Ch.12.6 Discrete-time filter structures (realizations)

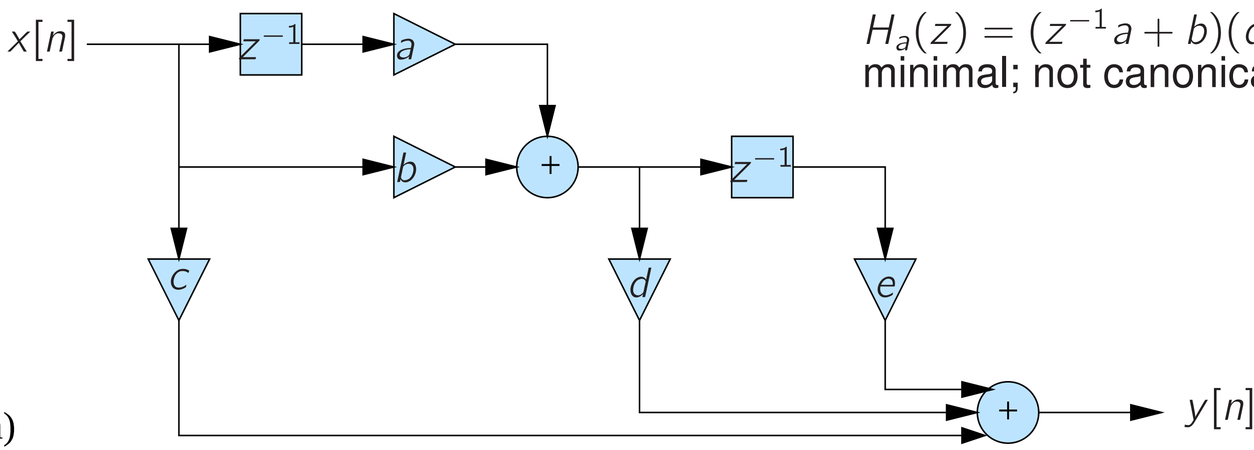
Contents

- FIR filters (direct form – tapped delay line)
- IIR filters (direct form 1, 2)
- Cascade and parallel structure
- Transposition

Discrete-time filter structures

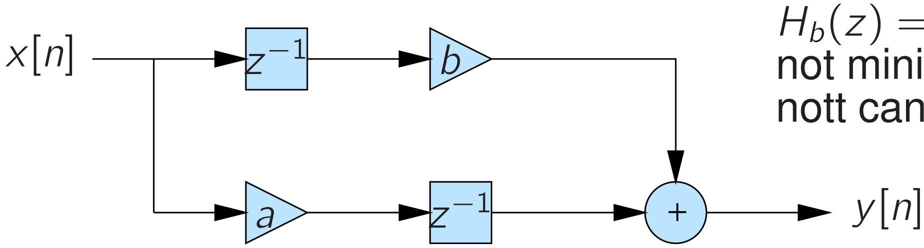
Minimal and canonical realizations

A structure which implements an N-th order transfer function is called minimal if it uses exactly N delay elements; and canonical if also the number of operations (multiplications with coefficients) is minimal while all possible transfer functions of that class can be realized.



(a)

$H_a(z) = (z^{-1}a + b)(d + z^{-1}e) + c$
minimal; not canonical for FIR filters



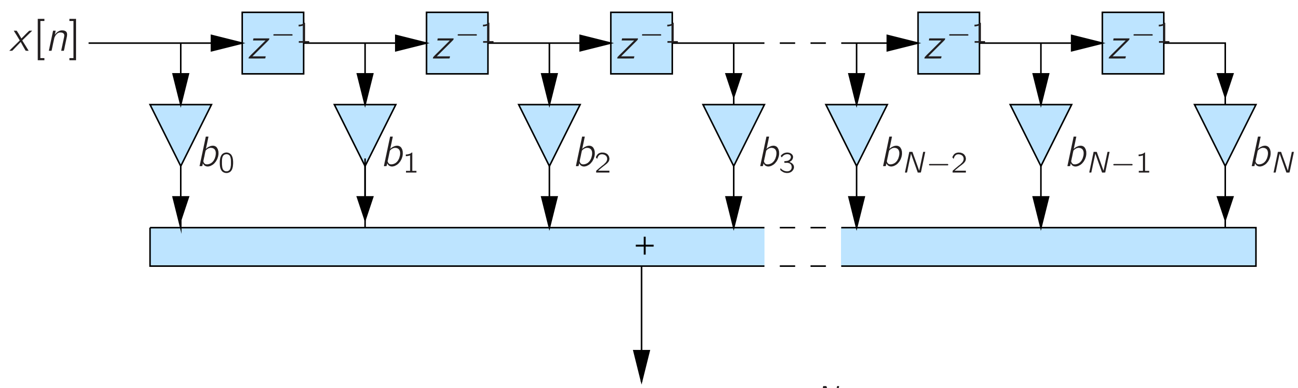
(b)

