行星流体动力学大规模计算的性能测试与分析

● 王婷 张云泉 孙相征 杨超
中国科学院软件研究所并行软件与计算科学实验室 北京 100190

摘要：
主要介绍了行星流体动力学大规模并行计算中病态压力方程的预条件子和迭代算法的选取，并根据强弱可扩展性的概念，对弱可扩展性的实验结果进行了分析，具体分析了在维持每核平均的浮点操作次数基本不变的情况下，倍增处理器规模时,程序运行时间，MPI消息传递的数量和大小对弱可扩展性的影响。

关键词：预条件，迭代，弱可扩展性，曙光5000A，深腾7000

随着人类对宇宙奥秘的不断探索，行星动力学已成为天文学最为活跃的研究领域之一。鉴于实物探测的难度较大，我们常常采用数值模拟的方法分析探讨行星旋转运动时的物理特征。此类问题时空尺度跨度大，非线性耦合，球形几何形状且快速旋转以及极端的动力学参数等特点，使得必须对其进行大规模并行求解，于是良好的可扩展性对此类问题有重要意义。莫则尧[1]对并行程序提出了实用的性能分析方法，提出用单机性能发挥提高率大于非数值冗余度作为产生超线性加速比的充要条件。陈军[2-3]等提出了一种近优可扩展性，兼顾并行系统的效率和执行时间两个因素。但这种可扩展性的度量标准均需要明确算法的通信和同步开销，给程序的性能分析带来一些难度。现在常用的并行应用程序的可扩展性一般分为强可扩展性(strong scaling)和弱可扩展性(weak scaling)两种。下面基于天体大规模数值模拟软件中的行星流体动力学PETSc程序[4,5]，调研了病态方程的预条件和迭代法的性能，并具体分析在维持平均每核的浮点操作次数基本不变的情况下，倍增处理器规模时，程序运行时间，MPI消息传递的数量和大小对弱可扩展性的影响。

下面的内容安排如下，首先在第2小节介绍了天体大规模数值模拟软件中的行星流体动力学PETSc模拟程序的背景。程序中压力病态方程的预条件和迭代法的性能测试在第3小节。在第4小节中，首先介绍了强弱可扩展性的概念，之后给出了弱可扩展性的测试结果及分析。最后，在第5小节进行了总结并讨论了下一步工作。

1. 行星流体动力学大规模计算程序简介

整体的程序模拟了旋转行星内部球壳中流体的热对流情况[4,6]。程序中先求解速度和温度方程，再求解压力方程（共5个方程，两个大方程组），完成一个时间步迭代后，继续返回速度方程求解下一个时间步。针对在球壳内Boussinesq近似下、不可压缩流体的归一化方程组，用有限差分方法进行离散，时间差分格式采用具有二阶精度的Crank-Nicolson格式，并结合近似因式分解方法分离压强p求解，得到的压力方程矩阵的数据分布情况如图1所示。我们对压力矩阵研究中发现其条件数为无穷，说明它是病态矩阵，这必然使得程序中预条件子，迭代算法，矩阵存储方式等需谨慎选择合适的方法，这些我们将在下一节中具体讨论。

