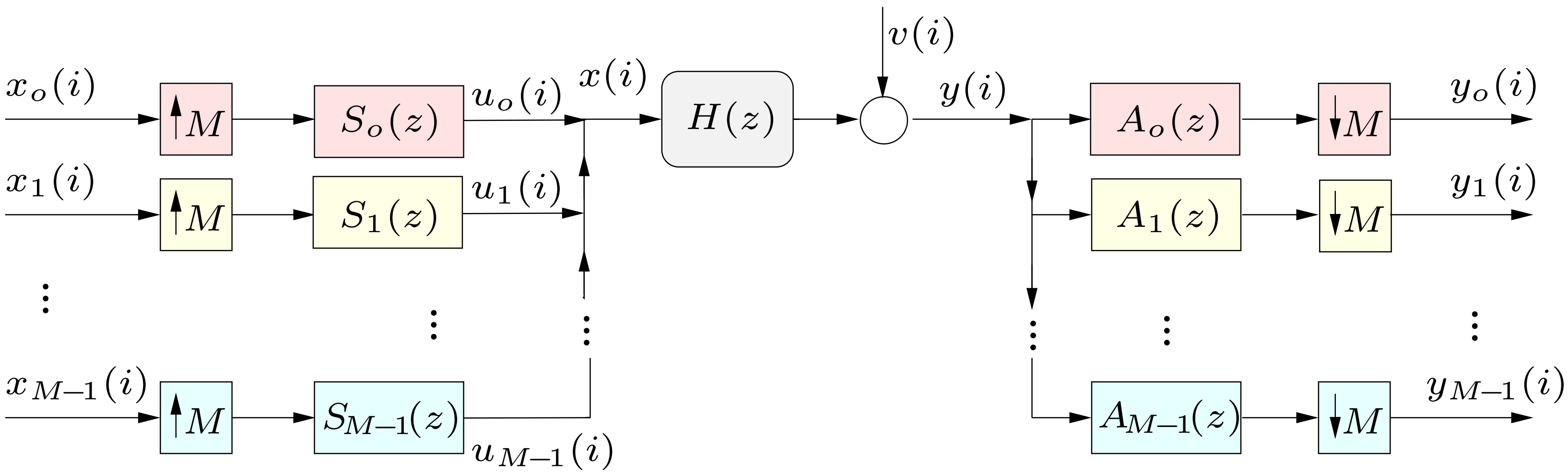
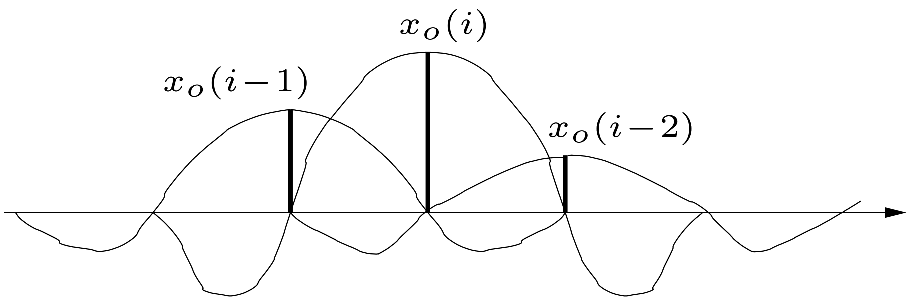
