Parallelization and Power Reduction Compiler for Heterogeneous Multicores for Emerging Applications

Hironori Kasahara
President Candidate in IEEE Computer Society Election 2016 (Aug. 1 – Sept. 26, 2016)
Voting: https://eballot4.votenet.com/IEEE/login.cfm
Professor, Dept. of Computer Science & Engineering
Director, Advanced Multicore Processor Research Institute
Waseda University, Tokyo, Japan
URL: http://www.kasahara.cs.waseda.ac.jp/

Waseda Univ. GCSC
Solar Powered
Smart phones
Cameras
Robots
Cool desktop servers
Emerging Applications
Industry-government-academia collaboration in R&D

For smart life
Multicore Engine ECU, ADAS (Driver Assistance), Self Driving (Deep Learning), HV, EV, FCV
Consumer electronic Internet TV/DVD
Camcorders Capsule inner cameras
Cameras
Operation/recharging by solar cells
Medical servers
Heavy particle radiation planning, cerebral infarction

OSCAR Multicore
Vector Acc.
Waseda University : R&D
Many-core system technologies with ultra-low power consumption

OSCAR Multicore Technology
OSCAR Many-core Comp
Compiler, API

Disaster Survival Supercomputer (Earthquakes, tsunami, Fire-spreading)

Solar Powered, Non–fan, cool, quiet servers
Supercomputers and servers

OSCAR

保护生命
保护环境
工业

智能家用电器
Engine Control by Multicores

Parallel processing of the engine control on multicore has been very difficult because of

- the hard real time control using local memory
- Millions lines of codes with conditional branches, basic blocks, functions and no loop.

The developed method can be applied both for hand-written codes and model based designed codes.

1.95 times speedup on 2core V850 multicore processor
MTG of Crankshaft Program Using Inline Expansion and Duplicating If-statements

MTG of crankshaft program before restructuring
- Succeed to reduce CP
  - 99% -> 60%

MTG of crankshaft program after restructuring
- Successfully increased coarse grain parallelism

CP accounts for over 99% of whole execution time.

CP accounts for about 60% of whole execution time.
Cancer Treatment
Carbon Ion Radiotherapy
(Previous best was 2.5 times speedup on 16 processors with hand optimization)

8.9 times speedup by 12 processors
Intel Xeon X5670 2.93GHz 12 core SMP (Hitachi HA8000)

55 times speedup by 64 processors
IBM Power 7 64 core SMP (Hitachi SR16000)
With 128 cores, OSCAR compiler gave us 100 times speedup against 1 core execution and 211 times speedup against 1 core using Sun (Oracle) Studio compiler.
OSCAR Parallelizing Compiler

To improve effective performance, cost-performance and software productivity and reduce power

Multigrain Parallelization
coarse-grain parallelism among loops and subroutines, near fine grain parallelism among statements in addition to loop parallelism

Data Localization
Automatic data management for distributed shared memory, cache and local memory

Data Transfer Overlapping
Data transfer overlapping using Data Transfer Controllers (DMAs)

Power Reduction
Reduction of consumed power by compiler control DVFS and Power gating with hardware supports.
OSCAR Heterogeneous Multicore

- DTU – Data Transfer Unit
- LPM – Local Program Memory
- LDM – Local Data Memory
- DSM – Distributed Shared Memory
- CSM – Centralized Shared Memory
- FVR – Frequency/Voltage Control Register
Hint for OSCAR Compiler to specify which part of the program can be executed on accelerators

```c
#pragma oscar_hint accelerator_task (ACCa) cycle(1000, ((OSCAR_DMAC())))
    for (i = 0; i < 10; i++) {
        x[i]++;
    }

#pragma oscar_hint accelerator_task (ACCb) cycle(100) in(var1, x[2:11]) out(x[2:11])
    call_FFT(var1, x);

void call_FFT(int var, int *x) {
    #pragma oscar_comment XXXXXXXXXXXX
    FFT(var, x);
}
```

Accelerator compiler or programmer specifies which parts of the programs can be executed on which accelerator
Multicore Program Development Using OSCAR API V2.0

Sequential Application Program in Fortran or C
(Consumer Electronics, Automobiles, Medical, Scientific computation, etc.)

 OSCAR API for Homogeneous and/or Heterogeneous Multicores and manycores
Directives for thread generation, memory, data transfer using DMA, power managements

Generation of parallel machine codes using sequential compilers

Waseda OSCAR Parallelizing Compiler
- Coarse grain task parallelization
- Data Localization
- DMAC data transfer
- Power reduction using DVFS, Clock/ Power gating

Hitachi, Renesas, NEC, Fujitsu, Toshiba, Denso, Olympus, Mitsubishi, Esol, Cats, Gaio, 3 univ.

Parallelized API F or C program
- Proc0
  Code with directives
  Thread 0
- Proc1
  Code with directives
  Thread 1
- Accelerator 1
  Code
- Accelerator 2
  Code

Low Power Homogeneous Multicore Code Generation
- API Analyzer
- Existing sequential compiler

Low Power Heterogeneous Multicore Code Generation
- API Analyzer
  (Available from Waseda)
- Existing sequential compiler

Server Code Generation
- OpenMP Compiler

Heterogeneous Multicores from Vendor B
Hitachi, Renesas, NEC, Fujitsu, Toshiba, Denso, Olympus, Mitsubishi, Esol, Cats, Gaio, 3 univ.

