digiteo
Research in science and technology of information
www.digiteo.fr

Abir Benabid and Claire Hanen

Poster n°2008-13

ORCYAE: Cyclic Scheduling for Embeded Architectures

Objectives

The overall objective is to improve cyclic scheduling for modern VLIW processors compilers.

1. Define the model to be used for cyclic instruction scheduling.
2. Two approaches:
   - approximation algorithms and efficiency bounds
   - exact methods
3. Computational experiments on industrial applications

1 Motivation

Embedded architectures used for devices such as mobile, automotive and consumer electronics need:
- high performance
- low silicon implementation costs
- low power consumption
- rapid development to ensure minimum time-to-market

⇒ Instruction level parallel processors such as VLIW processors.
★ Instruction scheduling is a performance critical optimization which takes place when compiling applications.
★ High-quality instruction schedules enable to reduce the operating frequency given real-time processing requirements.

2 Problem formulation

- Architecture: k types of resources $\{M_i\}_{1\leq i\leq k}$ with limited availabilities $\{m_i\}_{1\leq i\leq k}$.
- Precedence graph:
  - n tasks $\{T_i\}_{1\leq i\leq n}$ with unit processing time and unit resource request.
  - Precedence relations: uniform dependences, denoted $T_i \overset{l_{ij},h_{ij}}{\longrightarrow} T_j$, where the latency l_{ij} and the height h_{ij} are positive integers.

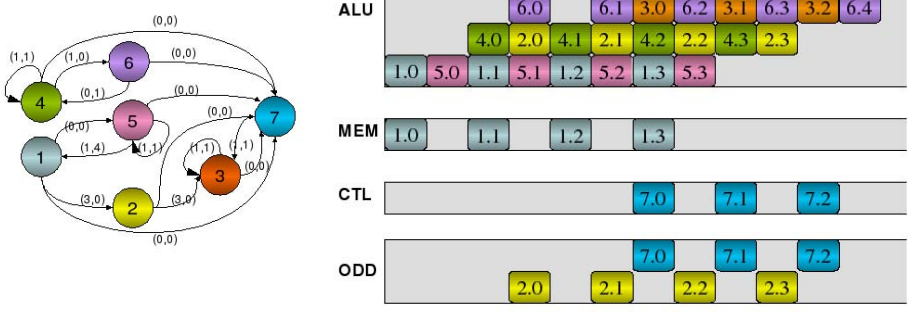
$$\left(T_i \overset{l_{ij},h_{ij}}{\longrightarrow} T_j\right) \Rightarrow \sigma(T_i,q)+l_{ij} \leq \sigma(T_j,q+h_{ij}) \ \ \forall q \in \mathbb{N}$$

Find 1-periodic cyclic schedule with the minimal period λ:
$\forall i \in \{1,\cdots,n\}, \forall q \in \mathbb{N}: \ \ \sigma(T_i,q)=\sigma(T_i,0)+q\cdot\lambda$

2.1 Example

```
int
prod(int n, short a[], short b[]){
   int s=0, i;
   for (i=0; i;n; i++){
      s+=a[i]*b;
   }
   return s;
}
```

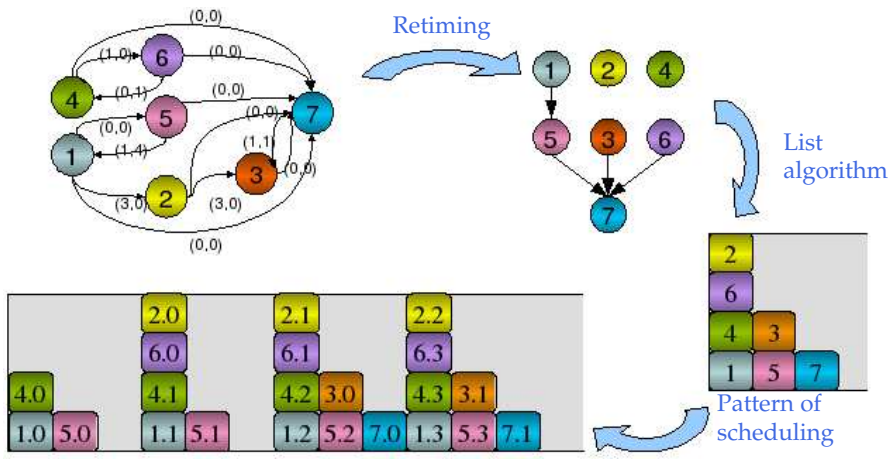
prod:
LDH A = 0, B
MUL C = D, A
ADD E = E, C
ADD F = F, 1
ADD G = G, 2
CMPNE H = I, F
BRF H, prod