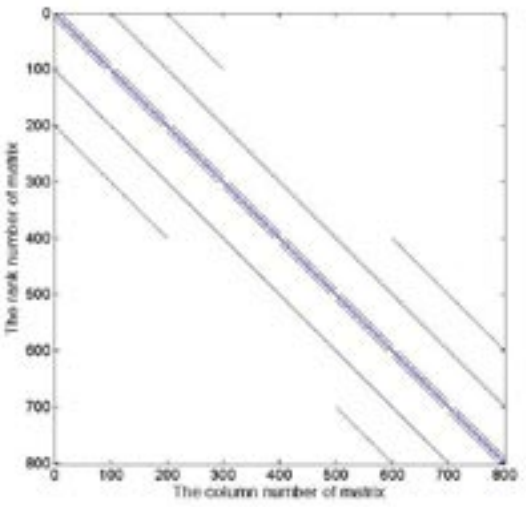

图1 压力方程矩阵数据分布情况

2. 病态矩阵预条件与迭代法测试

2.1 实验背景

我们采取80×80×50和160×160×100两种网格规模，分别记作规模I和规模II，其对应未知量的个数大约是网格规模的5倍，分别约为1.58×10^6和1.272×10^7。为选取合适预条件子和迭代算法，我们在测试中选取了时间步递进的方式，起初测试1个时间步时算子的可行性情况，之后在10个和20个时间步开展了验证性测试。在测试中，我们一致选取迭代法的绝对误差为1.e-15，速度、温度方程的预条件的相对误差为1.e-9，压力方程的预条件的相对误差为1.e-10。在表1-4中使用的是实验平台1，表5-6的结果是基于实验平台2。在实验平台1均采用32进程进行测试，实验平台2中两种规模分别采用128进程和512进程测试。两个实验平台的具体信息如下：

实验平台1：处理器AMD Opteron 8378，8路四核，64G内存，使用Debian squeeze/sid操作系统。采用的编译器为CC-4.4.4和Gfortran-4.4.5，MPI通讯库为MPICH2。

实验平台2：曙光5000A。其计算节点采用四路四核AMD Barcelona (主频1.9GHz)处理器，每个节点64GB内存，计算节点用InfiniBand高速网络进行互联，使用SuSe Linux操作系统。采用的编译器版本为ICC-11.1和IFORT-11.1，MPI通信库为MVAPICH-1.1.0。

PETSc中的适合此行星流体动力学程序中压力矩阵存储格式的预条件子参数类型有jacobi，bjacobi（块雅可比），sor，asm（加型施瓦兹），fieldsplit（场分裂法），mg（多重网格），ksp（Krylov迭代），此外PETSc还提供其他多种适合其他存储格式的预条件子，如pbjacobi（点块雅可比），lu，shell（自定义），eisenstat（Eisenstat技术SSOR），ilu（不完全LU分解），icc（不完全cholesky分解），cholesky，composite（混合），redundant（附加于子进程），nn（纽曼－纽曼），mat，galerkin，exotic（奇异），openmp，asa（自适应光滑聚集），cp（列投影），bfbt(BFBT)，spai（调用外接SPAI包），hypre（调用外接Hypre包），tfs（Tufo-Fischer并行直接算子，适合未知量少的粗网格）。Krylov迭代法中适合的参数类型有cg（共轭梯度），cgne(一般方程共轭梯度)，richardson，chebychev，gmres（广义极小残量），tcqmr（Tony Chan ’s 无转置准最小残量），bcgs(BiCGSTAB，稳定的双共轭梯度)，bcgsl(BiCGSTAB(L))，cgs（共轭梯度平方），tfqmr（无转置准最小残量），bicg(双共轭梯度)，lgmres（松弛广义极小残量），lcd（左共轭方向），另外它还配有其他类型的KSP参数，如nash，stcg，gltr（前3个与共轭梯度相关），ibcgs（改进的稳定双共轭梯度），cr（适应对称共轭梯度预条件），lsqr（最小二乘），preonly（只作用于预条件子），qcg（只适合信赖域算法），fgmres（可调广义极小残量），minres（最小残量），symmlq（只适合对称预条件）。

2.2 实验结果与分析

表1 不同预条件子基于BiCGSTAB迭代算法运算情况（规模I，1个时间步）

预条件子	Jacobi	SOR	BJacobi-ILU	BJacobi-SOR	ASM	Fieldsplit	MG	KSP
迭代次数	156	284	273	284	158	273	442	2
终止容差	2.01E-09	5.33E-09	6.06E-09	5.33E-09	6.94E-09	6.06E-09	9.06E-09	1.63E-14
时间（秒）	3.26	8.22	8.5	7.92	11.32	11.02	32.42	70.85

表2 不同预条件与不同迭代算法运算时间（规模I，1个时间步）

迭代法 / 预条件	BCGS	LGMRES	BCGSL	TFQMR	CGS	BICG	GMRES	LCD	Richardson	Chebychev
Jacobi	4.03	7.07	3.53	溢出	溢出	6.79	8.08	30.6	104.5	112.18
BJacobi-ILU	8.8	9.33	9.34	11.86	12.32	15.38	20.58	36.3	158.65	159.32
SOR	8.73	10.2	7.63	17.44	溢出	溢出	12.05	39.28	73.1	143.38

