ECE 3640 - Discrete-Time Signals and Systems
Convolution Details

Jake Gunther

UtahStateUniversity

Department of Electrical & Computer Engineering

convolution as

matrix-vector product

linear convolution

$$y(n) = \sum_k h(k)x(n-k)$$

- example: $x(n)$ has length $N = 7$ and $h(n)$ has length $M = 3$

$$\begin{aligned}
y(0) &= h(0)x(0), \\
y(1) &= h(0)x(1) + h(1)x(0), \\
y(2) &= h(0)x(2) + h(1)x(1) + h(2)x(0), \\
y(3) &= h(0)x(3) + h(1)x(2) + h(2)x(1), \\
y(4) &= h(0)x(4) + h(1)x(3) + h(2)x(2), \\
y(5) &= h(0)x(5) + h(1)x(4) + h(2)x(3), \\
y(6) &= h(0)x(6) + h(1)x(5) + h(2)x(4), \\
y(7) &= h(1)x(6) + h(2)x(5), \\
y(8) &= h(2)x(6),
\end{aligned}$$

convolution and structured matrices

$$
\begin{aligned}
y(0) &= h(0)x(0), \\
y(1) &= h(0)x(1) + h(1)x(0), \\
y(2) &= h(0)x(2) + h(1)x(1) + h(2)x(0), \\
y(3) &= h(0)x(3) + h(1)x(2) + h(2)x(1), \\
y(4) &= h(0)x(4) + h(1)x(3) + h(2)x(2), \\
y(5) &= h(0)x(5) + h(1)x(4) + h(2)x(3), \\
y(6) &= h(0)x(6) + h(1)x(5) + h(2)x(4), \\
y(7) &= h(1)x(6) + h(2)x(5), \\
y(8) &= h(2)x(6),
\end{aligned}
$$

$$
\underbrace{\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \\ y(6) \\ y(7) \\ y(8) \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} h(0) & & & & & & \\ h(1) & h(0) & & & & & \\ h(2) & h(1) & h(0) & & & & \\ & h(2) & h(1) & h(0) & & & \\ & & h(2) & h(1) & h(0) & & \\ & & & h(2) & h(1) & h(0) & \\ & & & & h(2) & h(1) & h(0) \\ & & & & & h(2) & h(1) \\ & & & & & & h(2) \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \end{bmatrix}}_{\mathbf{x}},
$$

$$
\mathbf{y} = \mathbf{H}\mathbf{x}
$$

convolution and structured matrices

$$
\begin{aligned}
y(0) &= h(0)x(0),\\
y(1) &= h(0)x(1) + h(1)x(0),\\
y(2) &= h(0)x(2) + h(1)x(1) + h(2)x(0),\\
y(3) &= h(0)x(3) + h(1)x(2) + h(2)x(1),\\
y(4) &= h(0)x(4) + h(1)x(3) + h(2)x(2),\\
y(5) &= h(0)x(5) + h(1)x(4) + h(2)x(3),\\
y(6) &= h(0)x(6) + h(1)x(5) + h(2)x(4),\\
y(7) &= h(1)x(6) + h(2)x(5),\\
y(8) &= h(2)x(6),
\end{aligned}
$$

$$
\underbrace{\begin{bmatrix} y(0)\\ y(1)\\ y(2)\\ y(3)\\ y(4)\\ y(5)\\ y(6)\\ y(7)\\ y(8) \end{bmatrix}}_{\mathbf{y}}
=
\underbrace{\begin{bmatrix}
x(0) & & \\
x(1) & x(0) & \\
x(2) & x(1) & x(0)\\
x(3) & x(2) & x(1)\\
x(4) & x(3) & x(2)\\
x(5) & x(4) & x(3)\\
x(6) & x(5) & x(4)\\
 & x(6) & x(5)\\
 & & x(6)
\end{bmatrix}}_{\mathbf{X}}
\underbrace{\begin{bmatrix} h(0)\\ h(1)\\ h(2) \end{bmatrix}}_{\mathbf{h}},
$$