2.2 State of the art

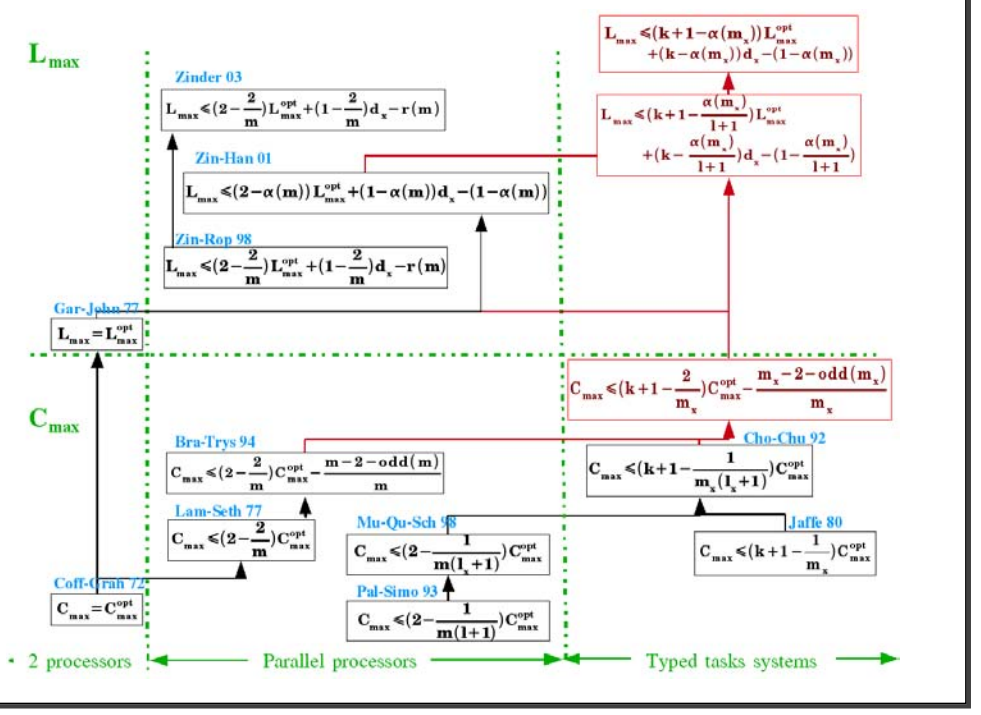
- Generating an optimal resource constrained cyclic scheduling with minimal period is known to be $\mathcal{NP}$-hard.
- The classic results of cyclic scheduling apply to problems that are too limited to be of practical use in instruction scheduling. These results assume:
  - simple precedence constraints on the order of tasks in a schedule, instead of precedences with delays like those in pipelined processors
  - machine models where each operation uses one of m identical processors for its execution

3 Decomposed Software Pipelining



$$\Sigma^k P|p_i=1, cyclic, l_{ij}|\lambda$$
$$\lambda^{\sigma} \leq \left(k+1-\frac{1}{m_x(\delta+1)}\right)\lambda^{opt}+\delta+1-\frac{1}{m_x}$$

4 Performance guarantees for non-cyclic problems



5 Perspectives

- Find a retiming that minimizes the makespan of the pattern (the period of the cyclic schedule).
- Experiments on STMicroelectronics applications: ST200 VLIW processor compilers.
- Branch and Bound based on Decomposed Software Pipelining.

Contacts : Abir.Benabid@lip6.fr · Claire.Hanen@lip6.fr

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE
LIP6
îledeFrance