$H_b(z) = z^{-1}(a + b)$
not minimal;
nott canonical for FIR filters

Discrete-time filter structures

Transversal filter

An FIR filter can be realized using a *transversal filter*:



$y[n] = \sum_{k=0}^{N} b_k x[n-k]$

- Minimal and canonical for the class of N-th order FIR filters:

  N delays; $N+1$ coefficients, corresponding to $N+1$ degrees of freedom in $H(z)$

- The coefficients $h[n] = b_n$ are directly used in the realization

- The transfer function is

$$H(z) = \frac{Y(z)}{X(z)} = b_0 + b_1 z^{-1} + \ldots + b_N z^{-N} = \sum_{k=0}^{N} b_k z^{-k} = \frac{\sum_{k=0}^{N} b_k z^{N-k}}{z^N}$$

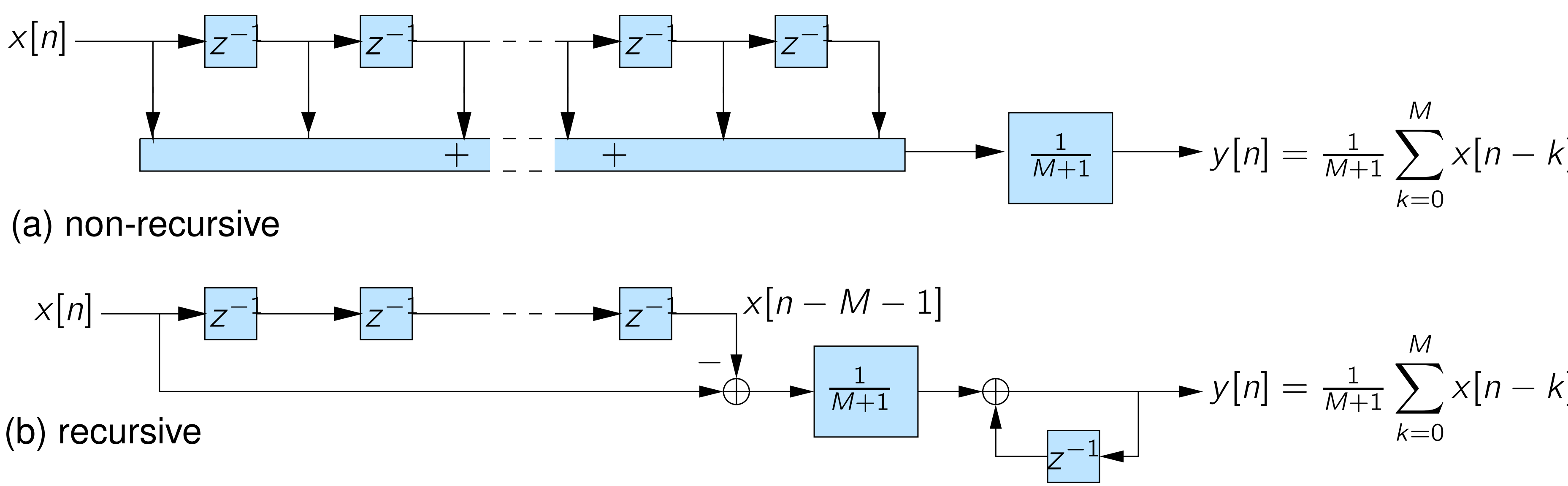
Discrete-time filter structures

An FIR filter can sometimes also be implemented recursively: e.g.,

$$y[n] = \frac{1}{M+1}\sum_{k=0}^{M} x[n-k] = \frac{1}{M+1}\left(x[n] + x[n-1] + \cdots + x[n-M]\right)$$