$$\mathbf{y} = \mathbf{X}\mathbf{h}$$

convolution tails

- a filter is not filtering until it is filled with data
- there are transients when the filter is filling and emptying

$$
\begin{aligned}
y(0) &= h(0)x(0), \\
y(1) &= h(0)x(1) + h(1)x(0), \\
y(2) &= h(0)x(2) + h(1)x(1) + h(2)x(0), \\
y(3) &= h(0)x(3) + h(1)x(2) + h(2)x(1), \\
y(4) &= h(0)x(4) + h(1)x(3) + h(2)x(2), \\
y(5) &= h(0)x(5) + h(1)x(4) + h(2)x(3), \\
y(6) &= h(0)x(6) + h(1)x(5) + h(2)x(4), \\
y(7) &= h(1)x(6) + h(2)x(5), \\
y(8) &= h(2)x(6),
\end{aligned}
$$

- length $x(n)$ is $N = 7$, length $h(n)$ is $M = 3$, length $y(n)$ is $N + M - 1 = 9$
- convolution tail $M - 1 = 2$ (assumes $N > M$)
- number of valid outputs is $N - M + 1 = 5$

convolution and structured matrices

$$
\begin{aligned}
y(0) &= h(0)x(0),\\
y(1) &= h(0)x(1) + h(1)x(0),\\
y(2) &= h(0)x(2) + h(1)x(1) + h(2)x(0),\\
y(3) &= h(0)x(3) + h(1)x(2) + h(2)x(1),\\
y(4) &= h(0)x(4) + h(1)x(3) + h(2)x(2),\\
y(5) &= h(0)x(5) + h(1)x(4) + h(2)x(3),\\
y(6) &= h(0)x(6) + h(1)x(5) + h(2)x(4),\\
y(7) &= h(1)x(6) + h(2)x(5),\\
y(8) &= h(2)x(6),
\end{aligned}
$$

$$
\underbrace{\left[\begin{array}{c}
y(0)\\ y(1)\\ \hline y(2)\\ y(3)\\ y(4)\\ y(5)\\ y(6)\\ \hline y(7)\\ y(8)
\end{array}\right]}_{\mathbf{y}}
=
\underbrace{\left[\begin{array}{ccccccc}
h(0) & & & & & & \\
h(1) & h(0) & & & & & \\ \hline
h(2) & h(1) & h(0) & & & & \\
 & h(2) & h(1) & h(0) & & & \\
 & & h(2) & h(1) & h(0) & & \\
 & & & h(2) & h(1) & h(0) & \\
 & & & & h(2) & h(1) & h(0)\\ \hline
 & & & & & h(2) & h(1)\\
 & & & & & & h(2)
\end{array}\right]}_{\mathbf{H}}
\underbrace{\left[\begin{array}{c}
x(0)\\ x(1)\\ x(2)\\ x(3)\\ x(4)\\ x(5)\\ x(6)
\end{array}\right]}_{\mathbf{x}},
$$

$$
\mathbf{y} = \mathbf{H}\mathbf{x}
$$

convolution tails

$$
\begin{aligned}
y(0) &= h(0)x(0),\\
y(1) &= h(0)x(1) + h(1)x(0),\\
y(2) &= h(0)x(2) + h(1)x(1) + h(2)x(0),\\
y(3) &= h(0)x(3) + h(1)x(2) + h(2)x(1),\\
y(4) &= h(0)x(4) + h(1)x(3) + h(2)x(2),\\
y(5) &= h(0)x(5) + h(1)x(4) + h(2)x(3),\\
y(6) &= h(0)x(6) + h(1)x(5) + h(2)x(4),\\
y(7) &= h(1)x(6) + h(2)x(5),\\
y(8) &= h(2)x(6),
\end{aligned}
$$