首先，根据经验BiCGSTAB迭代法是求解病态方程的快速高效算法之一，于是，固定迭代法为BiCGSTAB，测试1个时间步时不同预条件子对压力矩阵求解的影响。其中BJacobi-ILU表示对块雅可比预条件的子块上采用ILU分解求解，同理，BJacobi-SOR表示在子块上用SOR求解。由表1可以看出，此时Jacobi，SOR，BJacobi-ILU，BJacobi-SOR是速度较快的预条件方法。于是，对于雅可比，SOR和块雅克

比三大类预条件，测试不同的迭代算法，结果见表2。进而，对表2中速度较快的BCGS, BCGSL, GMRES, LGMRES的迭代法结合预条件Jacobi，SOR，BJacobi-ILU，BJacobi-SOR在不同网格规模下做了10个时间步的测试，结果见表3,4。由1个时间步扩展到10个时间步是因为在多物理量耦合的方程组求解中，压力方程1次时间步迭代的物理解需带入速度、温度方程右端项并继续求解，仅1个时间步收敛不能完全断定求解的性能。而且从表2，3中可以看出，同等网格规模下，即使选取相同的预条件子和迭代法，10个时间步的运行时间仍远大于1个时间步的10倍，这是因为1次时间步时没有每个时间步完成后将压力方程的物理解带入速度、温度方程右端项计算的时间。最后，还将这些组合的在实验平台2上在20个时间步下，对128核的规模I和512核的规模II做了对比测试，结果见表5，6。

表3 不同预条件与不同迭代算法运算时间
（规模I，10个时间步）

迭代法 / 预条件	BCGS	BCGSL	LGMRES	GMRES
Jacobi	163.82	183.74	204.75	533.95
BJacobi-ILU	164.03	174.16	185.45	385.52
BJacobi-SOR	148.54	157.95	184.13	394.64
SOR	152.84	158.88	191.98	403.49

表4 不同预条件与不同迭代算法运算时间
（规模II，10个时间步）

迭代法 / 预条件	BCGS	BCGSL	LGMRES	GMRES
Jacobi	1679.17	1895.23	2685.58	5521.61
BJacobi-ILU	1475.52	1487.22	1913.46	4202.81
BJacobi-SOR	1687.65	1798.56	2272.45	4513.23
SOR	1689.43	1797.95	2281.23	4519.92

表5 不同预条件与不同迭代法算法运算时间
（规模I，20个时间步，128核）

迭代法 / 预条件	BCGS	BCGSL	LGMRES
Jacobi	40.72	42.34	47.52
BJacobi-ILU	39.64	42.04	49.32
BJacobi-SOR	41.44	43.63	48.93
SOR	39.53	41.47	46.87

表6 不同预条件与不同迭代法算法运算时间
（规模II，20个时间步，512核）

迭代法 / 预条件	BCGS	BCGSL	LGMRES
Jacobi	213.09	237.62	268.05
BJacobi-ILU	237.56	237.31	264.81
BJacobi-SOR	229.11	234.38	288.89
SOR	220.58	220.55	239.15

由以上分析可以看出，在同样的预条件下，BiCGSTAB迭代法基本是应对此病态问题的最快速的迭代算法。而在同样的迭代法下，Jacobi，BJacobi-ILU和SOR算法表现都出不错的性能。我们结合程序中矩阵的存储特征，在后面的测试中选用的BJacobi-ILU预条件子和BiCGSTAB迭代法求解压力方程，且选用此参数进行过上万时间步的迭代求解，程序稳定。

