

Citation for published version: Sun, L & Zang, J 2013, 'OpenMP implementation for Fortran on HPC' HPC symposium , Bath, UK United Kingdom, 4/06/13 - 4/06/13, .

Publication date: 2013

**Document Version** Early version, also known as pre-print

Link to publication

### **University of Bath**

**General rights** Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

# OpenMP implementation for FORTRAN on HPC

Speaker: Dr Liang Sun

Email: <u>*l.sun@bath.ac.uk</u>*</u>

Co-author: Dr Jun Zang

Email: j.zang@bath.ac.uk

Department of Architecture and Civil Engineering University of Bath

4th June, 2013 · University of Bath

## Background









### 4th June, 2013 · University of Bath

## Numerical method



## Performance of sequential executable



#### 4th June, 2013 · University of Bath

## Better LU solver in Intel MKL

## LAPACK Routines: Linear Equations

### ?gesv

Computes the solution to the system of linear equations with a square matrix A and multiple righthand sides.

### **Syntax**

Fortran 77:

call zgesv( n, nrhs, a, lda, ipiv, b, ldb, info )

### **Reference:**

Intel® Math Kernel Library Reference Manual

http://software.intel.com/en-us/articles/intel-math-kernel-library-documentation

## Performance of sequential executable with MKL



#### 4th June, 2013 · University of Bath

## Implementation of OpenMP



#### 4th June, 2013 · University of Bath

## Data race



### Data-Sharing Attributes

42 !\$omp parallel default(none) &

43 !\$omp shared(node\_body,xyz\_p,xyz,amata,ncon,ncon\_p,rsn,nphi,nsys,v,nele\_body,ncn,bmata,nnode\_p,nelem) &
44 !\$omp private(inode,xp,yp,zp,value,bmat,ielem,i,check,wmat,wmat1,ip,j,jncon,ith,is,xyzco,el,dist,kk)

### > Threadprivate Directive

107 C\$omp threadprivate(/FGRIGR/,/HCOEF/)

4th June, 2013 · University of Bath

## Speedup of parallel executable



 $T_1$  is the execution time of the sequential algorithm  $T_n$  is the execution time of the parallel algorithm with *n* cores

## Efficiency of parallel executable



 $T_1$  is the execution time of the sequential algorithm  $T_n$  is the execution time of the parallel algorithm with *n* cores

### 4th June, 2013 · University of Bath

## Concluding remarks

Optimized LU solver in Intel MKL improves performance significantly

 OpenMP has been implemented successfully in current FORTRAN codes and all data race problems have been solved

 Running multithreaded executable for small problems is not economical considering total computational time. In large problems, much time can be saved by using parallel algorithm

# Acknowledgement

These computations were performed on the University of Bath's High Performance Computing Facility. Provision of services by BUCS HPC Support Team is gratefully acknowledged.

## Thank You !

## **Additional Information**

## Hardware and software on HPC



**HPC** Node

Intel FORTRAN Compiler Module: *icomp/11.1.075* 

Intel Math Kernel Library (MKL) Module: *imkl/10.2.7.041* 

Com

Compiler and Library

## Generation of sequential executable

GNU Make 3.81 Copyright (C) 2006 Free Software Foundation, Inc. This is free software; see the source for copying conditions. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

This program built for x86 64-redhat-linux-gnu



## Generation of sequential executable with MKL

Select Intel® product:	Intel(R) MKL 10.2	•
Select OS:	Linux*	•
Select compiler:	Intel(R) Fortran	•
Select architecture:	Intel(R) 64	•
Select dynamic or static linking:	Dynamic	
Select interface layer:	LP64 (32-bit integer)	
Select sequential or multi-threade	ed layer: Sequential	•
Select OpenMP library:	<select openmp=""></select>	v
Select cluster library:	CDFT (BLACS required) SCaLAPACK (BLACS required) BLACS	I)
Select MPI library:	<select mpi=""></select>	-
Select the Fortran 95 interfaces:	BLAS95 LAPACK95	
Link with Intel® MKL libraries expl	icitly:	
Use this link line: -L\$ (MKLROOT) /lib/em64t \$ (MI lmkl_intel_lp64 -lmkl_sequer Compiler options:	(LROOT)/lib/em64t/libmkl_lapack95_l ttial -lmkl_core -lpthread -lm	p64.a -
-I\$(MKLROOT)/include/em64t,	'lp64 -I\$(MKLROOT)/include	

LIBS = -L\$(MKLROOT)/lib/em64t \$(MKLROOT)/lib/em64t/libmkl\_lapack95\_lp64.a -lmkl\_intel\_lp64 -lmkl\_sequential -lmkl\_core -lpthread -lm