$$
\underbrace{\left[\begin{array}{c} y(0)\\ y(1)\\ \hline y(2)\\ y(3)\\ y(4)\\ y(5)\\ y(6)\\ \hline y(7)\\ y(8) \end{array}\right]}_{\mathbf{y}}
=
\underbrace{\left[\begin{array}{ccc} x(0) & & \\ x(1) & x(0) & \\ \hline x(2) & x(1) & x(0)\\ x(3) & x(2) & x(1)\\ x(4) & x(3) & x(2)\\ x(5) & x(4) & x(3)\\ x(6) & x(5) & x(4)\\ \hline & x(6) & x(5)\\ & & x(6) \end{array}\right]}_{\mathbf{X}}
\underbrace{\begin{bmatrix} h(0)\\ h(1)\\ h(2) \end{bmatrix}}_{\mathbf{h}},
$$

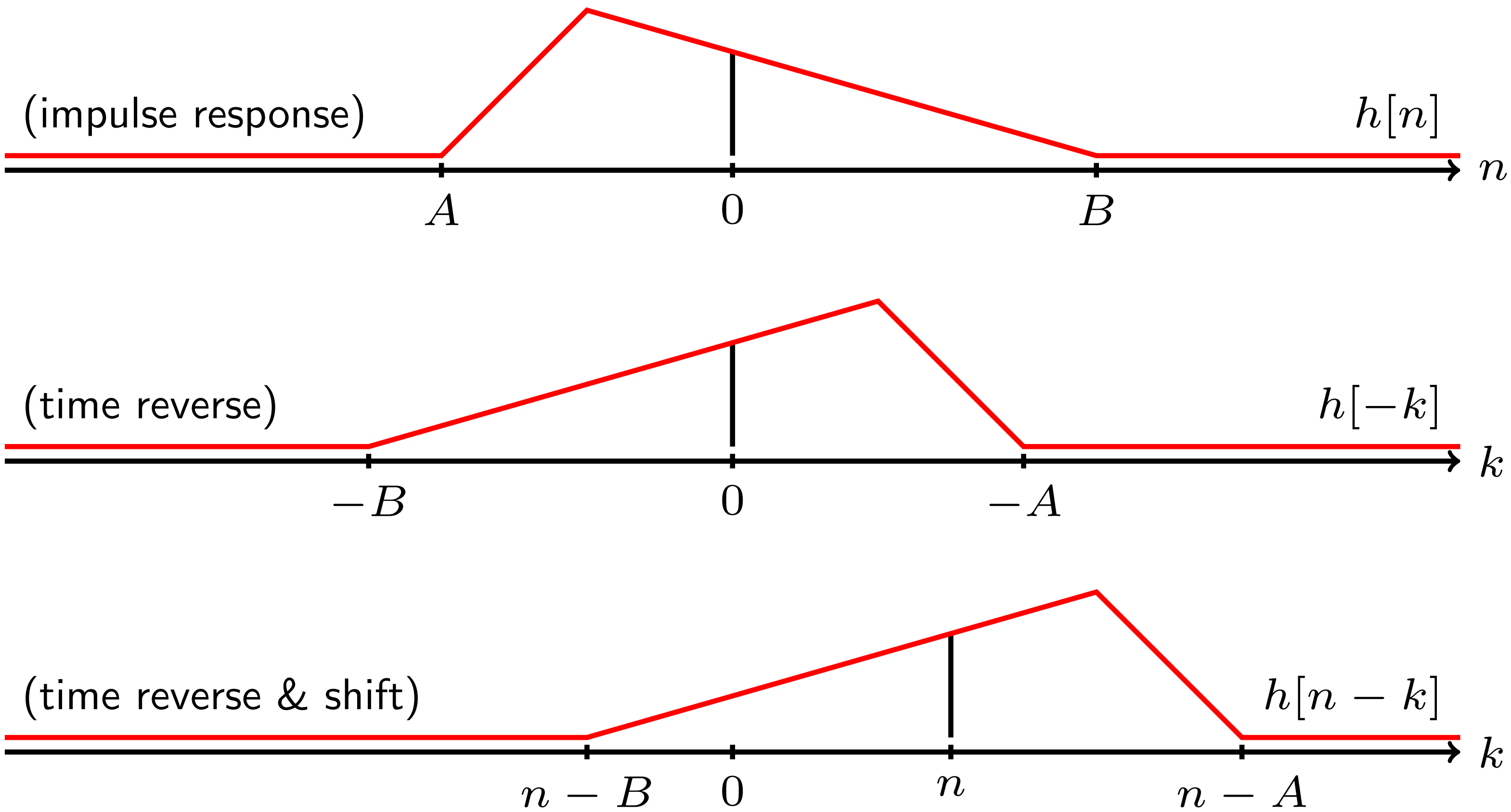
$$\mathbf{y} = \mathbf{X}\mathbf{h}$$

mechanics of convolution

mechanics of convolution

$$y[n] = x[n] * h[n] = \sum_k x[k]h[n-k]$$

how do we sketch $h[n-k]$? (k is the variable and n is fixed)