can be written as

$$\begin{aligned} y[n] &= \frac{1}{M+1}\left(x[n-1] + \cdots + x[n-M] + x[n-M-1]\right) + \frac{1}{M+1}\left(x[n] - x[n-M-1]\right) \\ &= y[n-1] + \frac{1}{M+1}\left(x[n] - x[n-M-1]\right) \end{aligned}$$

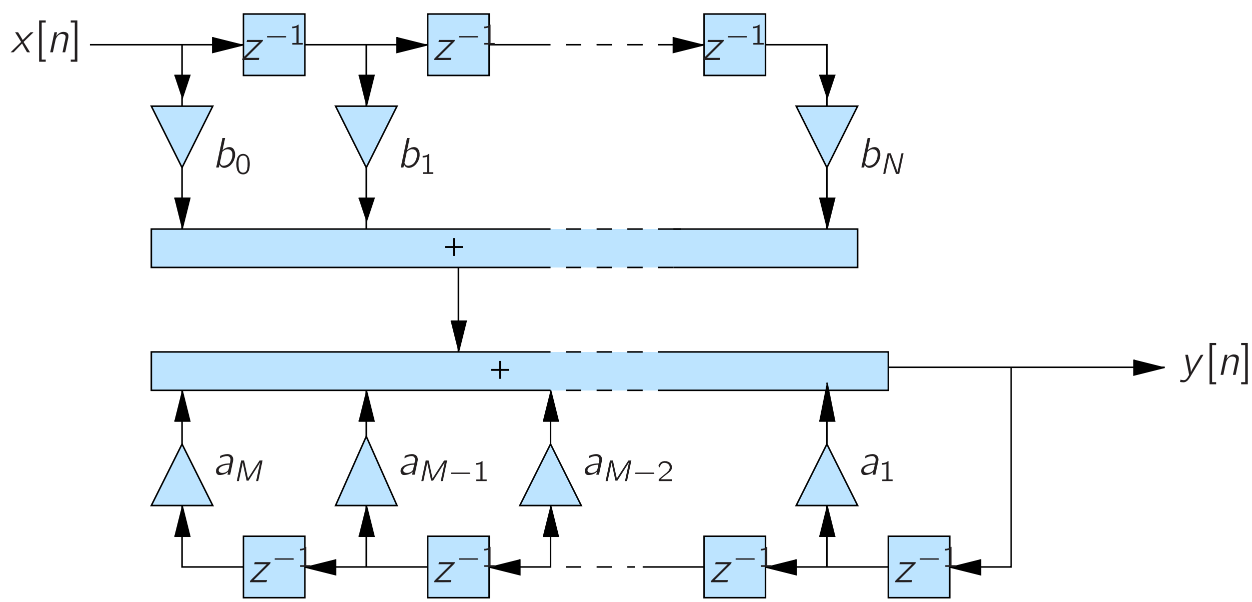


Discrete-time filter structures

Recursive filter: direct form no. 1

Realization for a general difference equation ($a_0 = 1$)