### 4th June, 2013 · University of Bath

## Generation of parallel executable

Select OS:       Linux*         Select compiler:       Intel(R) Fortran         Select architecture:       Intel(R) 64         Select dynamic or static linking:       Dynamic         Select interface layer:       LP64 (32-bit integer)         Select sequential or multi-threaded layer:       Multi-threaded         Select OpenMP library:       Intel(R) (libiomp5)         Select Cluster library:       CDFT (BLACS required)         Select MPI library:       ScalAPACK (BLACS required)         Select the Fortran 95 interfaces:       BLAS95         Link with Intel(® MKL libraries explicitly:       LAPACK95         Link with Intel(® MKL libraries explicitly:       Intel	Select Intel® product:	Intel(R) MKL 10.2	-
Select cosh       Linux       •         Select compiler:       Intel(R) Fortran       •         Select architecture:       Intel(R) 64       •         Select dynamic or static linking:       Dynamic       •         Select interface layer:       LP64 (32-bit integer)       •         Select interface layer:       LP64 (32-bit integer)       •         Select openMP library:       Intel(R) (libiomp5)       •         Select cluster library:       CDFT (BLACS required)       •         Select cluster library:       CDFT (BLACS required)       •         Select the Fortran 95 interfaces:       BLAS95       •         Select the Fortran 95 interfaces:       BLAS95       •         Link with Intel® MKL libraries explicitly:       •         Use this link line:       •         -L\$ (MKLROOT) / lib/em64t \$ (MKLROOT) / lib/em64t / libmk1_lapack95_lp64.a -         Inkl_intel_lp64 -lmk1_intel_thread -lmk1_core -lpthread -lm	Select OS:		
Select complier:       Intel(R) Fortran         Select architecture:       Intel(R) 64         Select dynamic or static linking:       Dynamic         Select interface layer:       LP64 (32-bit integer)         Select sequential or multi-threaded layer:       Multi-threaded         Select OpenMP library:       Intel(R) (libiomp5)         Select cluster library:       CDFT (BLACS required)         Select MPI library:       CDFT (BLACS required)         BLACS       Select the Fortran 95 interfaces:         Select the Fortran 95 interfaces:       BLAS95         Link with Intel® MKL libraries explicitly:       Intel(KLR00T) / lib/em64t \$ (MKLR00T) / lib/em64t / libmk1_lapack95_lp64.a -         Inkl_intel_lp64 - lmk1_intel_thread - lmk1_core -lpthread -lm       Inkl_intel_lp64 - lmk1_intel_thread -lmk1_core -lpthread -lm		Linux*	
Select architecture:       Intel(R) 64         Select dynamic or static linking:       Dynamic         Select interface layer:       LP64 (32-bit integer)         Select sequential or multi-threaded layer:       Multi-threaded         Select OpenMP library:       Intel(R) (libiomp5)         Select cluster library:       CDFT (BLACS required)         Select MPI library:       CDFT (BLACS required)         Select the Fortran 95 interfaces:       BLACS         Select the Fortran 95 interfaces:       BLAS95         Link with Intel® MKL libraries explicitly:       Intel(RLROOT)/11b/em64t \$ (MKLROOT)/11b/em64t/11bmk1_lapack95_lp64.a - 1mk1_intel_lp64 -lmk1_intel_thread -lmk1_core -lpthread -lm	Select compiler:	Intel(R) Fortran	-
Select dynamic or static linking:       Dynamic         Select interface layer:       LP64 (32-bit integer)         Select sequential or multi-threaded layer:       Multi-threaded         Select OpenMP library:       Intel(R) (libiomp5)         Select cluster library:       CDFT (BLACS required)         Select MPI library:       CDFT (BLACS required)         BLACS       Select MPI library:         Select the Fortran 95 interfaces:       BLAS95         Link with Intel® MKL libraries explicitly:       LAPACK95         Link with Intel® MKL libraries explicitly:       Imkl_intel_intel_thread -lmkl_core -lpthread -lm	Select architecture:	Intel(R) 64	-