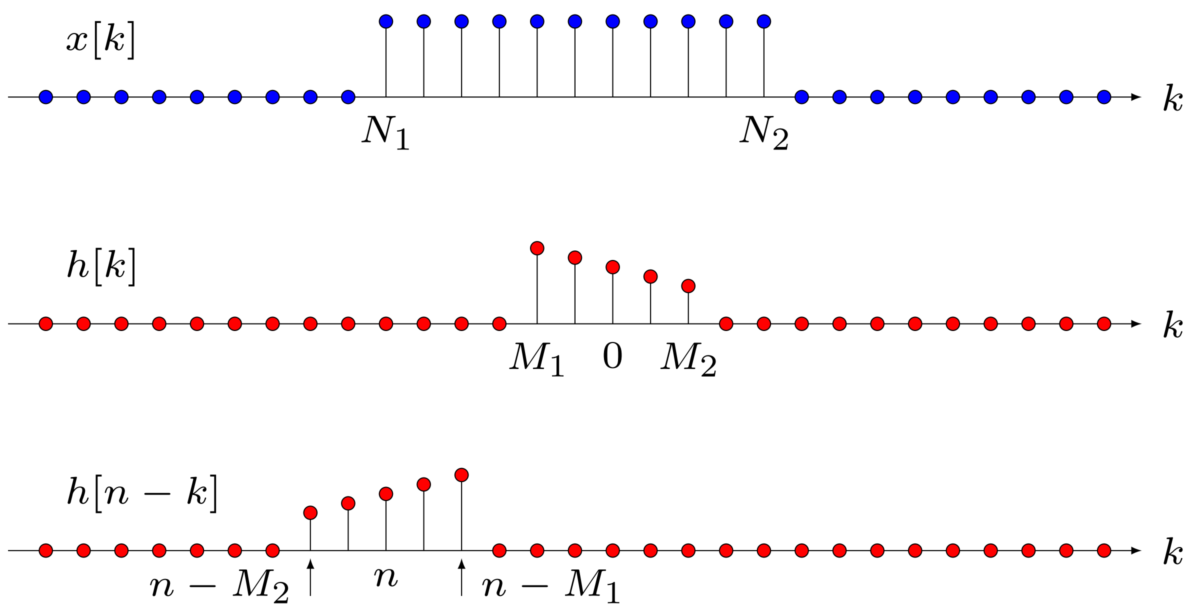


principle: line up zero point of $h[-k]$ on sample n

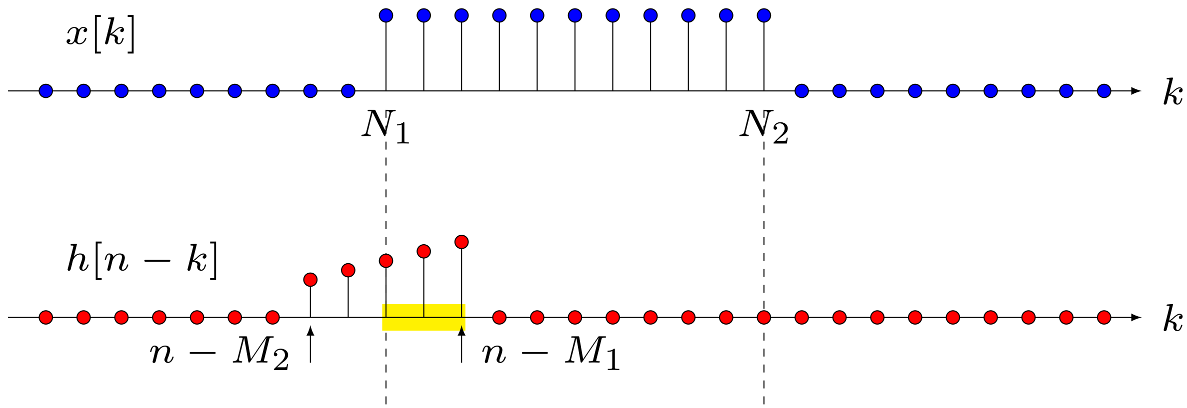
convolution: $$y[n] = \sum_k x[k]h[n-k]$$

$$x[n] = 0 \quad \text{for} \quad n < N_1 \quad \text{and} \quad n > N_2$$

$$h[n] = 0 \quad \text{for} \quad n < M_1 \quad \text{and} \quad n > M_2$$



