



Hierarchical Multigrain Parallelization

M. Obata^{1,2}, K. Ishizaka^{1,2}, K. Kimura^{1,2}, T. Tobita^{1,2}, T. Kodaka^{1,2}, A. Murai², K. Kaminaga², K. Nagasawa², J. Shirako², A. Yoshida^{1,3}, H. Kasahara^{1,2} (APC Technology Group, ²Waseda University, ³Toho University)

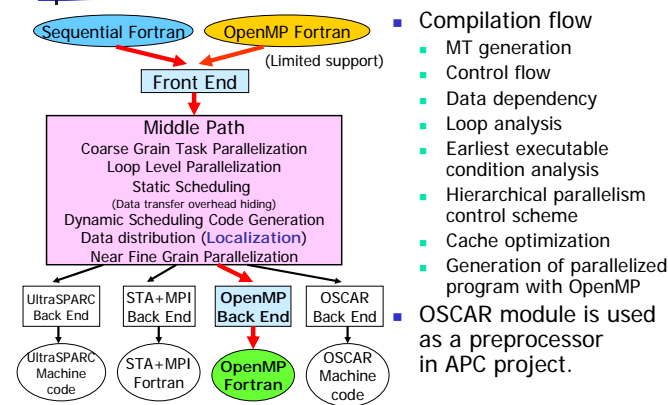
Multigrain parallelism

- Wide use of SMP from single chip multiprocessors to high performance computers
- The gap between peak performance and effective performance is getting larger
- Reaching maturity of loop level parallelization

Multigrain parallel processing has been proposed to improve effective performance.

- Coarse grain parallelism
- Loop level parallelism
- Near fine grain parallelism

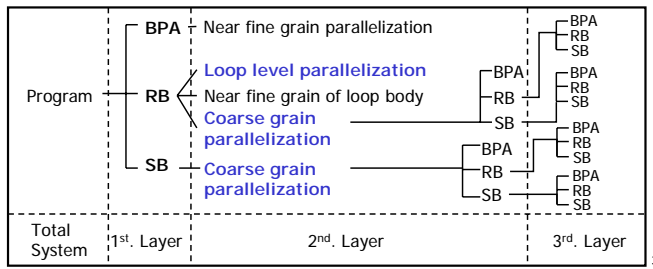
OSCAR module for multigrain parallelization



- Compilation flow
 - MT generation
 - Control flow
 - Data dependency
 - Loop analysis
 - Earliest executable condition analysis
 - Hierarchical parallelism control scheme
 - Cache optimization
 - Generation of parallelized program with OpenMP
- OSCAR module is used as a preprocessor in APC project.

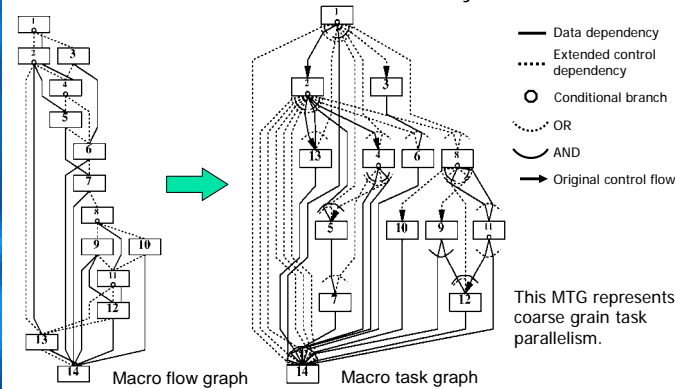
Generation of macro-tasks

- Program is decomposed into macro-tasks (MTs).
 - Block of Pseudo Assignments (BPA): Basic Block (BB)
 - Repetition Block (RB) : natural loop
 - Subroutine Block (SB) : subroutine