Select interface layer:       LP64 (32-bit integer)         Select sequential or multi-threaded layer:       Multi-threaded         Select OpenMP library:       Intel(R) (libiomp5)         Select cluster library:       CDFT (BLACS required)         Select MPI library:       Select MPI>         Select the Fortran 95 interfaces:       BLAS95         VLAPACK95       Link with Intel® MKL libraries explicitly:         Use this link line:       -L\$ (MKLROOT)/lib/em64t/libmk1_lapack95_lp64.a - lmk1_intel_lp64 -lmk1_intel_thread -lmk1_core -lpthread -lm	Select dynamic or static linking:	Dynamic	-
Select sequential or multi-threaded layer:       Multi-threaded         Select OpenMP library:       Intel(R) (libiomp5)         Select cluster library:       CDFT (BLACS required)         ScaLAPACK (BLACS required)       BLACS         Select MPI library: <select mpi="">         Select the Fortran 95 interfaces:       BLAS95         Link with Intel® MKL libraries explicitly:          Use this link line:       -L\$ (MKLROOT)/lib/em64t \$ (MKLROOT)/lib/em64t/libmk1_lapack95_lp64.a - lmk1_intel_lp64 -lmk1_intel_thread -lmk1_core -lpthread -lm</select>	Select interface layer:	LP64 (32-bit integer)	•
Select OpenMP library:       Intel(R) (libiomp5)         Select cluster library:       CDFT (BLACS required)         ScaLAPACK (BLACS required)       BLACS         Select MPI library:       Select MPI>         Select the Fortran 95 interfaces:       BLAS95         LAPACK95       Link with Intel® MKL libraries explicitly:         Use this link line:       -L\$ (MKLROOT)/lib/em64t/libmk1_lapack95_lp64.a - lmk1_intel_lp64 -lmk1_intel_thread -lmk1_core -lpthread -lm	Select sequential or multi-threaded layer:	Multi-threaded	-
Select cluster library: CDFT (BLACS required) ScaLAPACK (BLACS required) BLACS Select MPI library: Select the Fortran 95 interfaces: BLAS95 Link with Intel® MKL libraries explicitly: Luse this link line: -L\$ (MKLROOT) / lib/em64t \$ (MKLROOT) / lib/em64t / libmkl_lapack95_lp64.a - lmkl_intel_lp64 -lmkl_intel_thread -lmkl_core -lpthread -lm	Select OpenMP library:	Intel(R) (libiomp5)	-
Select MPI library: <select mpi="">         Select the Fortran 95 interfaces:       BLAS95         W LAPACK95         Link with Intel® MKL libraries explicitly:         Use this link line:         -L\$ (MKLROOT)/lib/em64t \$ (MKLROOT)/lib/em64t/libmkl_lapack95_lp64.a -         lmkl_intel_lp64 -lmkl_intel_thread -lmkl_core -lpthread -lm</select>	Select cluster library:	CDFT (BLACS required) CSCaLAPACK (BLACS required) BLACS	E
Select the Fortran 95 interfaces: BLAS95 Link with Intel® MKL libraries explicitly: Use this link line: -L\$(MKLROOT)/lib/em64t \$(MKLROOT)/lib/em64t/libmk1_lapack95_lp64.a - lmk1_intel_lp64 -lmk1_intel_thread -lmk1_core -lpthread -lm	Select MPI library:	<select mpi=""></select>	Ŧ
Link with Intel® MKL libraries explicitly:	Select the Fortran 95 interfaces:	ELAS95	
Use this link line: -L\$(MKLROOT)/lib/em64t \$(MKLROOT)/lib/em64t/libmkl_lapack95_lp64.a - lmkl_intel_lp64 -lmkl_intel_thread -lmkl_core -lpthread -lm	Link with Intel® MKL libraries explicitly:		
L\$(MKLROOT)/lib/em64t \$(MKLROOT)/lib/em64t/libmkl_lapack95_lp64.a - lmkl_intel_lp64 -lmkl_intel_thread -lmkl_core -lpthread -lm	Use this link line:		
	-L\$(MKLROOT)/lib/em64t \$(MKLROOT)/ lmkl_intel_lp64 -lmkl_intel_thread	lib/em64t/libmkl_lapack95_lp64.a - -lmkl_core -lpthread -lm	
Compiler options:	Compiler options:		
	openny =14 (nk1kod) / include/ enote	1901 -19 (HRENOUL) / Include	4

### 4th June, 2013 · University of Bath

## References

- <u>https://wiki.bath.ac.uk/display/HPC/OpenMP</u>
- Intel® Math Kernel Library Reference Manual (<u>http://software.intel.com/en-us/articles/intel-math-kernel-library-documentation</u>)
- Intel® Math Kernel Library Link Line Advisor (<u>http://software.intel.com/en-us/articles/intel-mkl-link-line-advisor</u>)
- Chapman, B., Jost, G., van der Pas, R., 2007. <u>Using OpenMP:</u> <u>Portable Shared Memory Parallel Programming</u>. MIT Press, Cambridge, Massachusetts, USA.
- <u>www.openmp.org</u>