Ch.8. Iteration Bound

Spring, 2014

Wang Wei
wangwei@cqupt.edu.cn

College of Electronics Engineering, CQUPT

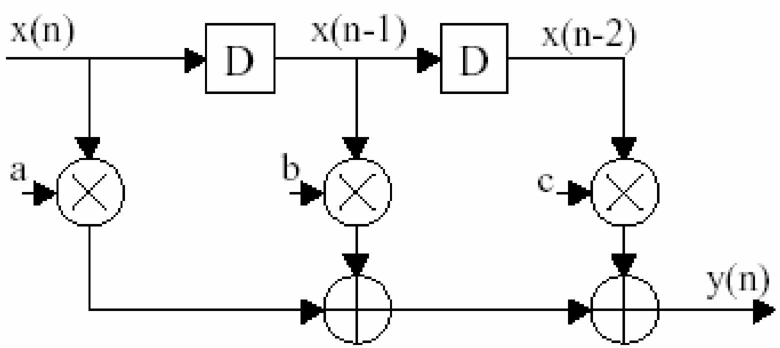
Outlook

- Introduction
- Data Flow Graph (DFG) Representation
- Loop Bound and Iteration Bound
- Compute the Iteration Bound
- Conclusion

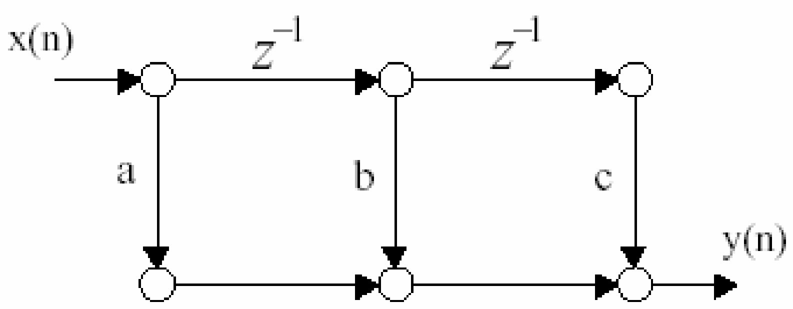
Representation Methods of DSP systems

Example: $y(n)=a*x(n)+b*x(n-1)+c*x(n-2)$

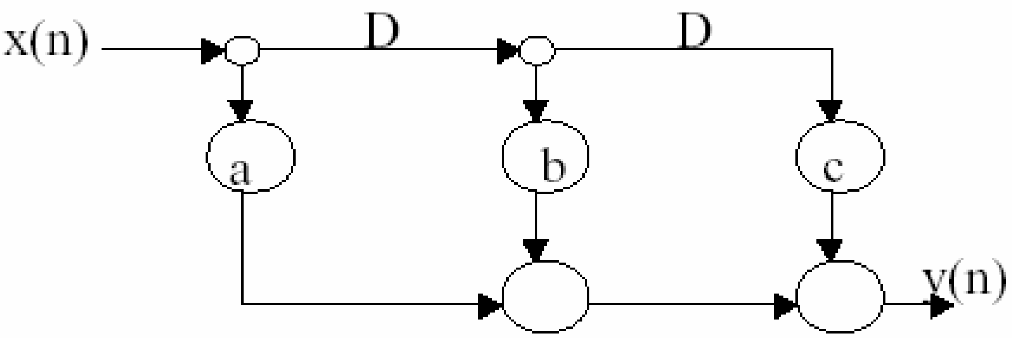
- Graphical Representation Method 1: Block Diagram
  - Consists of functional blocks connected with directed edges, which represent data flow from its input block to its output block



- Graphical Representation Method 2: Signal-Flow Graph
  - SFG: a collection of nodes and directed edges
  - Nodes: represent computations and/or task, sum all incoming signals
  - Directed edge (j, k): denotes a linear transformation from the input signal at node j to the output signal at node k
  - Linear SFGs can be transformed into different forms without changing the system functions. For example, ***Flow graph reversal*** or ***transposition*** is one of these transformations (Note: only applicable to single-input-single-output systems)
  - Usually used for linear time-invariant DSP systems representation



- Graphical Representation Method 3: Data-Flow Graph
  - DFG: nodes represent computations (or functions or subtasks), while the directed edges represent data paths (data communications between nodes), each edge has a nonnegative number of delays associated with it.
  - DFG captures the data-driven property of DSP algorithm: any node can perform its computation whenever all its input data are available.
  - Each edge describes a precedence constraint between two nodes in DFG:
    - Intra-iteration precedence constraint: if the edge has zero delays
    - Inter-iteration precedence constraint: if the edge has one or more delays
    - DFGs and Block Diagrams can be used to describe both linear single-rate and nonlinear **multi-rate** DSP systems
    - Fine-Grain DFG