Generation of macro-task graph

- Earliest Executable Condition analysis

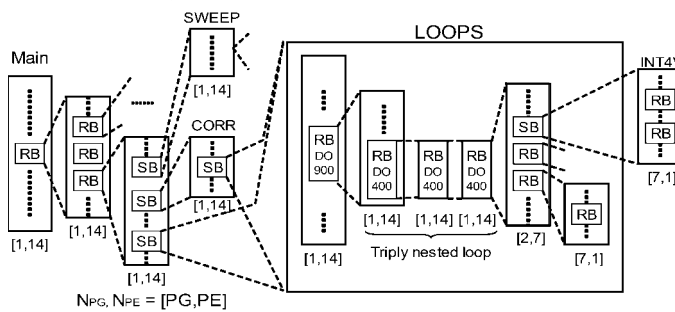


Hierarchical parallelism control scheme

- Estimate parallelism in each layer
 - Calculation of sequential cost of MTG and CP length
 - Considering coarse grain and loop level parallelism
- Assign required number of processors for each MTG considering parallelism of the layer
 - Assignment of processors from upper layer to lower layer
 - Selection of suitable parallel processing layer
 - Sequential execution of a layer having no parallelism

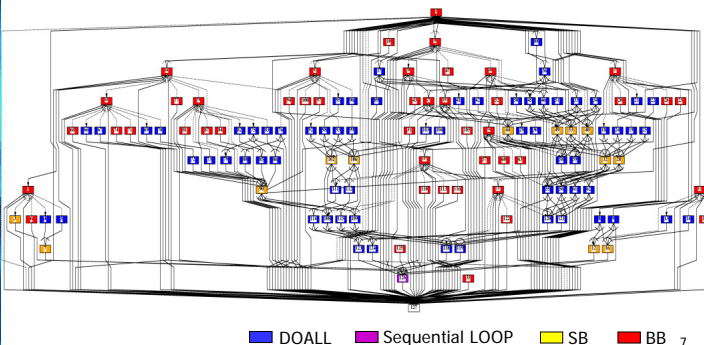
Automatic processor assignment of 103.su2cor

- Using 14 processors
 - Coarse grain parallelization within DO400 of subroutine LOOPS



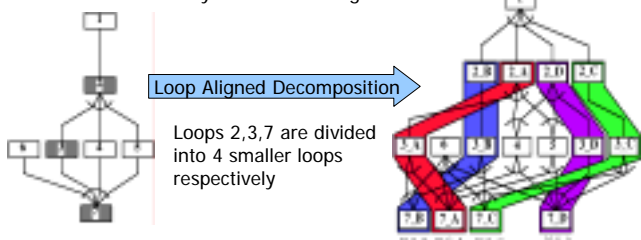
MTG of Su2cor-LOOPS-DO400

- Coarse grain parallelism PARA_ALD = 4.3



Cache optimization using Data Localization scheme

- Find Target Loop Group (TLG)
- Divide the loops considering cache size
 - Define Data Localizable Loop (DLG)
- Static and dynamic scheduling for data localization
 - Dt/gain static scheduling
 - Partial static dynamic scheduling



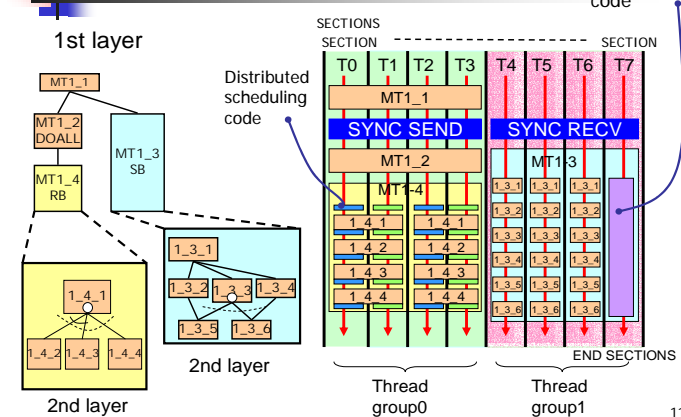
Code generation using OpenMP

- Compiler generates a parallelized program using OpenMP API
- One time single level thread generation for hierarchical multigrain parallelization
 - Threads are forked only once at the beginning of a program by OpenMP "PARALLEL SECTIONS" directive
 - Forked threads join only once at the end of program
- Compiler generates codes for each threads using static or dynamic scheduling schemes
- Hierarchical multigrain parallel processing is realized by ordinary OpenMP API.