3. 可扩展性

3.1 理论知识

3.1.1 强可扩展性

强可扩展性是考察当总任务固定，增加处理器个数时，程序性能的变化。当任务规模固定，增加处理器个数时。根据Amdahl定律[7-8]，用α表示任务中必须串行执行部分的百分比，则剩下的1-α为可通过p个处理器并行加速的部分，那么增大处理器数目为p时，程序加速比Speedup为：

$$speedup = \frac{1}{\alpha + \frac{1-\alpha}{p}} \quad (1)$$

若程序串行执行时的运行时间为T_s，处理器数目为p时，程序运行时间$T_p=\frac{T_s}{Speedup}$。

程序运行加速比与总任务中串行部分所占的比例成反比，即串行部分所占比例越小，程序运行加速比越大，运行时间减少越明显，程序的强可扩展性越好。当α=0，程序的加速比是线性的。

3.1.2 弱可扩展性

当单个处理器任务不变时。若每个处理器任务中串行部分的运行时间为T_s'，并行部分运行的时间为T_p，串行部分所占的比例为α'，当处理器规模增加到p时，根据Gustafson定律[8]，整个任务在串行执行时，用时为$T_s' + p \times Tp$，程序加速比Speedup ’ 为：

$$Speedup' = \frac{S+p\times P}{S+P} = \alpha' + p \times (1-\alpha') \quad (2)$$

程序运行加速比与单个任务中串行部分所占比例成反比，当α'=0，处理器数量为p时，加速比为p。

弱可扩展性是指单个处理器上的任务量不变，增加处理器个数时，程序性能的变化[9-10]。当问题规模不变处理器规模增大到p倍时，程序加速比为p，运行时间为串行的1/p，程序是强可扩展的(strong scalability)；当保持单个处理器任务规模不变，增大处理器规模时，程序的运行时间保持不变[9]，程序是弱可扩展的(weak scalability)。若保持单个处理器核上

的任务量不变，当处理器规模变大时，问题规模也应随之变大。若将单个处理器上程序运行时间分为两部分：用于通讯的时间T_{comm}，其余的归为计算时间T_{comp}，程序运行时间T可以表示为：

$$T = T_{comp} + T_{comm} \quad (3)$$

在保持单个处理器任务不变增大处理器数目的情况下，T_{comp}不变，而且仅用一个处理器时，$T_{comm}=0$。在处理器数为p时整个程序运行时间与仅用一个处理器时的比值为$1+T_{comm}/T_{comp}$。因此若各处理器任务的通信开销在处理器数目变化时变化很小，或者通信时间开销与计算时间开销相比很小时，程序的通讯时间开销随处理器规模增大对程序的运行时间影响不大，程序的弱可扩展性好。当各个处理器的任务相互独立时，增大处理器个数时，同时增大问题规模，使得程序运行时间保持不变，程序是弱可扩展的。

3.2 实验结果及分析

关于强可扩展性的实验结果及分析，见参考文献[14, 15]。下面仅给出弱可扩展性的结果及分析。

本次测试的实验平台是超级计算机深腾7000。深腾7000采用的是全局共享文件系统，用LSF HPC7.0管理作业调度。实验中在刀片节点（双路Intel Xeon四核处理器，3GHz，32GB内存）的队列提交作业，其中编译器为Intel C/C++ Compiler 11.0.081和Intel Fortran Compiler 11.0.081，MPI通信库为MVAPICH-2-1.2p1。

常见的一些弱可扩展的示例，一般是对于O(N)的算法（N代表问题规模），按处理器核数增加的比例增加问题规模，可以保持单个处理器核上的任务量不变，进而保持程序运行时间不变，说明程序是弱可扩展的。对于复杂度不是O(N)或者通信开销随处理器数目变化较大的程序，同样等比例增加处理器核数和问题规模，单个处理器核上的任务量不会相等，进而不能得到整体相近的运行时间。我们通常用浮点操作次数（FLOP）衡量处理器的任务量，于是，下面的实验研究当平均每核上的浮点操作次数，即平均GFLOP大致相同时（见图2），处理器核数倍增时，不同的网格规模下程序的运行情况。在图10中，从16核以2为倍数增加处理器核数到1024核，平均GFLOP基本不变的网格规模分别为64×64×64, 80×80×64, 96×96×66, 128×128×72, 120×120×106, 170×170×110, 240×240×108。可见对于这种多变量耦合、复杂方程组求解的问题，处理器核数倍增时，网格规模并不是呈线性增长的。