Introduction

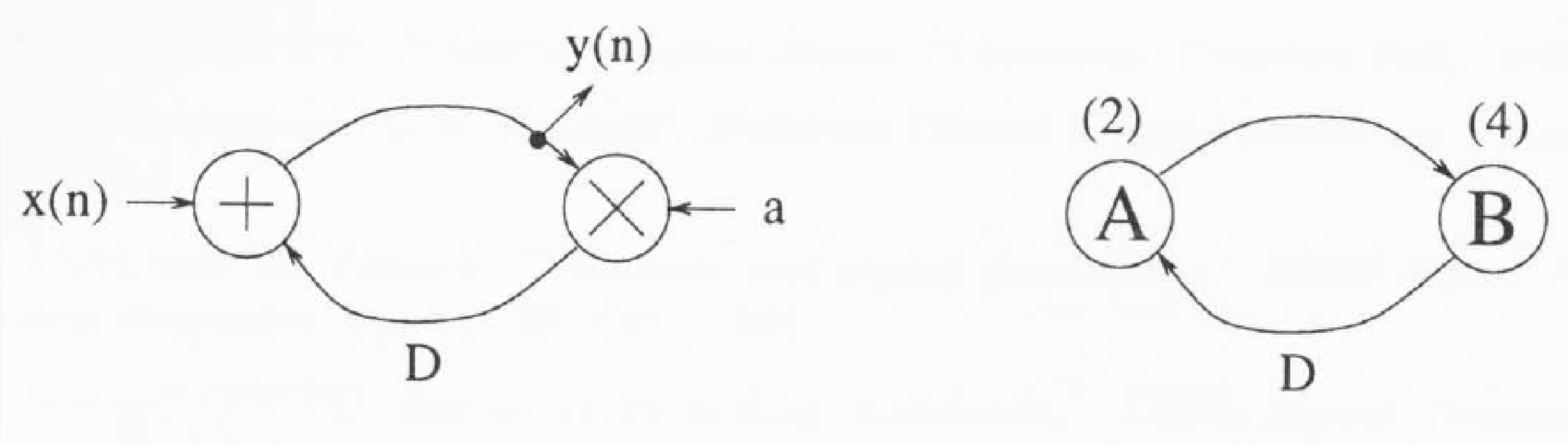
- Iteration Bound=maximum loop bound OR maximum non-loop bound
- Clock period=critical path period=cycle time =1/clock rate
- Sample rate= the number of samples processed in the DSP system per second (throughput rate)
- Impossible to achieve an iteration bound less than the theoratical iteration bound with infinite processors

DSP Program:Example

$$for \quad n=0 \quad to \quad \infty$$

$$y(n) = ay(n-1) + x(n)$$

DFG



(a) (b)

Each edge of DFG defines a precedence constraint

Precedence Constraints:

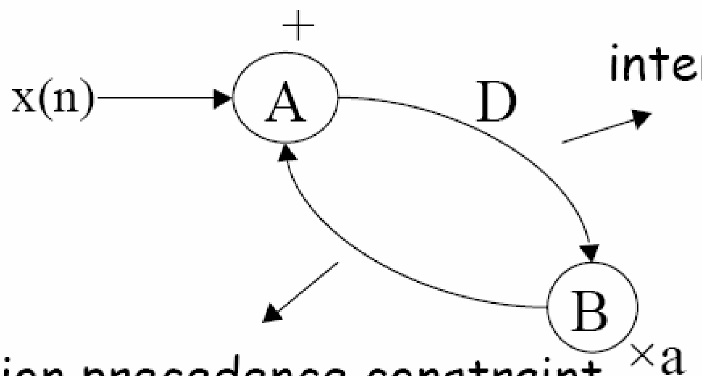
– Intra-iteration ---- edges with no delay elements

– Inter-iteration ---- edges with non-zero delay elements

Intra Iteration Period: $A_k \rightarrow B_k$

Inter Iteration Period: $B_k \Rightarrow A_{k+1}$

$y(n)=ay(n-1) + x(n)$



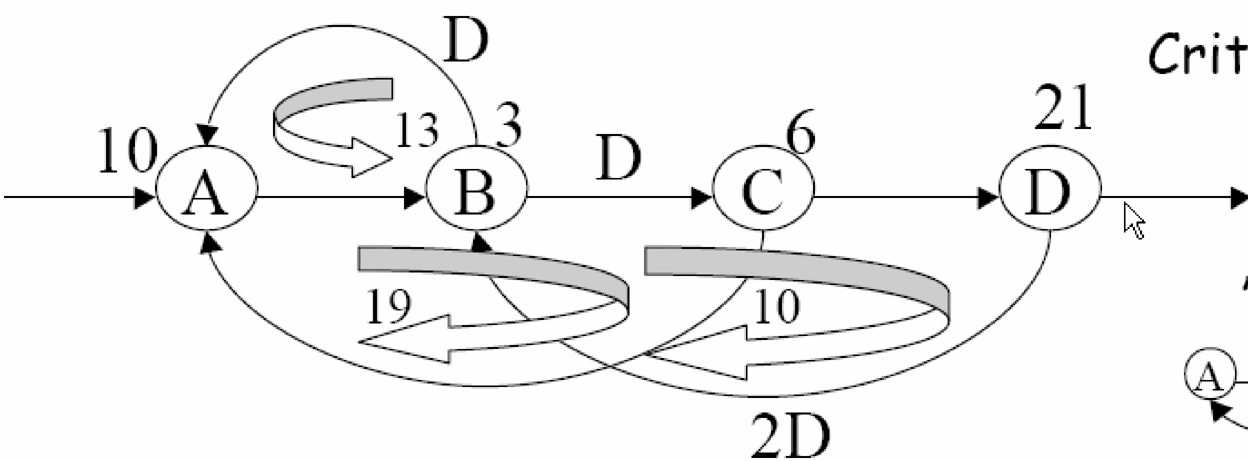
inter-iteration precedence constraint

$A_1 \rightarrow B_2$

$A_2 \rightarrow B_3$

$B_1 \rightarrow A_1 \Rightarrow B_2 \rightarrow A_2 \Rightarrow B_3 \rightarrow A_3 \Rightarrow \ldots\ldots$