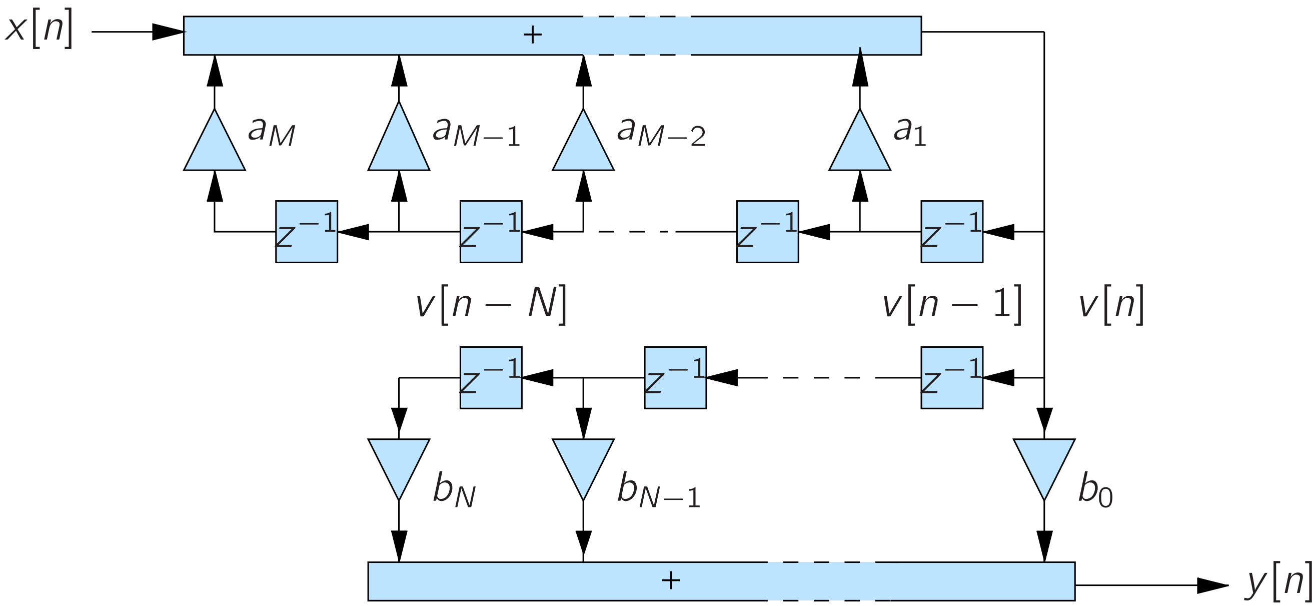
$$y[n] = \sum_{i=0}^{N} b_i x[n-i] + \sum_{i=1}^{M} a_i y[n-i] \qquad \Rightarrow \qquad H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i=0}^{N} b_i z^{-i}}{1 - \sum_{i=1}^{M} a_i z^{-i}}$$



This is not a minimal structure ($M + N$ delays instead of $\max(M, N)$ delays).

Recursive filter: direct form no. 2

Use the commutative property of the convolution: $h_1 * h_2 = h_2 * h_1$. We may reverse the order of both partial systems.

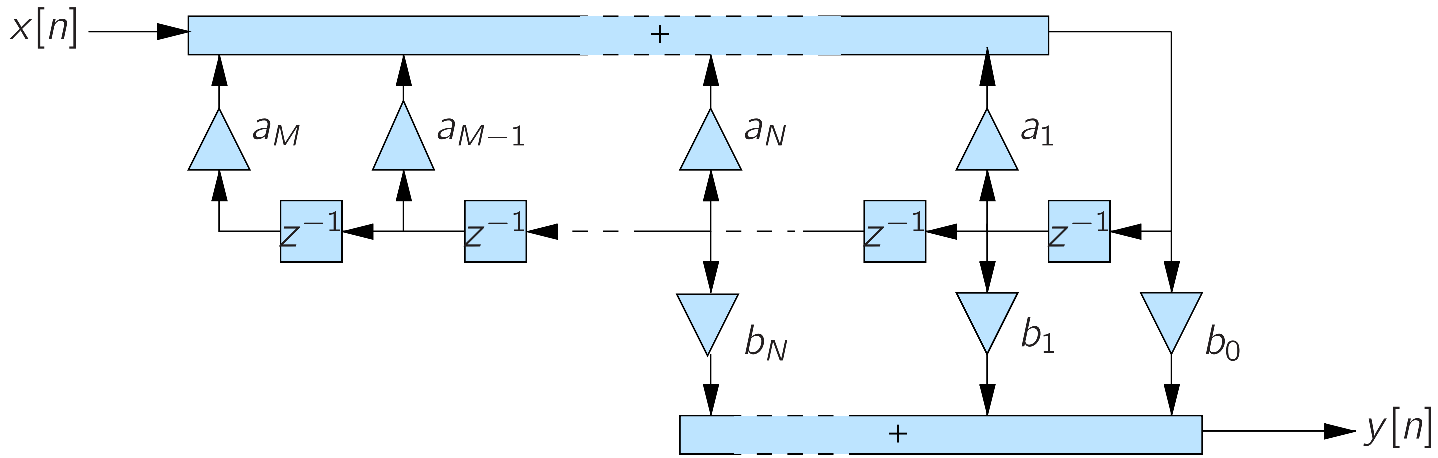


It is seen that the delay lines can be merged (they transport the same signal $v[n]$)

