大次数，它主要是由CPU的主频决定的。计算公式为：理论浮点峰值＝CPU主频×CPU每个时钟周期执行浮点运算次数×CPU数量。而CPU每个时钟周期执行浮点运算的次数是由处理器中浮点运算单元的个数及每个浮点运算单元在每个时钟周期能处理几条浮点运算来决定的。

2、使用HPL基准测试一般需要和收集的信息包括:

（1）Rpeak：系统的最大的理论峰值性能，用GFLOPS表示；

（2）Nmax：获得最高GFLOPS值时的矩阵规模或问题规模；

（3）Rmax：在Nmax规定的问题规模下，达到的最大GFLOPS；

（4）NB：对于数据分配和计算粒度，HPL使用的块尺度NB；

（5）P×Q=系统CPU数=进程数。一般来说一个进程对于一个CPU可以得到最佳性能。对于IntelXeon来说，关闭超线程可以提高HPL性能。P≤Q；一般来说，P的值尽量取得小一点，因为列向通信量（通信次数和通信数据量）要远大于横向通信。P=2n，即P最好选择2的幂。HPL中，L分解的列向通信采用二元交换法（BinaryExchange），当列向处理器个数P为2的幂时，性能最优。例如，当系统进程数为4的时候，P×Q选择为1×4的效果要比选择2×2好一些。在集群测试中，P×Q=系统CPU总核数。如系统为总核数为16核，则P\*Q值应该为4.

3、理论计算：

假设有一个线性方程组求解问题：Ax=b;A∈Rn·n;x，b∈Rn

通过对于方程式n行n+1列的系数矩阵的行基本变换进行LU因式分解，最后得到:

Ly=Pb

Ux=y

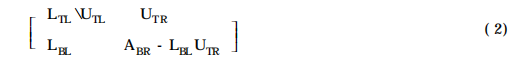
对列主元（即行的基等变换）LU分解算法如下：假设A为m×n矩阵，L为m×n单位下三角矩阵，U为m×n上三角矩阵，P为m×m置换矩阵，P=Pn-1…P1，Pj为第j次循环完成第j行与主元行进行交换的置换矩阵。

1. 基本算法：
2. 假定已得到置换矩阵P1，…，Pj使得:

Pj…P1A=



其中，ATL、LTL和UTL是j×j矩阵，原矩阵被下面的矩阵所覆盖：



把式(1)中右下角部分矩阵仍记为ABR，下面对ABR继续进行列主元LU分解，其步骤如下:

①把ABR划分为如下形式:

ABR=(aB1|AB2)

②找到aB1中绝对值最大的元素及所在行，假设为k，得到主元和置换矩阵Pj+1。

③作变换Pj+1A，实际上是对(ABL|ABR)的第一行和主元行作变换，然后划分ABR为:



④作更新a21←a21a11。

⑤作更新ABR←ABR-a21a12T。

⑥此时ABR为(m-j-1)×(n-j-1)的矩阵阶，若min(m-j-1，n-j-1)>1，则重复前面的步骤，否则结束。

(2)分块算法

在前面的算法中，一阶更新对O(n2)个数据只进O(n2)个操作，算法的计算通信比很低，因此并行计算的性能将很低。为了提高性能，必须采用分块算法。

LU分解的分块算法，只要在划分ABR时把式(3)中的a11用b×b的块矩阵A11代替，相应地a21和a12，T也用(m-j×b)×b和b×(n-j×b)的矩阵A21和A12代替，即可得到分块LU

分解算法:

①用前面的算法分解A11得到A11=L11U11，同时得到A21的更新，仍记为A21。

②解右端下三角线性代数方程组L11U11=A12，仍把U12记为A12。

③计算ABR=ABR-A21A12。

④此时ABR为(m-j×b)×(n-j×b)阶矩阵，若min(m-j×b，n-j×b)>1，则重复前面的步骤，否则结束。

上面的j和b分别是循环次数和分块大小。

四、总结：

机群系统和并行计算有极其广的应用前景，在核爆效应规律研究领域可以发挥重要作用。Lmpack测试为机群系统浮点性能的测试提供了基准，在现有理论基础上，研究Linpack(HPL)测试众多参数的优化配置簽略．大幅减少进行测试所投入的人力物力，对于日益广泛的Linpack测试具有非常现实和重要的意义。