intra-iteration precedence constraint



Critical Path = 27ut

$T_{clk} \geq 27ut$

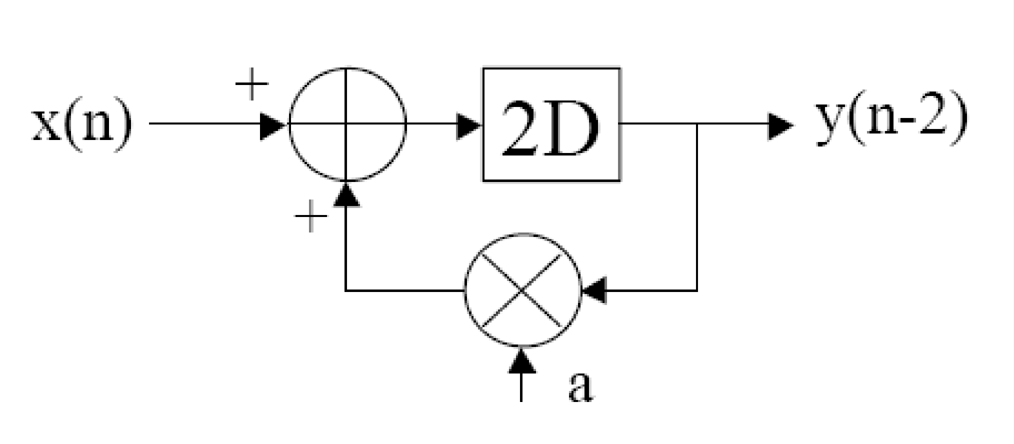
APG of this graph is

Definitions:

Loop: a directed path that begins and ends at the same node;

Loop bound of the j-th loop: defined as Tj/Wj, where Tj is the loop computation time & Wj is the number of delays in the loop

y(n) = a*y(n-2) + x(n), we have:



$$T_{loopbound} = \frac{T_m + T_a}{2} = 5ns$$

Loop Bounds in DFG

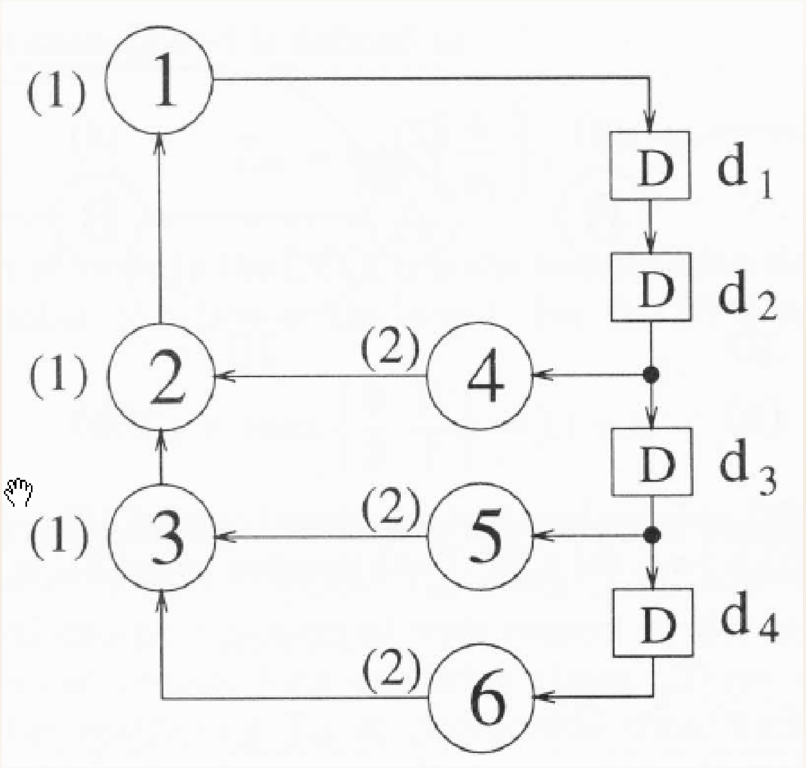
Critical Path: The path with the longest computation time among all paths that contain zero delays

Loop: 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.