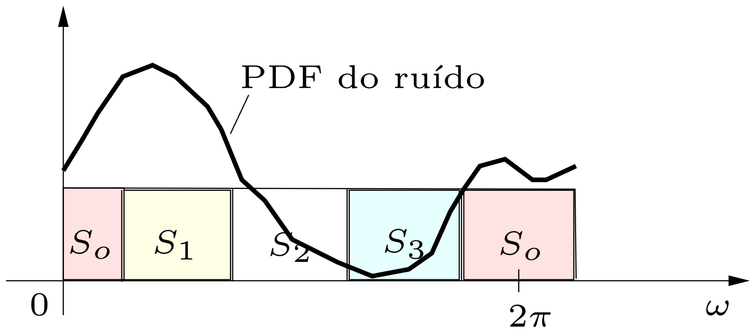
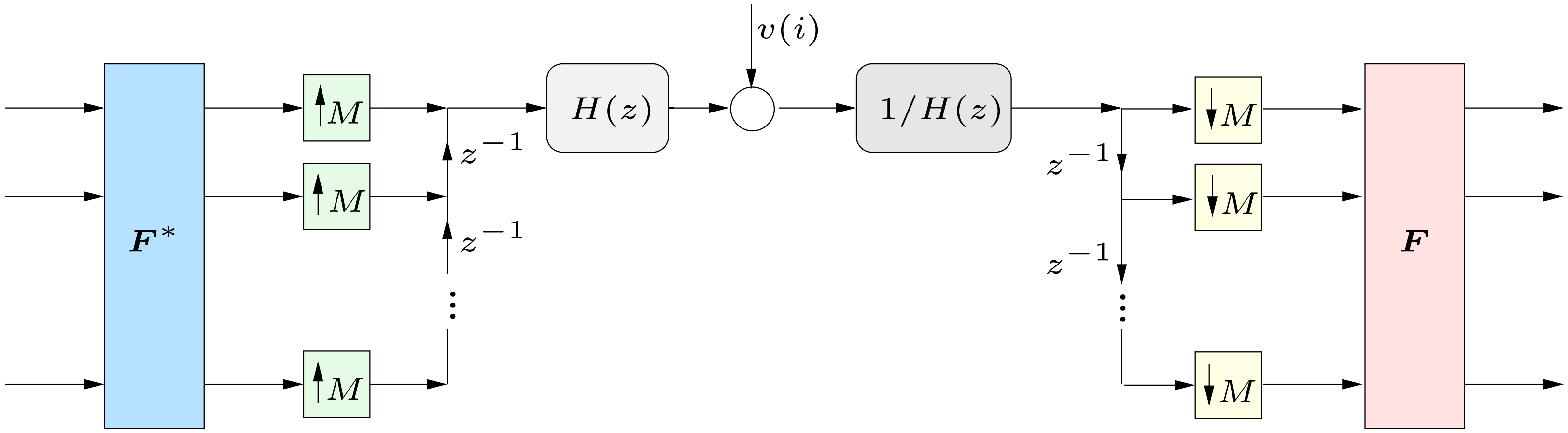
Transmultiplexador Digital

