A Study of Load Balanced Real-Time Sheduling on Asymmetric Multiprocessor

Tomohisa Ishikawa (510007)

School of Information Science,
Japan Advanced Institute of Science and Technology

February 8, 2007

Keywords: Function-Distributed Multiprocessor, Dynamic Load Balancing, Real-Time Scheduling, Embedded System.

1 Introduction

Recently, the use of multiprocessors has increased in embedded systems. Because, multiprocessors enable to increase performance and at the same time to decrease power consumption. In embedded systems, FDMP(Function-Distributed Multiprocessor [1], or it's called Asymmetric Multiprocessor) is often used.

However, it is difficult to achieve dyanmic load balancing in FDMP, since FDMP consists of different ISAs (Instruction Set Architecture). In this paper, I propose a dynamic load balancing on FDMP. It is a method that moves a task from a high load processor to a low load processor.

2 Real-Time System and FDMP

It is important for real-time systems to finish task execution by their deadline. I target soft real-time system where deadline misses are allowed to occur to some extent.

Applications are fixed in embedded systems. FDMP consists of special processors such as DSP and general processors. The special processor can

execute a part of the application fast. Therefore, FDMP is suitable for embedded systems. However, each task is statically assigned to a processor. When many tasks are invoked on a processor, deadline misses would occur.

3 Load-Balanced Scheduling on FDMP

FDMP consits of processors whose ISA is defferent from each other. A processor cannot execute tasks whose binary does not follow the ISA. To achive dynamic load balancing processors have to be able to execute all tasks in the system. Each processor has an executable binary which includes all tasks in the sysytem. Therefore, tasks becomes movable from a high load processor to a low load processor.

When a task moves from a high load processor to a low load processor, the task is stored in a shared memory. The shared memory in the FDMP has space that is called shared task pool. A processor communicates with other processors through the shared task pool.

The scheduler performs deadline miss prediction. According to the result, the scheduler performs the following:

1. When a task is predicted to miss the deadline, the processor moves the task to the shared task pool.
2. Except 1 and when there are few tasks in the ready queue, the processor picks up a task in the shared task pool.
3. Except 1 and 2, the processor performs normal scheduling.

4 Evaluation

I implemented an FDMP simulator that consits of MIPS32-ISA simulator [2] and TMS320C54xDSP-ISA simulator [3]. Simulator inputs are binary files of MIPS32-ISA and TMS320C54xDSP-ISA. The source files of taskset and scheduler are compiled.

The simulater executes the input binaries and outputs the number of deadline misses. As a result, deadline misses are decreased when MIPS load and DSP load are not balanced.

5 Conclusion

In this thesis, I proposed a real-time scheduling with dynamic load balancing on FDMP. The scheduler moves a task from a high load processor to a low load processor when deadline misses are predicted. The evaluation by simulation revealed that the proposed method decreased deadline misses when processor loads were not balanced.

References

[1] "A Real-Time OS for Function-Distributed Multiprocessor", Hiroaki Takada, Shinya Honda, Information Processing Vol.47 No.1 2006

[2] "MIPS32 Architecture For Programmers" Revision 2.50, MIPS Technologies Inc.

[3] "TMS320C54x DSP Reference Set", Texas Instruments Inc.