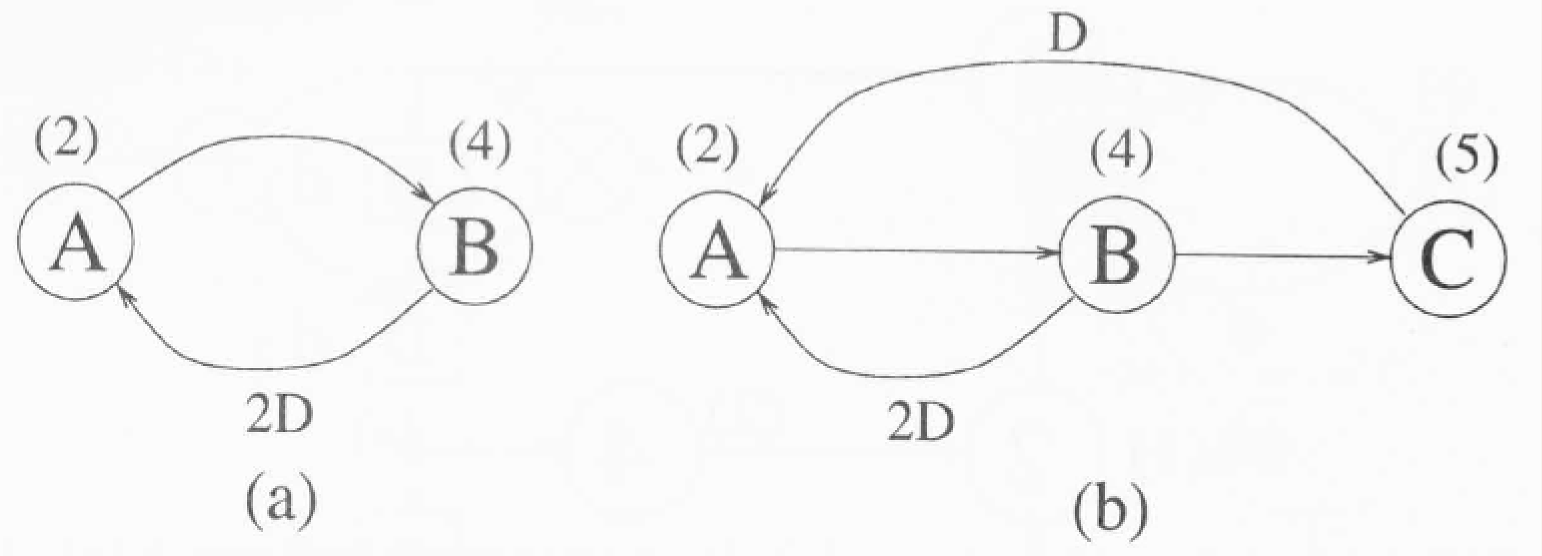
Iteration Bound: maximum loop bound / non-loop bound, i.e., a fundamental limit for recursive / non-recursive algorithms



Loop bounds:4/2 u.t. (max), 5/3 u.t. , 5/4 u.t.

Iteration Bound

Definition: $$T_\infty = \max_{l \in L}\{\frac{t_l}{w_l}\}$$



(a) Iteration bound= 6/2 = 3 u.t.

(b) Iteration bound= Max { 6/2 , 11/1 } = 11 u.t.

迭代边界的计算

最长路径矩阵算法（longest path matrix, LPM）

最小环均值算法 (minimum cycle mean, MCM)

负周期检测算法 (negtive cycle detection)

longest path matrix, LPM

A series of matrix is constructed, and the iteration bound is found by examining the diagonal elements of the matrices.

d : # of delays

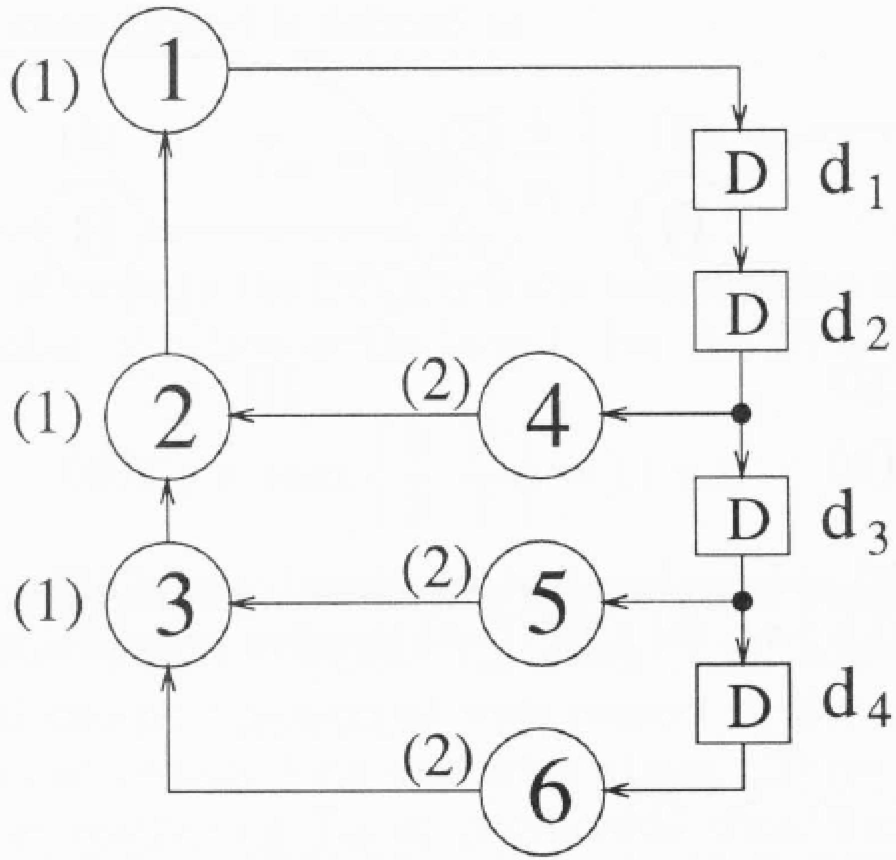
Compute L1~Lm, m=1, 2, 3, …, d

$l_{i,j}^{(m)}$: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* equals -1.