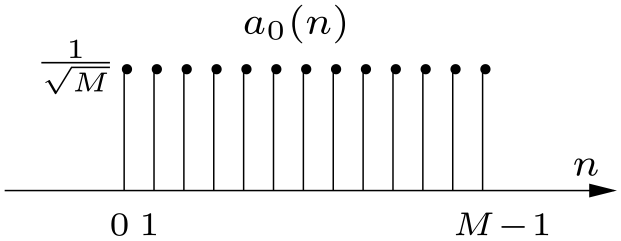




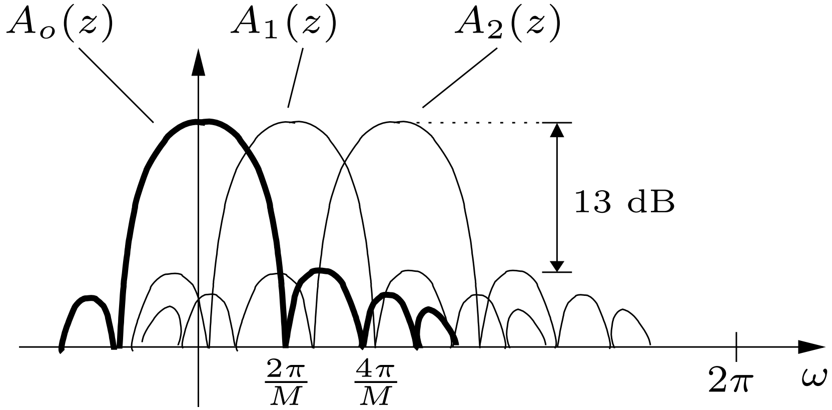




$v(i)$
F^*
M
z^{-1}
$H(z)$
$1/H(z)$
F



$a_0(n)$
$\frac{1}{\sqrt{M}}$
n
0 1
$M-1$



$A_o(z)$
$A_1(z)$
$A_2(z)$
13 dB
$\frac{2\pi}{M}$
$\frac{4\pi}{M}$
2π
ω

Discrete Multitone Modulation (DMT)

- O transmultiplexador é usado em diversas configurações de um sistema de comunicação, dependendo da interpretação dos sinais $x_k(n)$ e $x_(n)$.

- Em DMT, os sinais $x_k(n)$ são gerados a partir de uma mesma sequência $s(n)$, na etapa denominada *parsing*, da seguinte forma:

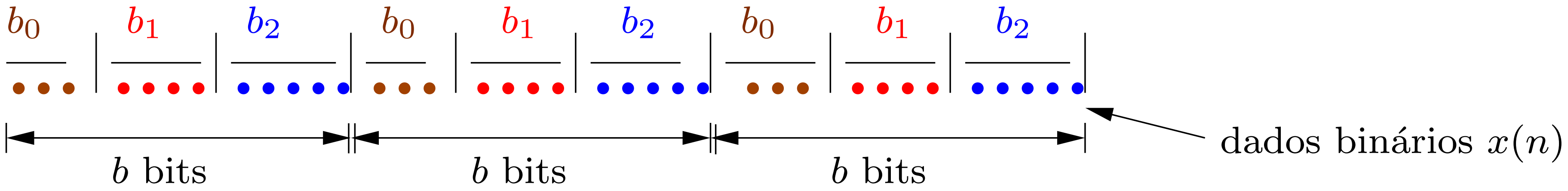
  1. $s(n)$ representa uma sequência binária a ser transmitida, que é dividida em blocos de b bits cada.
  2. Cada bloco de b bits é dividido em M grupos de k bits, de forma que

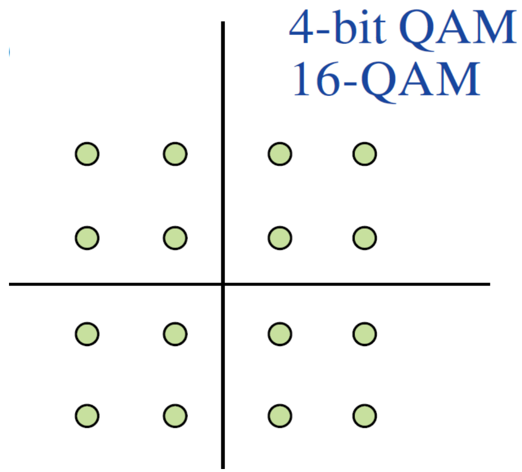
  $$b = \sum_{k=0}^{M-1} b_k$$

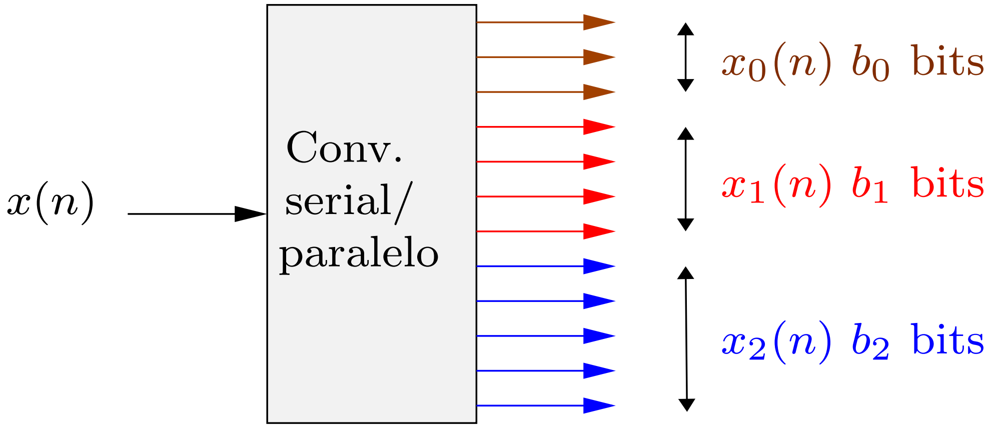
  onde os b_k bits representam o símbolo $x_k(n)$.

