Parallelization of SSIIM on Njord and Multi-core PCs using OpenMP

by Nils Reidar B. Olsen, NTNU, Norway

23. January 2008.

Introduction

SSIIM is a 3D CFD program made in the C language. It has been developed over the last 17 years. The program is used in hydraulic engineering and sediment transport.

There are two versions of SSIIM. SSIIM 1 has a structured grid and SSIIM 2 has an unstructured grid. SSIIM 1 has some more sequential code and more OpenMP forks than SSIIM 2.

Computers and compilers

Njord is a UNIX system with 56 nodes, where each node is a shared-memory system with 8 dual-core power5+ chips (p575) that share 32 GB memory. Maximum number of processors on Njord is 16 for OpenMP applications, but in a special SMT mode, this can be split into 32 logical processors. The processors have a speed of 1.9 GHz, 1.9 MB L2 cache and 32 MB L3 cache. Each node has 8 memory controllers. The memory bus speed is 1066 MHz, with 64 parallel accesses.

Stallo is a Linux cluster with 704 nodes of two quad-core Xeon processors, running at 2.66 GHz. Each node has 16 GB RAM.

The dual core PC has an Intel Core 2 CPU (E6400) running at 2.13 GHz, with 2 GB RAM. This processor has 2 MB L2 cache, and a front side bus speed of 1066 MHz.

The quad core PC has an Intel Core 2 Quad CPU (Q6700) running at 2.66 GHz, with 2 GB RAM. This processor has 8 MB L2 cache, and a front side bus speed of 1066 MHz.

The 8 core Dell PC with Xeon processors runs at 2 GHz, and it has 16 GB RAM.

The compiler switches used on Windows were:

Microsoft*: cl -openmp -Ox -arch:SSE2 -TC*

Intel*: icc /c /Ox /Ob1 /D "NDEBUG" /D "_CONSOLE" /D "WIN32" /D "_MBCS" /D "_VC80_UPGRADE=0x0600" /GF /FD /EHsc /MT /GS /Gy /GR /o".\Release/" /W3 /nologo /Gd /Qopenmp /QaxT*

The compiler switches used on Njord were:

xlc_r -O3 -qstrict -qarch=pwr5 -qtune=pwr5 -qsmp=omp -lm

The compiler switch on Stallo was *icc -openmp -lm -O3*

SSIIM 1

For SSIIM 1, the 3D test case is of flow over natural dunes in the delta of Lake Øyern. The grid has 520x220x10 = 1.14 million cells. The RAM requirement is 727 MB. The test case was run for 20 iterations. The intel compiler was used for all machines.

Table 1: Total computing time in seconds, SSIIM 1

No. processors	Njord	Stallo	Quad core Pentium	Dual core Pentium 3GHz	8 core Xeon 2GHz
sequential	234	230	201	195	
1	261	253	233	225	326
2	149	158	150	159	189
4	76	107	143		160
8	33	101			127
16	21				
32 (SMT)	37				

The case was run without block-correction. Using block-correction one time in streamwise direction for the pressure-correction equation, increased the computational time to 21.7 seconds for 16 processor run on Njord. This block-correction algorithm was sequential.

SSIIM 2

SSIIM 2 has an unstructured grid. The 3D test case consisted of 595 010 cells from a rectangular pool with inflow on the left side and outflow at the top. The case needed 880 MB RAM to store the grid. The case was run for 20 iterations in a steady state solver. The test used a multi-grid algorithm with some sequential code. Without the multi-grid solver, the speedup and linearization was marginally better.

.

Table 2: Total computing time in seconds, SSIIM 2

No. processors	Njord	Quad PC Intel compiler	Dual core PC Intel compiler	Dual core PC MS compiler
sequential		48.3	55.7	66.5
1	90.6	49.3	54.4	65.4
2	49.3	34.9	37.8	43.4
4	21.4	32.8		
8	10.6			
16	6.2			
32 (SMT)	5.0			

Scaling of individual algorithms

Testing the speed of the program was done using the *omp_get_wtime* function. This enabled only the computational time to be estimated, without reading/writing input/result files. It also enabled an estimation of linearization efficiency of the individual algorithms of the programs.

Table 3: Scaling of different algorithms on Njord, SSIIM 1

Algorithm	1-2 speedup|	1-4 speedup	1-8 speedup	1-16 speedup
coefficient generation, velocities, k and epsilon	2.0	3.9	7.8	12.9
coefficient generation, SIMPLE	2.0	3.8	7.6	12.7
solver	1.7	3.5	9.6	17.5
residuals	1.9	4.0	8.9	16.7
turbulence source terms	2.0	3.9	8.6	13.6
wall-laws source terms	2.0	4.0	9.0	14.2
SIMPLE, corrections	2.0	4.2	9.2	14.8
Total program	1.75	3.46	8.02	12.51

There are some linear code segments between the main algorithms, making the speedup of the total program slower than the average of the main algorithms.

Conclusions:

On a dual-core PC, SSIIM 1 scales with a factor 1.4 between one and two processors. SSIIM 2 scales with a factor 1.5. On the Njord machine, SSIIM 1 scales with an average factor 1.8 between one and 16 processors. The SSIIM 2 program scales linearly between 1 and 16 processors on Njord (average factor 2.0).

The PC scales very poorly, especially for more than two processors. This is most likely due to limitations in the capacity of the data transfer between memory and CPU. The conclusion is based on two facts:

1. Njord has much better capacity to transfer data between memory and CPU than the PC
2. If only the solver is parallelized with OpenMP, then this part of the program scales linearly also on the PC. But once the rest of the program is parallelized, then the solver does not scale linearly anymore. This test is done by printing time-stamps at different parts of the program (*omp_get_wtime*).

On the quad-core PC, the SSIIM 1 program actually ran faster on three cores (139 seconds) than four cores (143 seconds).