Iteration Bound

Lan-Da Van (范倫達), Ph. D.
Department of Computer Science
National Chiao Tung University
Taiwan, R.O.C.
Fall, 2015

ldvan@cs.nctu.edu.tw
http://www.cs.nctu.tw/~ldvan/
Outline

- Introduction
- Data Flow Graph (DFG) Representations
- Loop Bound and Iteration Bound
- Compute the Iteration Bound
  - Longest Path Matrix Algorithm (LPM)
  - Minimum Cycle Mean Method (MCM)
- Conclusion
Introduction

Many DSP algorithms such as recursive and adaptive digital filters contain feedback loops, which impose an inherent fundamental lower bound on the achievable iteration or sample period.

- Iteration bound = maximum loop bound
- Clock period = critical path period = cycle time = 1/clock rate
- Sample rate = throughput rate
- Impossible to achieve an iteration bound less than the theoretical iteration bound with infinite processors
Outline

- Introduction
- Data Flow Graph (DFG) Representations
- Loop Bound and Iteration Bound
- Compute the Iteration Bound
  - Longest Path Matrix Algorithm (LPM)
  - Minimum Cycle Mean Method (MCM)
- Conclusion
DSP Program: Example

\[ y(n) = ay(n-1) + x(n) \]

\[
\text{for } n = 0 \text{ to } \infty
\]

Intra - Iteration Period: \( A_k \rightarrow B_k \)

Inter - Iteration Period: \( B_k \Rightarrow A_{k+1} \)

Data Flow Graph
Loop Bounds in DFG

- **Critical Path**: The path with the longest computation time among all paths that contain zero delays.
- **Loop**: A directed path that begins and ends at the same node.
- **Loop Bound**: $t_l/w_l$, where $t_l$ is the loop computation time and $w_l$ is the number of delays in the loop.
- **Critical Loop**: The loops in which has maximum loop bound.
- **Iteration Bound**: Maximum loop bound, i.e., a fundamental limit for recursive algorithms.

Loop bounds: $4/2 \text{ u.t. (max)}, \ 5/3 \text{ u.t.}, \ 5/4 \text{ u.t.}$

---

Lan-Da Van

VLSI-DSP-2-6
Iteration bound

Definition: \[ T_\infty = \max_{l \in L} \left\{ \frac{t_l}{w_l} \right\} \]

(a) Iteration bound = \( \frac{6}{2} = 3 \) u.t.

(b) Iteration bound = \( \max \{ \frac{6}{2}, \frac{11}{1} \} \) = 11 u.t.
Outline

- **Introduction**
- **Data Flow Graph (DFG) Representations**
- **Loop Bound and Iteration Bound**
  - *Compute the Iteration Bound*
    - Longest Path Matrix Algorithm (LPM)
    - Minimum Cycle Mean Method (MCM)
- **Conclusion**
Longest Path Matrix (LPM) Algorithm

- A series of matrix is constructed, and the iteration bound is found by examining the diagonal elements of the matrices.
- \( d \) : # of delays
- Compute \( L^{(1)} \sim L^{(m)} \), \( m=1, 2, 3, \ldots, d \)
- \( l_{i,j}^{(m)} \) : The longest computation time of all paths from delay element \( d_i \) to delay element \( d_j \) that pass through exactly \( m-1 \) delays, where the delay \( d_i \) and delay \( d_j \) are not included for \( m-1 \) delays.

- If no path exists, then the value of \( l_{i,j}^{(m)} \) equals -1.
A DFG with Three Loops Using LPM

$$L^{(1)} = \begin{bmatrix}
-1 & 0 & -1 & -1 \\
4 & -1 & 0 & -1 \\
5 & -1 & -1 & 0 \\
5 & -1 & -1 & -1 \\
\end{bmatrix}$$
A DFG with Three Loops Using LPM (2/3)

\[ l_{i,j}^{(m+1)} = \max_{k \in K} (-1, l_{i,k}^{(1)} + l_{k,j}^{(m)}) \]

\[
L^{(2)} = \begin{bmatrix}
-1 & 0 & -1 & -1 \\
4 & -1 & 0 & -1 \\
5 & -1 & -1 & 0 \\
5 & -1 & -1 & -1 \\
\end{bmatrix}
\times
\begin{bmatrix}
-1 & 0 & -1 & -1 \\
4 & -1 & 0 & -1 \\
5 & -1 & -1 & 0 \\
5 & -1 & -1 & -1 \\
\end{bmatrix}
\]

\[
L^{(2)} = \begin{bmatrix}
4 & -1 & 0 & -1 \\
5 & 4 & -1 & 0 \\
5 & 5 & -1 & -1 \\
-1 & 5 & -1 & -1 \\
\end{bmatrix}
\]
A DFG with Three Loops Using LPM (3/3)

\[
\mathbf{L}^{(3)} = \begin{bmatrix}
5 & 4 & -1 & 0 \\
8 & 5 & 4 & -1 \\
9 & 5 & 5 & -1 \\
9 & -1 & 5 & -1 \\
\end{bmatrix}
\]

\[
\mathbf{L}^{(4)} = \begin{bmatrix}
8 & 5 & 4 & -1 \\
9 & 8 & 5 & 4 \\
10 & 9 & 5 & 5 \\
10 & 9 & -1 & 5 \\
\end{bmatrix}
\]