convolution: $$y[n] = \sum_k x[k]h[n-k]$$



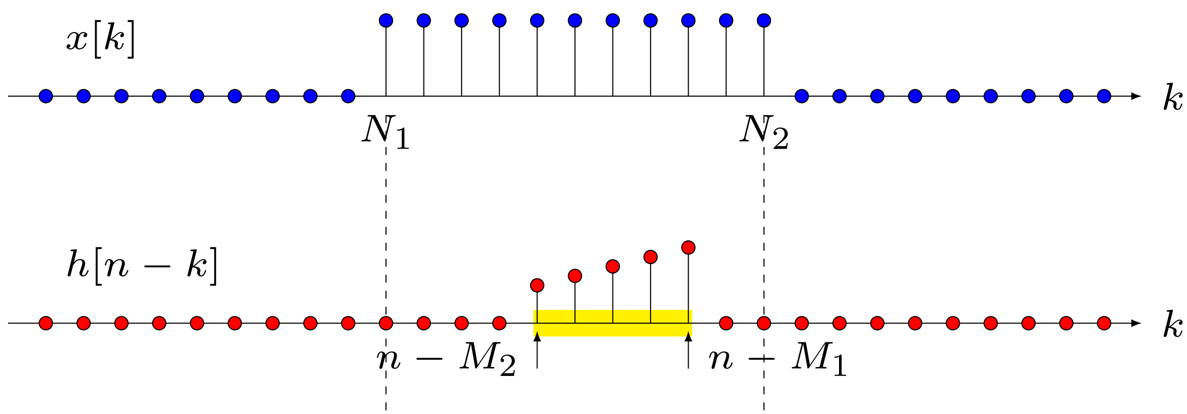
This placement is valid for output times

$$\begin{aligned} n - M_2 &\leq N_1 \\ N_1 &\leq n - M_1 \end{aligned} \quad \Longrightarrow \quad N_1 + M_1 \leq n \leq N_1 + M_2$$