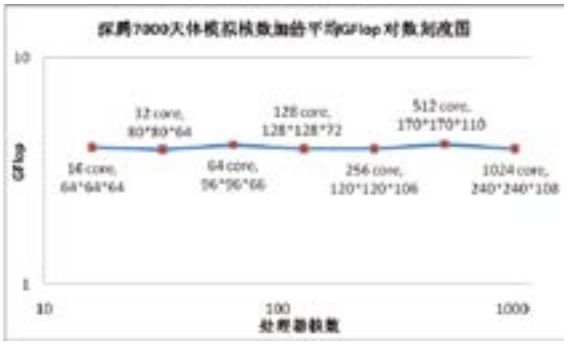

图2 处理器核数加倍时每核平均FLOP对数刻度图

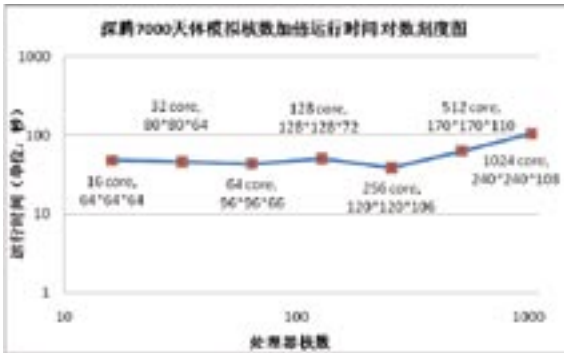

图3 处理器核数加倍时运行时间对数刻度图

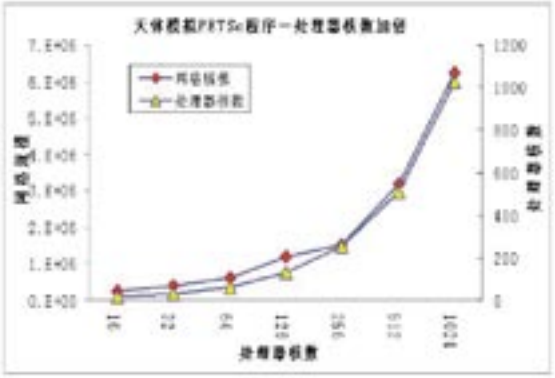

图4 处理器核数加倍时网格规模与处理器核数的变化趋势

我们选择的是平均GFLOP相差不超过5%的实验数据，在图3中可以看出，程序的运行时间在16－256核时变化不大，而在512－1024核逐渐显示出增加趋势。而在图4中可以看出，对于这两种实验，网格规模和处理器核数的变化趋势大致相当。也就是说，若让机器保持一定的任务量运行，对于此程序中的网格规模和核数，可根据一个的变化大致确定另一个的取值。

图5和图6给出了在PETSc中用-log_summary参数命令统计的平均每核的MPI消息长度和数量。可以看出，在保持平均每核浮点操作次数GFLOP大致相当时，处理器核数较少时，如16核，MPI消息的长度和数量均相对较小。而随着处理器核数的增加，如32－256核，平均每核的MPI消息长度逐渐减少，消

息数量逐渐增加，也就是说，MPI消息随着处理器核数和网格规模的增加，由“较少的长消息”变成了“较多的短消息”。此区间程序的运行时间仍可保持基本一致。而在256－1024核区间中，若维持平均GFLOP不变，MPI消息无论在数量还是长度上变化均不再显著，同时程序的运行时间加长，说明处理器虽然像之前一样“卖力”工作（平均GFLOP不变），而且每核上的工作量从数量和大小上均没有较大程度地区别于256核时，但程序运行时间的加长，说明处理器的时间比小规模时更多地耗费在了通信等待上。我们猜测，如果在256－512核间，程序MPI消息的数量和长度仍能延续32－256核的趋势，那么程序将会取得更好的弱可扩展性。