A DFG with Three loops using LPM(1/3)

$$\mathbf{L}^{(1)} = \begin{bmatrix} -1 & 0 & -1 & -1 \\ 4 & -1 & 0 & -1 \\ 5 & -1 & -1 & 0 \\ 5 & -1 & -1 & -1 \end{bmatrix}$$

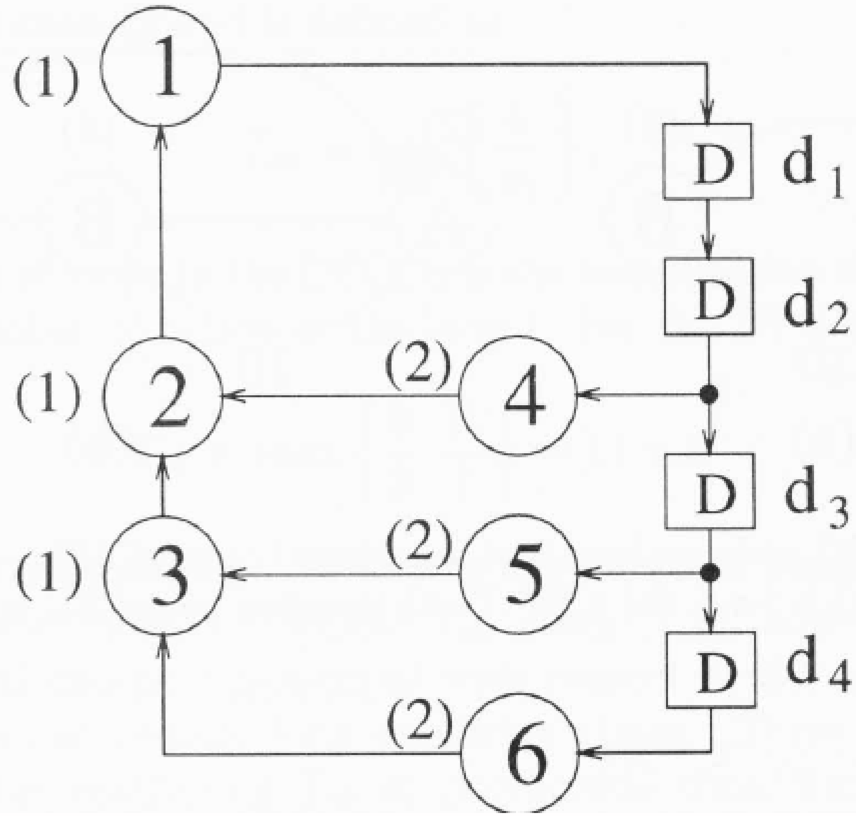


A DFG with Three loops using LPM(2/3)

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

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

$$\mathbf{L}^{(2)} = \begin{bmatrix} 4 & -1 & 0 & -1 \\ 5 & 4 & -1 & 0 \\ 5 & 5 & -1 & -1 \\ -1 & 5 & -1 & -1 \end{bmatrix}$$



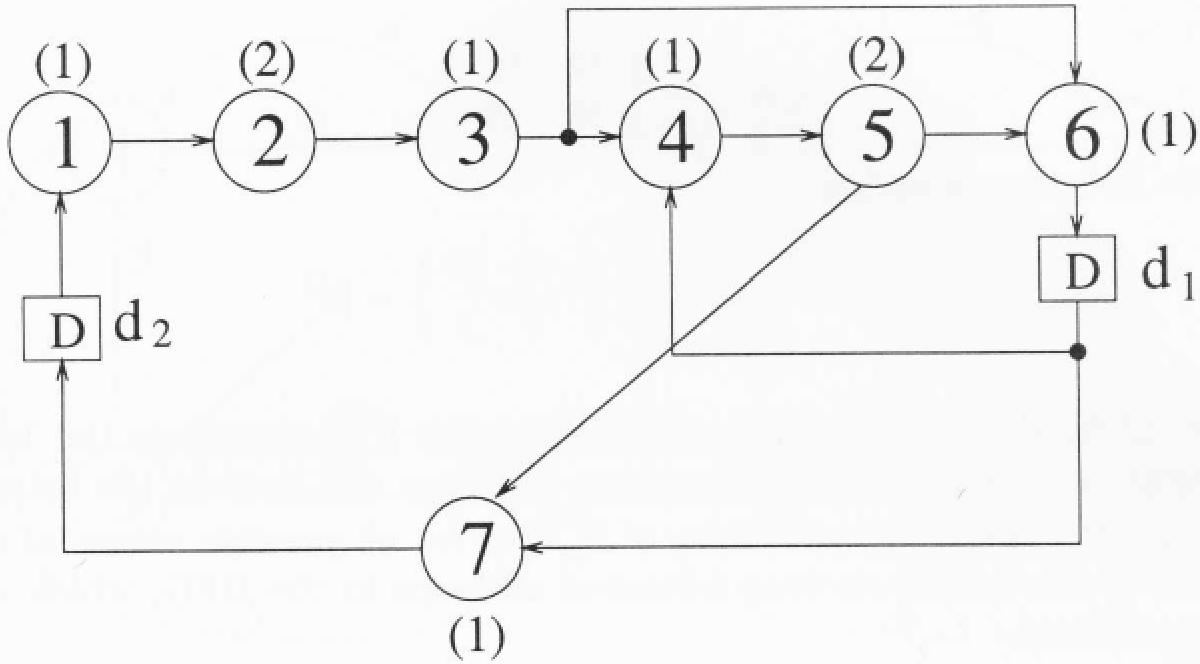
A DFG with Three loops using LPM(3/3)