Limits on the summation obtained by intersecting support sets

$$y[n] = \sum_{k=N_1}^{n-M_1} x[k]h[n-k]$$

convolution: $$y[n] = \sum_k x[k]h[n-k]$$



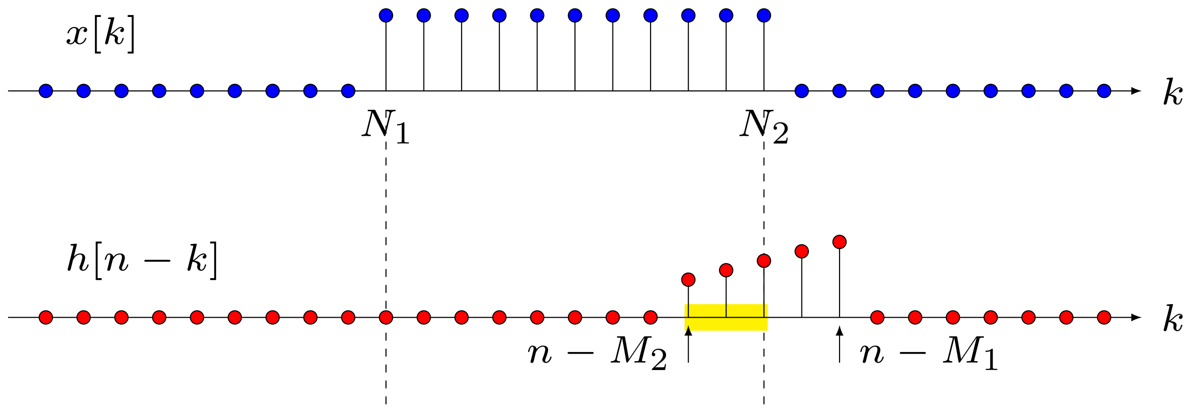
This placement is valid for output times

$$\begin{array}{r} N_1 \le n - M_2 \\ n - M_1 \le N_2 \end{array} \implies N_1 + M_2 \le n \le N_2 + M_1$$

Limits on the summation obtained by intersecting support sets

$$y[n] = \sum_{k=n-M_2}^{n-M_1} x[k]h[n-k]$$

convolution: $$y[n] = \sum_k x[k]h[n-k]$$



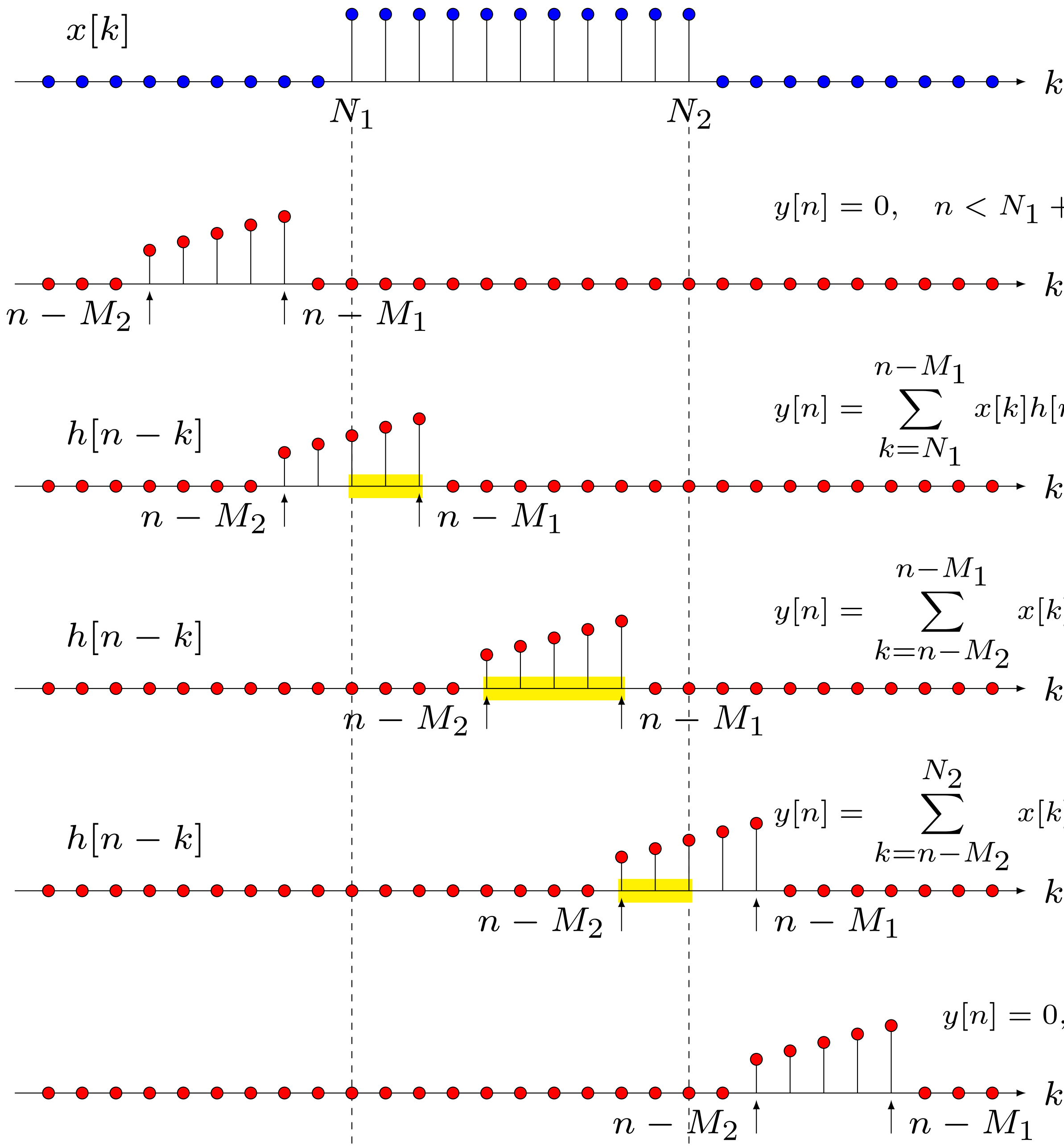
This placement is valid for output times