\[
T_\infty = \max_{i,m \in \{1, 2, \ldots, d\}} \left\{ \frac{l_{i,i}^{(m)}}{m} \right\}
\]

\[
T_\infty = \max_{i,m \in \{1, 2, \ldots, d\}} \left\{ \frac{4}{2}, \frac{4}{2}, \frac{5}{3}, \frac{5}{3}, \frac{5}{3}, \frac{8}{4}, \frac{8}{4}, \frac{5}{4}, \frac{5}{4} \right\} = 2
\]
A Filter Using LPM

\[
L^{(1)} = \begin{bmatrix} 4 & 4 \\ 8 & 8 \end{bmatrix}, \quad L^{(2)} = \begin{bmatrix} 12 & 12 \\ 16 & 16 \end{bmatrix}
\]

\[
T_\infty = \max\{\frac{4}{1}, \frac{8}{1}, \frac{12}{2}, \frac{16}{2}\} = 8
\]
Minimum Cycle Mean (MCM) Method

1. Construct the new graph $G_d$ and $\overline{G_d}$
   - Transform from original DFG $G$
   - Decide the # of nodes from the # of delays in $G$.
   - Decide the weight of each edge

2. Compute the minimum cycle mean
   - Construct the series of $d+1$ vectors $f^{(m)}$, $m=0, 1, 2, \ldots, d$
     - An arbitrary reference node is chosen in $\overline{G_d}$ (called this node $s$). The initial vector $f^{(0)}$ is formed by setting $f^{(0)}(s)=0$ and setting the remaining nodes of $f^{(0)}$ to infinity.
   - find the min cycle mean
A DFG with Three Loops Using MCM

(1/4)

- Cycle mean = Average length of the edge in c (Cycle = Loop)
- Longest path length
  - path that passes through no delays
  - longest: two loops that contain $D_a$ and $D_b$
    - max { 6, 4 } = 6
  - cycle mean = 6/2 = 3

Diagram:

(a) A Directed Acyclic Graph (DAG) with three loops:

- Node a connected to b, d, and c
- Node b connected to d
- Node d connected to c
- Node c connected to a

(b) A cycle within the graph:

- Cycle $\alpha$ connecting nodes a and b
- Cycle $\beta$ connecting nodes b and c

Each edge in the graph is labeled with a number indicating the delay, with $D_\alpha$ and $D_\beta$ indicating specific delays.
A DFG with Three Loops Using MCM (2/4)

- delay => node
- longest path length (computation time) => weight \( w(i,j) \)
  - If no zero-delay path exists from delay \( d_i \) to delay \( d_j \), then the edge \( i \rightarrow j \) does not exist in \( G_d \).
we will find $d+1$ vectors, $f^{(m)}$ $m=0,1,...,d$

$$f^{(0)} = \begin{bmatrix} 0 \\ \infty \\ \infty \\ \infty \end{bmatrix} \quad f^{(1)} = \begin{bmatrix} \infty \\ 0 \\ \infty \\ \infty \end{bmatrix} \quad f^{(2)} = \begin{bmatrix} -4 \\ \infty \\ 0 \\ \infty \end{bmatrix} \quad f^{(3)} = \begin{bmatrix} -5 \\ -4 \\ \infty \\ 0 \end{bmatrix} \quad f^{(4)} = \begin{bmatrix} -8 \\ -5 \\ -4 \\ \infty \end{bmatrix}$$

$$f^{(m)}(j) = \min_{i \in I} (f^{(m-1)}(i) + \overline{w}(i,j))$$
A DFG with Three Loops Using MCM

\( T_\infty = - \min_{i \in \{1, 2, \ldots, d\}} \left( \max_{m \in \{0, 1, 2, \ldots, d-1\}} \left( \frac{f^{(d)}(i) - f^{(m)}(i)}{d - m} \right) \right) \)

\[
\begin{array}{|c|c|c|c|c|}
\hline
m = 0 & m = 1 & m = 2 & m = 3 & \max_{0 \leq m \leq 3} \left\{ \frac{f^{(4)}(i) - f^{(m)}(i)}{4 - m} \right\} \\
\hline
i = 1 & -2 & -\infty & -2 & -3 & -2 \\
\hline
i = 2 & -\infty & -5/3 & -\infty & -1 & -1 \\
\hline
i = 3 & -\infty & -\infty & -2 & -\infty & -2 \\
\hline
i = 4 & \infty & -\infty & \infty & \infty & \infty \\
\hline
\end{array}
\]

\( T_\infty = -\min\{-2, -1, -2, \infty\} = 2 \)
A Filter Using MCM

$$f^{(0)} = \begin{bmatrix} 0 \\ \infty \end{bmatrix}$$

$$f^{(1)} = \begin{bmatrix} -4 \\ -4 \end{bmatrix}$$

$$f^{(2)} = \begin{bmatrix} -12 \\ -12 \end{bmatrix}$$

$$T_{\infty} = -\min\{-8, -6\} = 8$$
Conclusion

- When the DFG is recursive, the iteration bound is the fundamental limit on the minimum sample period of a hardware implementation of the DSP program.

- Two algorithms to compute iteration bound, LPM and MCM, were explored.