$$\mathbf{L}^{(3)} = \begin{bmatrix} 5 & 4 & -1 & 0 \\ 8 & 5 & 4 & -1 \\ 9 & 5 & 5 & -1 \\ 9 & -1 & 5 & -1 \end{bmatrix} \qquad \mathbf{L}^{(4)} = \begin{bmatrix} 8 & 5 & 4 & -1 \\ 9 & 8 & 5 & 4 \\ 10 & 9 & 5 & 5 \\ 10 & 9 & -1 & 5 \end{bmatrix}$$

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

$$T_\infty = \max_{i,m\in\{1,2,\ldots,d\}} \left\{ \frac{4}{2}, \frac{4}{2}, \frac{5}{3}, \frac{5}{3}, \frac{5}{3}, \frac{8}{4}, \frac{8}{4}, \frac{5}{4}, \frac{5}{4} \right\} = 2$$

A Filter Using LPM



$$\mathbf{L}^{(1)} = \begin{bmatrix} 4 & 4 \\ 8 & 8 \end{bmatrix} \qquad \mathbf{L}^{(2)} = \begin{bmatrix} 12 & 12 \\ 16 & 16 \end{bmatrix}$$

$$T_{\infty} = \max\{\frac{4}{1}, \frac{8}{1}, \frac{12}{2}, \frac{16}{2}\} = 8$$

MCM

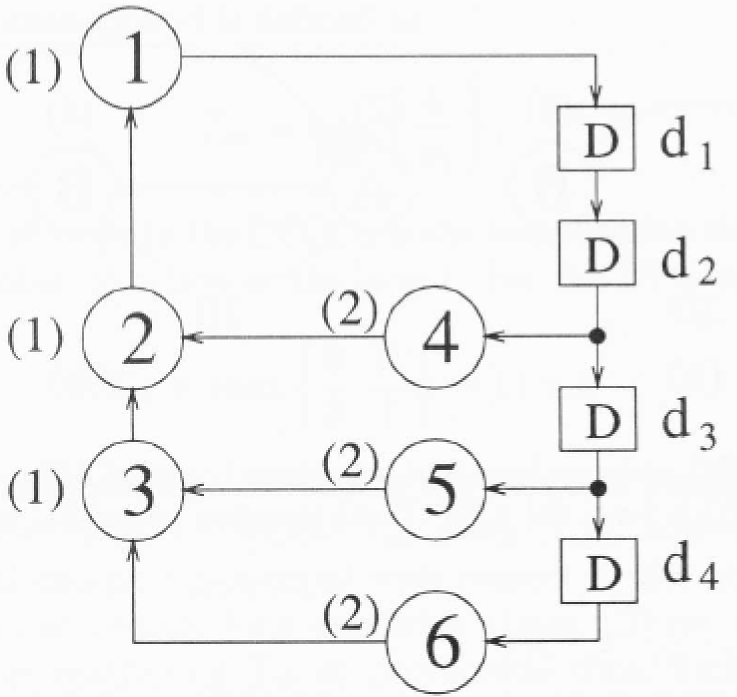
1. Construct the new graph G_d (DFG G)
   - transform from DFG
   - decide the weight of each edge
2. Compute the maximum cycle mean
   - construct the series of d+1 vectors $f^{(m)}$, *m=0, 1, 2, …,d*
     - *An arbitrary reference node is chosen in Gd (called this node s). The initial vector* **f**$^{(0)}$ is formed by setting $f^{(0)}(s)=0$ and setting the remaining nodes of **f**$^{(0)}$ to infinity.
   - find the max cycle mean
3. Find the min cycle mean between each cycle