$$\begin{aligned} n - M_2 &\leq N_2 \\ N_2 &\leq n - M_1 \end{aligned} \quad \Longrightarrow \quad N_2 + M_1 \leq n \leq N_2 + M_2$$

Limits on the summation obtained by intersecting support sets

$$y[n] = \sum_{k=n-M_2}^{N_2} x[k]h[n-k]$$

convolution summary: $$y[n] = \sum_k x[k]h[n-k]$$



$$y[n] = 0, \quad n < N_1 + M_1$$

$$y[n] = \sum_{k=N_1}^{n-M_1} x[k]h[n-k], \quad N_1 + M_1 \le n \le N_1 + M_2$$

$$y[n] = \sum_{k=n-M_2}^{n-M_1} x[k]h[n-k], \quad N_1 + M_2 \le n \le N_2 + M_1$$

$$y[n] = \sum_{k=n-M_2}^{N_2} x[k]h[n-k], \quad N_2 + M_1 \le n \le N_2 + M_2$$

$$y[n] = 0, \quad N_2 + M_2 < n$$

tableau method

for convolution

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2							
0	1							
1	0							
2	2							
m	$h(m)$							

- Fill in the two signals being convolved and the corresponding times

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

- form all the products $h(m)x(n)$

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

- to compute $y(k) = \sum_m h(m)x(k-m) = \sum_m h(m)\, x(n)|_{n=k-m}$, sum all entries in the table where $k = m + n$

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

$y(-3) = 4$ (beginning convolution tail)

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

$y(-2) = 8$ (beginning convolution tail)

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

$y(-1) = 5$ (beginning convolution tail)

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

$y(0) = 9$ (valid filter output)

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

$y(1) = 8$ (valid filter output)

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

$y(2) = 4$ (valid filter output)

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

$y(3) = 5$ (ending convolution tail)

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

$y(4) = 0$ (ending convolution tail)

tableau method by example

		-2	-1	0	1	2	3	n
		2	3	1	2	0	1	$x(n)$
-1	2	4	6	2	4	0	2	
0	1	2	3	1	2	0	1	
1	0	0	0	0	0	0	0	
2	2	4	6	2	4	0	2	
m	$h(m)$							

$y(5) = 2$ (ending convolution tail)