Discrete-time filter structures

Recursive filter: direct form no. 2 (cont'd)

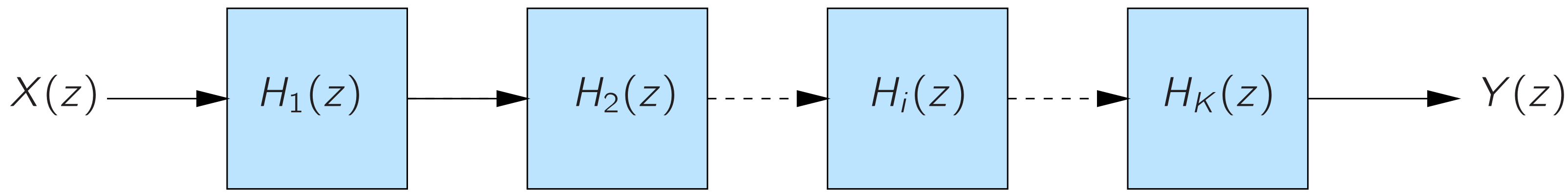
The resulting filter (minimal and canonical):



- Also in this realization the filter coefficients are directly related to the parameters in the difference equation.
- This realization is very sensitive to small disturbances (quantization) of the coefficients: the poles/zeros can move a lot.

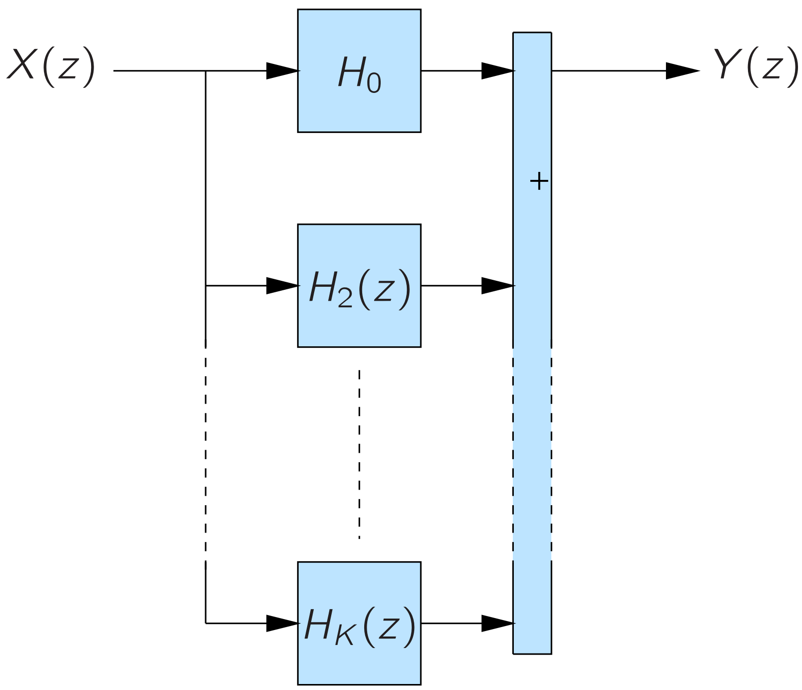
Discrete-time filter structures

Cascade structure



$$H(z) = H_1(z) \cdot H_2(z) \cdots H_K(z) \qquad \text{e.g. } H_k(z) = G_k \frac{(1 - z_k z^{-1})(1 - z_k^* z^{-1})}{(1 - p_k z^{-1})(1 - p_k^* z^{-1})}$$

Parallel structure



$$H(z) = H_1(z) + H_2(z) + \cdots + H_K(z)$$

$$\text{e.g. } H_k(z) = \frac{A_k}{1 - p_k z^{-1}} + \frac{A_k^*}{1 - p_k^* z^{-1}}$$