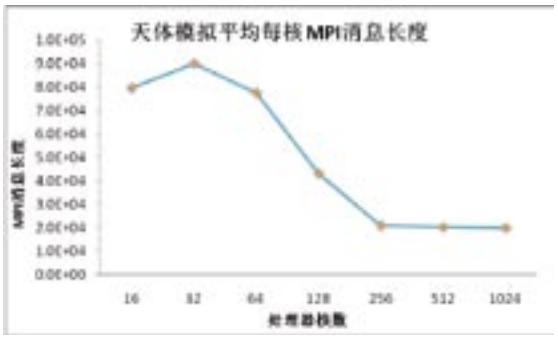

图5 处理器核数加倍时平均每核上的MPI消息长度

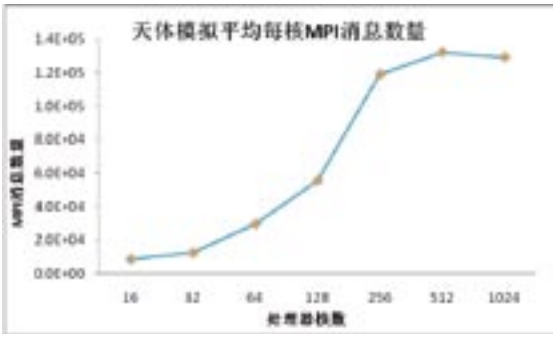

图6 处理器核数加倍时平均每核上的MPI消息数量

4. 总结及下一步工作

本文主要介绍了行星流体动力学大规模并行计算中病态压力方程的预条件子和迭代算法的测试和分析，并根据强弱可扩展性的概念，给出了弱可扩展性的实验结果，并具体分析了在维持平均每核的浮点操作次数基本不变的情况下，倍增处理器规模时，对程序运行时间，MPI消息传递的数量和大小以及弱可扩展性的影响。

要进一步提高程序的性能，增加强弱可扩展性，应在改进大规模计算时病态方程的求解算法，进一步加大程序的数据局部性，尽量减少全局通信上做深入的研究探讨。

参考文献：

[1] 莫则尧。实用的并行程序性能分析方法[J]。数值计算与计算机应用，2000年12月，21(4): 266 - 275。

[2] 陈军，李晓梅。近优可扩展性：一种实用的可扩展性度量[J]。计算机学报，2001年2月，24(2): 179 - 182。

[3] 陈军，莫则尧，李晓梅，袁国兴。大规模并行应用程序的可扩展性研究[J]。计算机研究与发展，2000年11月。

[4] Chao Yang, Ligang Li, Yunquan Zhang. Development of a scalable solver for the Earth ’ s core convection. Lecture Notes in Computer Science, vol. 5938, pp 497 - 502, Springer, 2010.

[5] S. Balay, K. Buschelman, W. D. Gropp, et. al. . PETSc Users Manual [DB/OL], Argonne National Laboratory, 2009.

[6] Li Ligang, Liao Xinhao, Zhang Keke. Countertraveling waves in rotating Rayleigh - Benard convection[J]. Physical review E, 2008, 77(2): 027301.

[7] Amdahl. G.M. Validity of the single - processor approach to achieving large scale computing capabilities[C]. In AFIPS Conference Proceedings, vol. 30 (Atlantic City, N.J.. Apr. 18 - 20). AFIPS Press, Reston. Va., 1967, 483 - 485.

[8] John L. Gustafson, Reevaluating Amdahl ’ s law [J], Communications of the ACM, 1988, 31 (5), 532 - 533.

[9] Alexey L. Lastovetsky, Jack J. Dongaraa, High Performance Heterogeneous Computing [M], Malden MA, USA: Wiley - Interscience, 2009, 104 - 105.

[10] Adolfy Hoisie, Olaf Lubeck, Harvey Wasserman, Performance and Scalability Analysis of Teraflop - Scale Parallel Architectures Using Multidimensional Wavefront Applications[J], International Journal of High Performance Computing Applications, 2000, 14 (4), 330 - 346.