Executable on various multicores

Parallelized API F or C program
- Proc0
  Code with directives
  Thread 0
- Proc1
  Code with directives
  Thread 1
- Accelerator 1
  Code
- Accelerator 2
  Code

Hartego OSCAR Parallelizing Compiler
- Coarse grain task parallelization
- Data Localization
- DMAC data transfer
- Power reduction using DVFS, Clock/ Power gating

Hitachi, Renesas, NEC, Fujitsu, Toshiba, Denso, Olympus, Mitsubishi, Esol, Cats, Gaio, 3 univ.
An Image of Static Schedule for Heterogeneous Multi-core with Data Transfer Overlapping and Power Control

<table>
<thead>
<tr>
<th></th>
<th>CPU0</th>
<th></th>
<th>CPU1</th>
<th></th>
<th>CPU2</th>
<th></th>
<th>CPU3</th>
<th></th>
<th>DRP0</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CORE</td>
<td>DTU</td>
<td>CORE</td>
<td>DTU</td>
<td>CORE</td>
<td>DTU</td>
<td>CORE</td>
<td>DTU</td>
<td>CORE</td>
<td>DTU</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LOAD</td>
<td>LOAD</td>
<td>LOAD</td>
<td>LOAD</td>
<td></td>
<td></td>
<td>LOAD</td>
<td>LOAD</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SEND</td>
<td>SEND</td>
<td>SEND</td>
<td>SEND</td>
<td></td>
<td></td>
<td>SEND</td>
<td>SEND</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
<td></td>
<td></td>
<td>OFF</td>
<td>OFF</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>STORE</td>
<td>STORE</td>
<td>STORE</td>
<td></td>
<td></td>
<td>STORE</td>
<td>STORE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MTG1
- MT1-1: LOAD
- MT1-2: LOAD
- MT1-3: SEND
- MT1-4: SEND

MTG2
- MT2-1: LOAD
- MT2-2: OFF
- MT2-3: SEND
- MT2-4: SEND
- MT2-5: SEND
- MT2-6: SEND
- MT2-7: MT3-4
- MT2-8: MT3-7

MTG3
- MT3-1: LOAD
- MT3-2: LOAD
- MT3-3: LOAD
- MT3-4: SEND
- MT3-5: SEND
- MT3-6: SEND
- MT3-7: MT3-8
- MT3-8: OFF

TIME
33 Times Speedup Using OSCAR Compiler and OSCAR API on RP-X
(Optical Flow with a hand-tuned library)

Y. Yuyama, et al., "A 45nm 37.3GOPS/W Heterogeneous Multi-Core SoC", ISSCC2010
Power Reduction in a real-time execution controlled by OSCAR Compiler and OSCAR API on RP-X (Optical Flow with a hand-tuned library)

Without Power Reduction

Average: 1.76 [W]

With Power Reduction by OSCAR Compiler

70% of power reduction

Average: 0.54 [W]

1 cycle: 33 [ms] → 30 [fps]
OSCAR Vector Multicore and Compiler for Embedded to Servers with OSCAR Technology

Target:
- Solar Powered with compiler power reduction.
- Fully automatic parallelization and vectorization including local memory management and data transfer.

Fujitsu Vector Multiprocessor Supercomputer VPP700
Fujitsu VPP500/NWT: PE Unit
Summary

- OSCAR Automatic Parallelizing and Power Reducing Compiler has succeeded speedup and/or power reduction on homogeneous and heterogeneous multicores of emerging applications including “Automobile Engine Control”, “Earthquake Wave Propagation”, “Cancer Treatment Using Carbon Ion”, “Drinkable Inner Camera”, “Medical Image Processing”, Convolution for Deep Learning” and “Human Face Detection”

- In automatic parallelization, 33 times speedup for “Optical Flow” on RP-X heterogeneous multicore having 8 processor cores and 4 DRP (Dynamic Reconfigurable Processor) accelerator cores, 110 times speedup for “Earthquake Wave Propagation Simulation” on 128 cores of IBM Power 7 against 1 core, 55 times speedup for “Carbon Ion Radiotherapy Cancer Treatment” on 64 cores IBM Power 7, 1.95 times for “Automobile Engine Control” on Renesas 2 cores using SH4A or V850, 55 times for “JPEG-XR Encoding for Capsule Inner Cameras” on Tilera 64 cores Tile64 manycore.

- The compiler will be available on market from OSCAR Technology.

- In automatic power reduction, consumed powers for real-time multi-media applications like Optical flow were reduced to 1/3 on RP-X heterogeneous multicore having 8 processor cores and 4 DRP (Dynamic Reconfigurable Processor) accelerator cores, Human face detection, H.264, mpeg2 and optical flow were reduced to 1/2 or 1/3 using 3 cores of ARM Cortex A9 and Intel Haswell and 1/4 using Renesas SH4A 8 cores against ordinary single core execution.

- A super low power multicore processor using vector accelerator cores is being designed for Automobiles, Medical Systems, IoT, Disaster Survival Servers, etc.