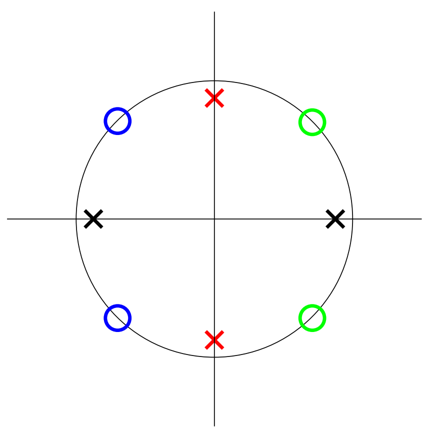
(2nd order section for complex conj. poles)

Discrete-time filter structures

Example

$$H(z) = G\frac{(1-e^{j\pi/4}z^{-1})(1-e^{-j\pi/4}z^{-1})(1-e^{j3\pi/4}z^{-1})(1-e^{-j3\pi/4}z^{-1})}{(1-0.9z^{-1})(1+0.9z^{-1})(1-0.9j\,z^{-1})(1+0.9j\,z^{-1})}$$

There are several possibilities to split this into 2nd order sections with real-valued coefficients. For a cascade, we can also choose which pair of zeros we combine with which pair of poles. With infinite accuracy (no quantization) this does not make a difference.

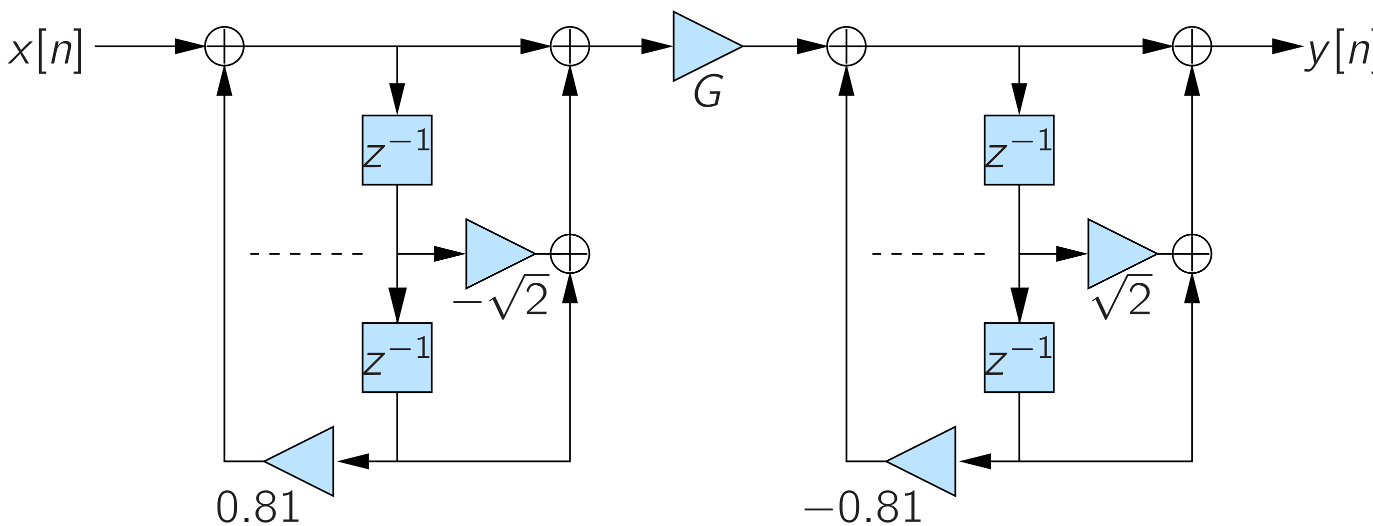


Example (cont'd)

E.g. $H(z) = GH_1(z)H_2(z)$, with

$$H_1(z) = \frac{(1-e^{j\pi/4}z^{-1})(1-e^{-j\pi/4}z^{-1})}{(1-0.9z^{-1})(1+0.9z^{-1})} = \frac{1-\sqrt{2}z^{-1}+z^{-2}}{1-0.81z^{-2}}$$

$$H_2(z) = \frac{(1-e^{j3\pi/4}z^{-1})(1-e^{-3j\pi/4}z^{-1})}{(1-0.9j\,z^{-1})(1+0.9j\,z^{-1})} = \frac{1+\sqrt{2}z^{-1}+z^{-2}}{1+0.81z^{-2}}$$



Second order sections are needed for a canonical realization of transfer functions with *real-valued* coefficients.