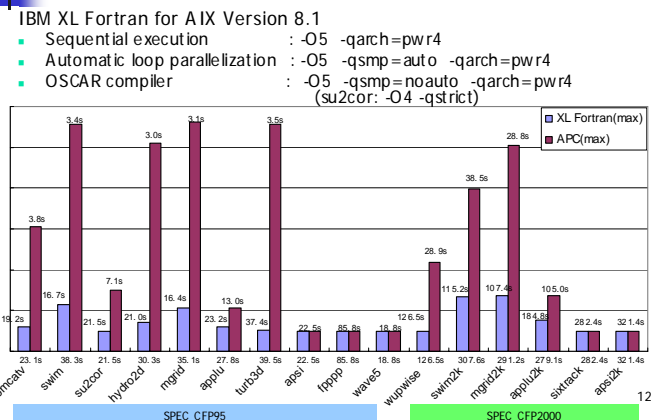
Scheduling of coarse grain tasks on MTG

- Static Scheduling
 - At compilation time, macro-tasks are assigned to processors statically.
 - Minimization of scheduling overhead and data transfer overhead
- Centralized/Distributed Dynamic Scheduling
 - Macro-tasks are assigned dynamically by the scheduling code exclusively generated for the MTG by compiler
 - Cope with runtime uncertainty like conditional branches
- A suitable scheduling scheme is chosen for each layer of Macro-Task Graph

Hierarchical code image

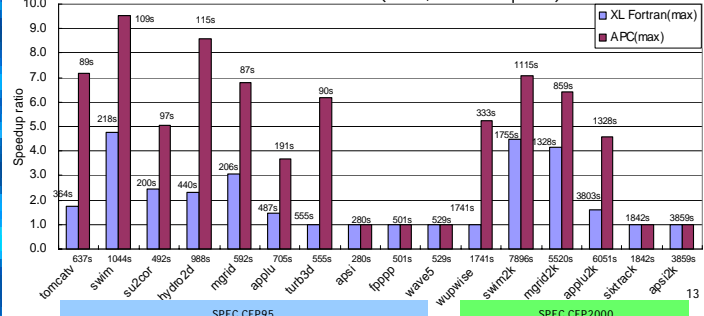


Performance of hierarchical multigrain parallelization on 16 processor High-end Server IBM pSeries690



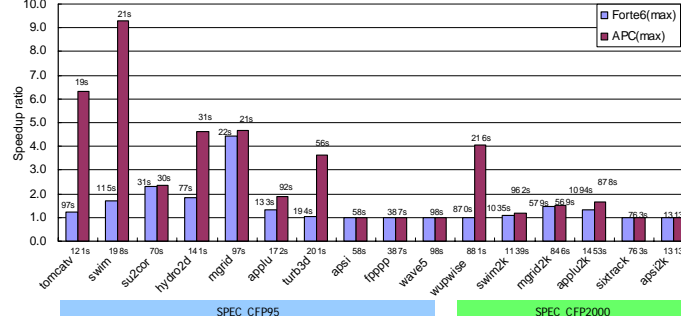
Performance of hierarchical multigrain parallelization on 8 processor Server IBM RS6000 SP 604e

- IBM XL Fortran for AIX Version 7.1
 - Sequential execution : -O5 -qarch=ppc
 - Automatic loop parallelization : -O5 -qsmp=auto -qarch=ppc
 - OSCAR compiler : -O3 -qsmp=noauto -qhot -qarch=ppc (swim, su2cor: -qstrict)



Performance of hierarchical multigrain parallelization on 4 processor workstation Sun Ultra80

- Sun Forte Developer 6 Update 2
 - Sequential execution : -fast
 - Automatic loop parallelization : -fast -autopar -reduction -stackvar
 - OSCAR compiler : -fast -explicitpar -mp=openmp -stackvar



Conclusions

- Automatic multigrain parallelizing compiler
 - Multigrain parallelism exploitation
 - Hierarchical parallelism control scheme
 - Cache optimization using data localization scheme
 - One-time single level thread generation
- Performance of hierarchical multigrain parallelization for Fortran77 16 programs of SPEC CFP95 and SPEC CFP2000
 - Hierarchical multigrain parallelization boosted up the performance of XL Fortran compiler 3.5 times in average on IBM Regatta, 2.4 times on IBM RS6000 and the performance of Forte 6 Update 2 compiler 2.0 times on Sun Ultra80.