Lecture 5: Linear Loop Tiling

CR11 – Hardware Compilation and Simulation
ENS-Lyon – M2iF

Christophe Alias

Motivation

for i := 0 to N
for j := 0 to N
B: a[i,j] := a[i-1,j] + a[i,j-1];

- Tune the operational intensity
- Software compilation: cache → tiling
- Hardware compilation: PP/BW → OI → tiling → cache

Parallellepipedic Loop Tiling

- Slice with hyperplanes $\vec{\tau}_i$ by $b_i$
- Tiled domain: $(T_1, T_2, i, j) \in \hat{D}_B$

Loop Tiling

- Partition into atomic blocks (tiles)
- Scope: perfect loop nests with uniform dependences
Quizz

Draw the tile execution order with \( \theta(T_1, T_2, i, j) = (T_2, T_1, i, j) \)

Computing \( \hat{D}_B \)

Find a relation between \( T_k, \vec{\tau}_k \cdot \vec{x} \) and \( b_k \)

Quizz

Generate the loop tiling for \( \theta(T_1, T_2, i, j) = (T_2, T_1, i, j) \)

Correctness

The tile dependence graph is schedulable
**Quizz**

```
for i := 1 to N
for j := 1 to N
B: a[i] := a[i-1] + a[i] + a[i+1];
```

Is the tiling valid?

**Correctness, II**

```
for i := 1 to N
for j := 1 to N
B: a[i] := a[i-1] + a[i] + a[i+1];
```

\(\vec{\tau}\) is valid if \(\vec{\tau} \cdot \vec{d} \geq 0\) \(\forall \vec{d} \in \Delta D\)

**Quizz**

```
for i := 0 to N
for j := 0 to N
B: a[i] := a[i-1] + a[i] + a[i+1];
```

Find a valid tiling

**Communication Minimal Tiling**

```
for i := 0 to N
for j := 0 to N
B: a[i] := a[i] + 1;
```

Tend to 0 communication, minimize \(\vec{\tau} \cdot \vec{d} \) \(\forall \vec{d} \in \Delta D\)
Algorithm

\[
\begin{align*}
\text{for } i & := 0 \text{ to } N \\
\text{for } j & := 0 \text{ to } N \\
B: & \quad a[i] := a[i] + 1;
\end{align*}
\]

\[
\begin{align*}
\text{for } i & := 1 \text{ to } \dim D_B \\
\text{Solve the ILP:} & \\
\min & \quad M \\
\text{s.t.} & \quad \vec{\tau}_i \in C_T \setminus \{\vec{0}\} \\
& \quad (\vec{\tau}_1, \ldots, \vec{\tau}_i) \text{ linearly independent} \\
& \quad \vec{\tau}_i \cdot \vec{d} \leq M \quad \forall \vec{d} \in \Delta D
\end{align*}
\]

Quizz

\[
\begin{align*}
\text{for } i & := 0 \text{ to } N \\
\text{for } j & := 0 \text{ to } N \\
B: & \quad a[i] := a[i] + 1;
\end{align*}
\]

Apply the algorithm

Lab

\[
\begin{align*}
\text{for } t & := 1 \text{ to } N \\
\text{for } i & := 1 \text{ to } N \\
B: & \quad a[t,i] := a[t-1,i-1] + a[t-1,i] + a[t-1,i+1];
\end{align*}
\]

1. Find a communication-minimal tiling
2. Generate the code with \texttt{iscc}