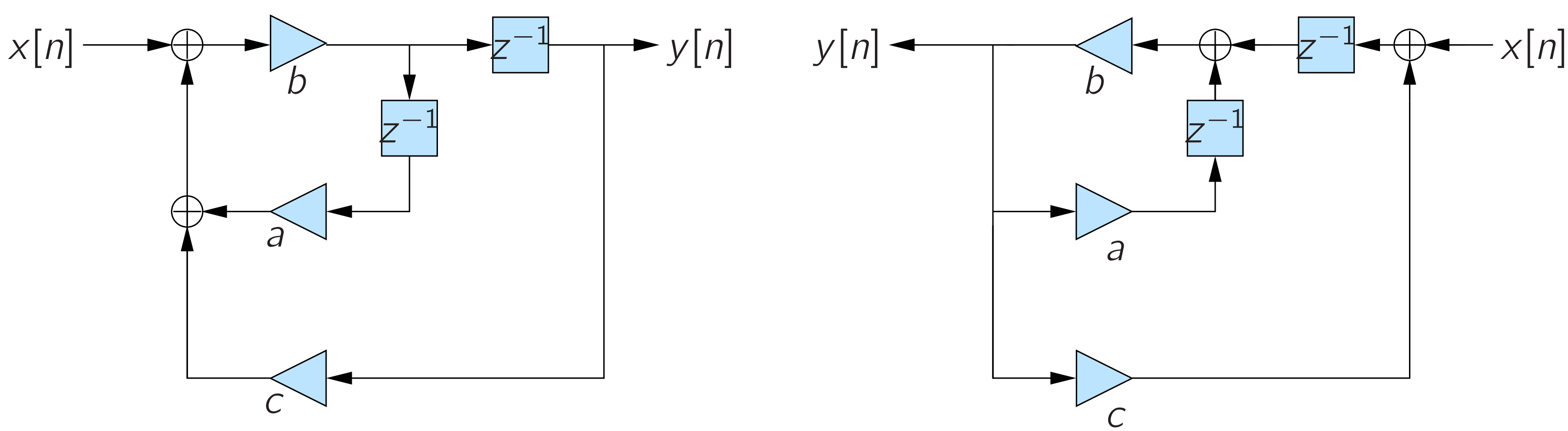
Transposition

Proposition: Given a realization (graph/network with nodes and edges). Make the following changes:

1. Reverse the direction of every edge (adders $\leftrightarrow$ nodes)
2. Reverse input and output

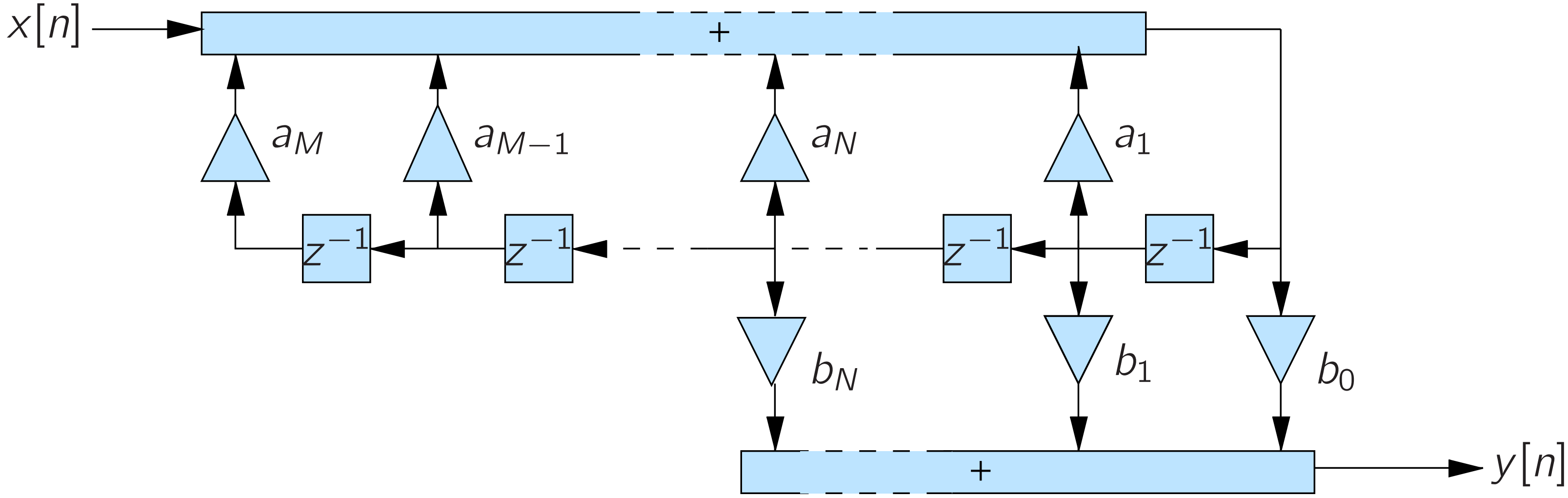
The transfer function is not changed.

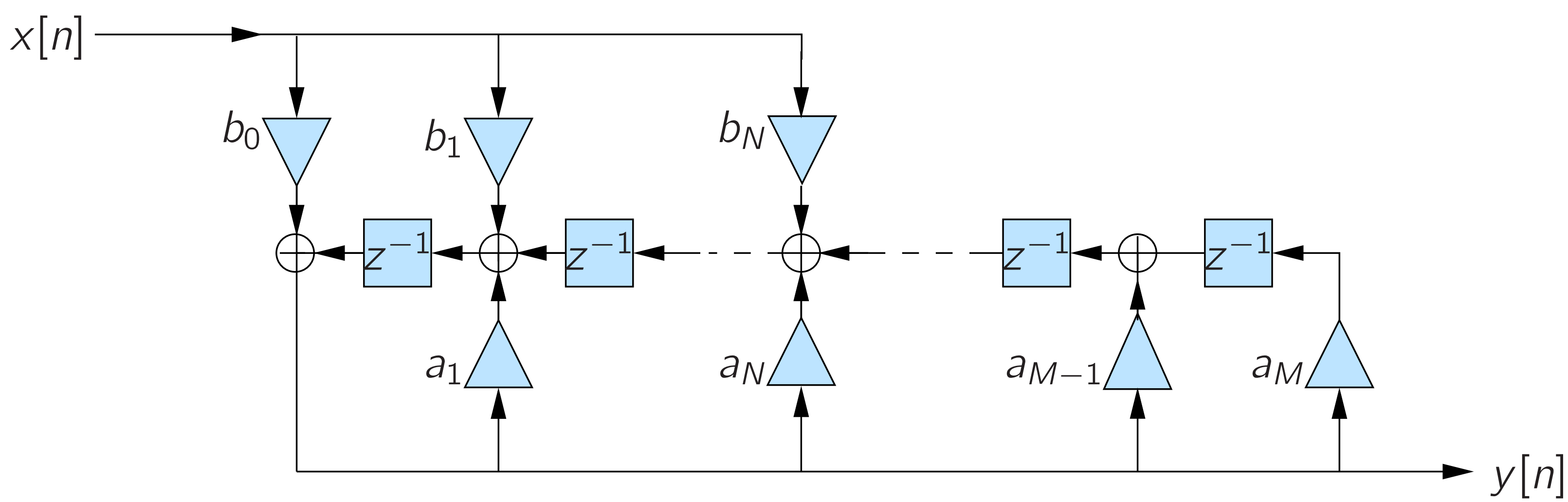
Example: $H(z) = \dfrac{\dfrac{bz^{-1}}{1 - abz^{-1}}}{1 - c\,\dfrac{bz^{-1}}{1 - abz^{-1}}}$



Discrete-time filter structures

Application to direct form no. 2





Advantage: a much shorter critical path (all adders can operate in parallel).