[11] 张云泉，孙家昶，唐志敏，迟学斌。数值计算程序的存储复杂性分析[J]，计算机学报，2000年4月，23(4)：363 - 373。

[12] 张云泉。面向高性能数值计算的并行计算模型DRAM(h) [J]，计算机学报，2003年12月，26(12)：1660 - 1670。

[13] Jing Chen，Yunquan Zhang，Linbo Zhang and Wei Yuan, Performance Evaluation of Allgather Algorithms on Terascale

Linux Cluster with Fast Ethernet[C], Eighth International Conference on High - Performance Computing in Asia - Pacific Region (HPCAsia ' 05), Beijing, Nov.30 - Dec.3, 2005, pp. 437 - 442.
[14] 王婷，孙相征，张云泉等。曙光5000A天体大规模数值模拟性能测试实验与分析[J]。西安交通大学学报。2009年10月，43(10): 71 - 75。
[15] Yuan Yu, Yunquan Zhang, Ting Wang, Jiachang Sun, Yuxin Tang, Xianyi Zhang, Li Rao. Early Performance Evaluation of Dawning 5000A and DeepComp 7000[C]. In the Proceedings of the 15th IEEE The International Conference on Parallel and Distributed Systems (ICPADS 2009), pp. 578 - 585, Dec. 8 - 11, Shenzhen, China.

要闻集锦

SDSC参与超大规模超级计算机研发

据www.hpcwire.com网站2010年11月4日消息报道，美国加利福尼亚大学的圣地亚哥超级计算机中心（SDSC）将为开发下一代超大规模超级计算机项目提供长期技术支持。

该项目是美国防部国防高级研究计划局（DAR - PA）正在推进的“泛在高性能计算（UHPC）”计划的一部分。Intel公司参与了此项目，并在应用方面与SDSC研究人员密切合作。目前，该项目的前两个阶段已延期至2014年，预计在这两个阶段完成一台完整系统的设计和仿真。第三和第四阶段的任务还未授予，初始系统预计2018年完成。

SDSC性能建模与特征（PMaC）实验室将协助In - tel - DARPA项目，分析和映射战略应用以便在Intel硬件上高效运行。值得关注的应用有：实时传感器数据的快速处理、建立图表内复杂的连接关系以及复杂的战略规划。

超大规模计算机的能耗问题是Intel研究小组面临的巨大挑战之一。目前顶级超级计算机的运算水平在千万亿次级（petascale），即每秒执行一千万亿次计算，更高一级就是百亿亿次级（exascale）了，即计算速度达到每秒一百亿亿次，比目前的计算机快1000倍。

Intel公司表示，该项目将把重点放在新的电路拓扑、芯片和系统架构以及编程技术上，可使每次计算所需的能量减少2～3个数量级。这意味着超大规模计算机每次计算所需的能量将比目前能效最高的计算系统减少100到1000倍。

SDSC副主管、超级计算机中心PMaC实验室负责人Allan Snavely表示：“我们正在建立一个集成硬件/软件堆栈，能够极高效的管理数据移动。Intel研究小组的研究人员中还包括PMaC实验室的专家，提供低功耗器件设计、优化编译程序、程序语言表达、高性能应用等方面的专业技术支持。”

Snavely表示，所有领域的工作都必须协调合作，以确保信息在移至存储器分级体系时不会移到较高或较低层次。他表示：“目前原始简单的内存缓存和预取策略不适用于百亿亿次级，因为数据移动伴随着巨大的能耗。现在移动一个字节，甚至是很短的距离，也需要一毫微焦耳（十亿分之一焦耳）。当增加到1018字节时，要移动的话，以今天的技术需要相当于一个核电厂的瞬时功率。”

Intel项目的合作伙伴还有计算机科学与工程学方面顶尖的特拉华州立大学和伊利诺斯州立大学，以及来自Reservoir实验室和ET International的顶级工业研究人员。

根据DARPA本月初发布的版本，UHPC计划完全符合奥巴马总统在“美国创新战略”中所表达的主要优先级事项。优先级事项包括：世纪重大挑战百亿亿次级超级计算、高能效计算以及劳动生产率。由此产生的UHPC功能将提供至少50倍能量、计算和生产效率，大大削减设计、开发复杂的计算应用所需的时间。

（金 溪）