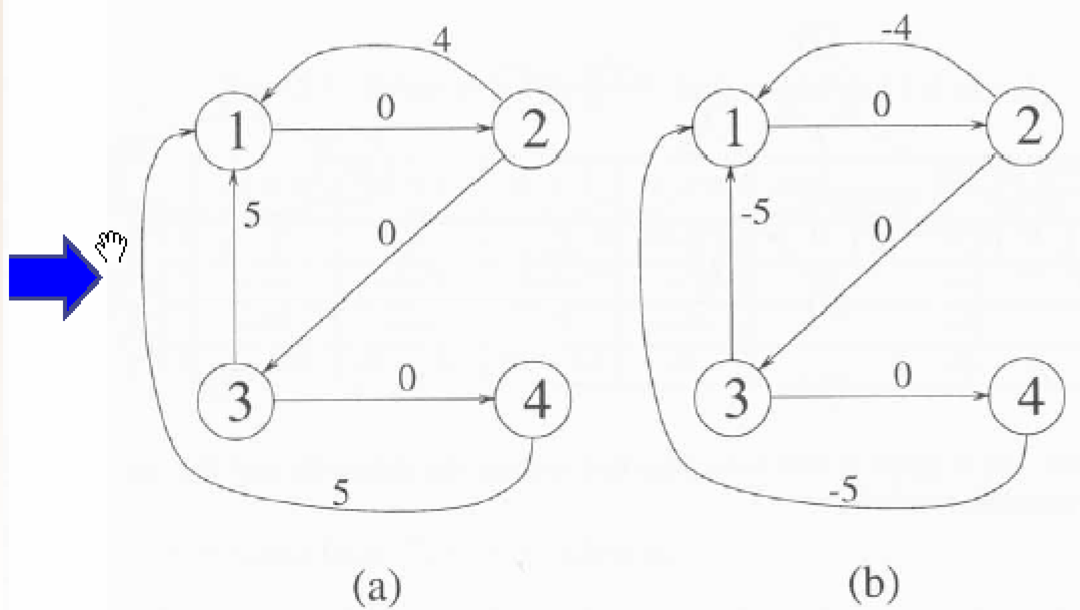
A DFG with 3 loops using MCM (1/5)

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 -> j does not exist in Gd.

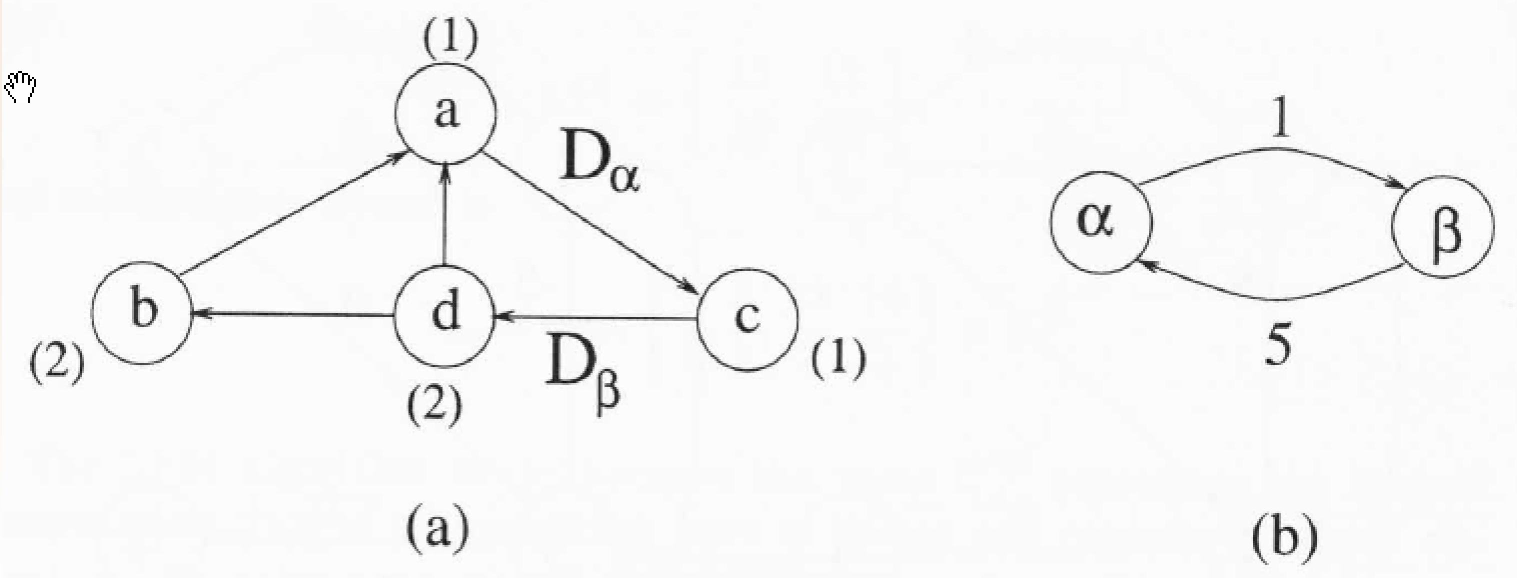




A DFG with 3 loops using MCM (2/5)

Longest path length

- path which pass through no delays
- longest : two loops that contain D_a and Db
  - max { 6,4 } = 6
- cycle mean = 6/2=3



A DFG with 3 loops using MCM (3/5)

Cycle mean= Average length of the edge in c (Cycle = Loop)

Compute the cycle mean

A DFG with 3 loops using MCM (4/5)

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

$$f^{(m)}(j) = \min_{i \in I}\left(f^{(m-1)}(i) + \overline{w}(i, j)\right)$$

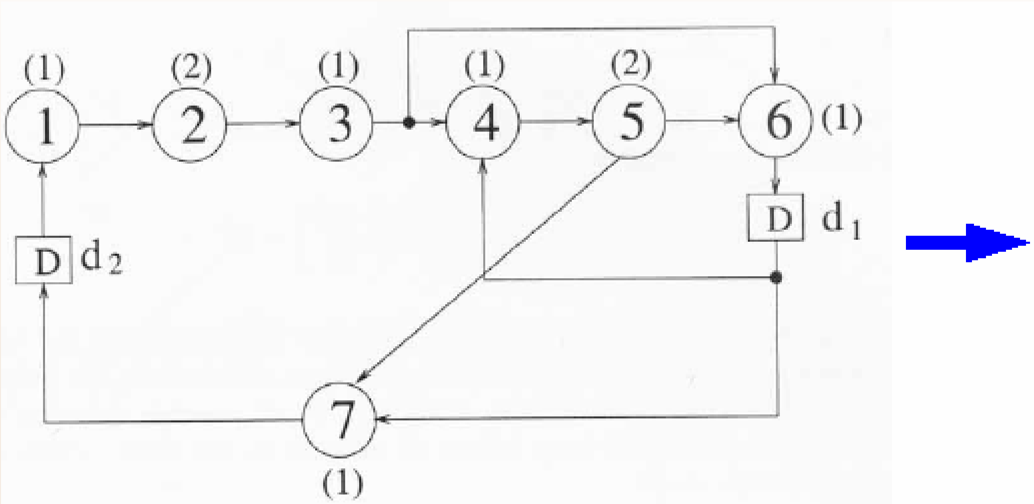
A DFG with 3 loops using MCM (5/5)

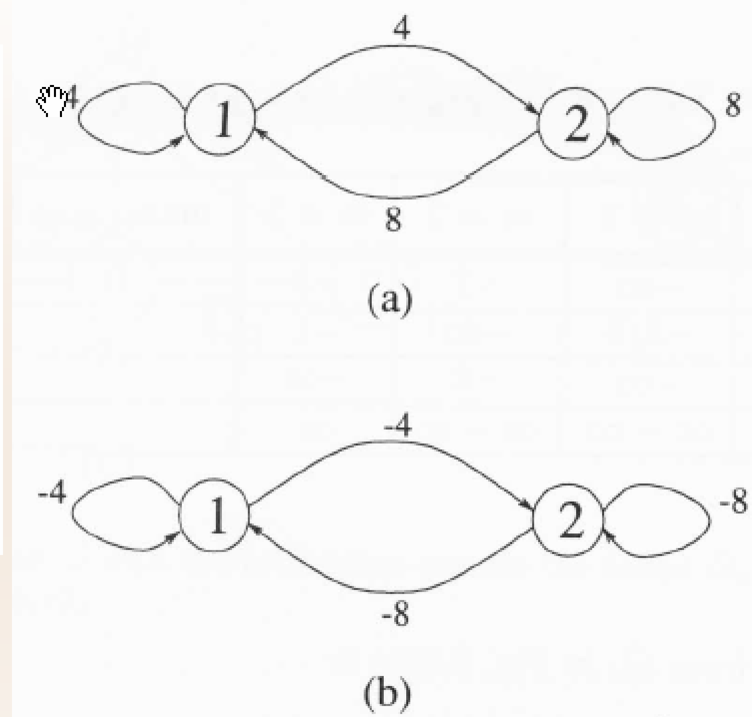
$$T_\infty = -\min_{i\in\{1,2,\ldots,d\}}\left(\max_{m\in\{0,1,2,\ldots d-1\}}\left(\frac{f^{(d)}(i)-f^{(m)}(i)}{d-m}\right)\right)$$

	$m=0$	$m=1$	$m=2$	$m=3$	$\max_{0\le m\le 3}\left\{\frac{f^{(4)}(i)-f^{(m)}(i)}{4-m}\right\}$
$i=1$	-2	$-\infty$	-2	-3	-2
$i=2$	$-\infty$	$-5/3$	$-\infty$	-1	-1
$i=3$	$-\infty$	$-\infty$	-2	$-\infty$	-2
$i=4$	$\infty-\infty$	$\infty-\infty$	$\infty-\infty$	∞	∞

$$T_\infty = -\min\{-2,-1,-2,\infty\} = 2$$

A Filter Using MCM





$$f^{(0)} = \begin{bmatrix} 0 \\ \infty \end{bmatrix} \quad f^{(1)} = \begin{bmatrix} -4 \\ -4 \end{bmatrix} \quad f^{(2)} = \begin{bmatrix} -12 \\ -12 \end{bmatrix}$$

	$m=0$	$m=1$	$\max_{0\le m\le 1}\left\{\frac{f^{(2)}(i)-f^{(m)}(i)}{2-m}\right\}$
$i=1$	$-12/2$	$-8/1$	-6
$i=2$	$-\infty$	$-8/1$	-8

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

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