- O conjunto $\{x_0(n), x_1(n), \ldots, x_{M-1}(n)\}$ é denominado símbolo DMT.

Parsing



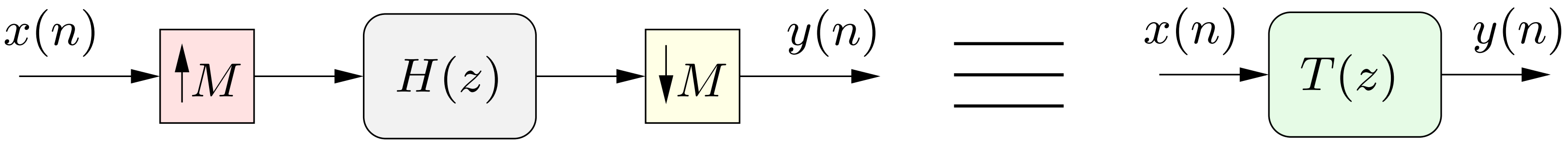




- Para uma dada constelação, a potência pode ser controlada escalando-se seus pontos, e o critério de Waterfilling pode ser aproximado.

Biortogonalidade e Sistemas de Reconstrução Perfeita

- **Propriedade útil :**



$$t(n) = h(Mn)$$
$$\text{ou } T(z) = H(z)_{\downarrow M}$$

- Seja $D_{km}(z)$ a função de transferência entre $x_m(\cdot)$ e $y_m(\cdot)$. Se não houver nenhum tipo de equalização, temos

$$D_{km}(z) = [S_k(z)H(z)A_m(z)]_{\downarrow M}$$

- Se $D_{km}(z)$ não é nulo para $m \neq k \Longrightarrow$ interferência entre-bandas.

- Se $D_{km}(z)$ não é constante para $m = k \Longrightarrow$ interferência intra-banda.

- Se o banco de filtros for ideal, não há interferência entre bandas. Além disso, se $H(z)$ for completamente equalizado por um filtro $1/H(z)$, o sistema é totalmente livre de ISI.

- Filtros ideais não são possíveis e aproximações são complexas.

- Como vimos, é possível projetar-se filtros de reconstrução perfeita mesmo com sobreposição entre bandas.

- Por ex., assumindo que $C(z) = 1$, temos reconstrução perfeita se, e só se, os bancos de análise e síntese satisfazem biortogonalidade:

$$[S_k(z)A_m(z)]_{\downarrow M} = \delta(k - m)$$

$\Longrightarrow G_{km}(z) \triangleq S_k(z)A_m(z)$ deve ter a propriedade de Nyquist(M):

$$g_{km}(Mn) = 0$$

Banco de Filtros Ortonormais

- O sinais $u_k(n)$ são as saídas dos filtros interpoladores, e podem ser interpretados como um subespaço gerado pelas funções bases

$$s_k(n+M), \;\; s_k(n), \;\; s_k(N-M), \;\; s_k(n-2M), \ldots$$

- O conjunto de filtros $\{S_k(z)\}$ é dito ortonormal, se as funções bases são ortonormais umas as outras.

- Mais ainda, para recuperarmos os símbolos perfeitamente, devemos ter **biortogonalidade**:

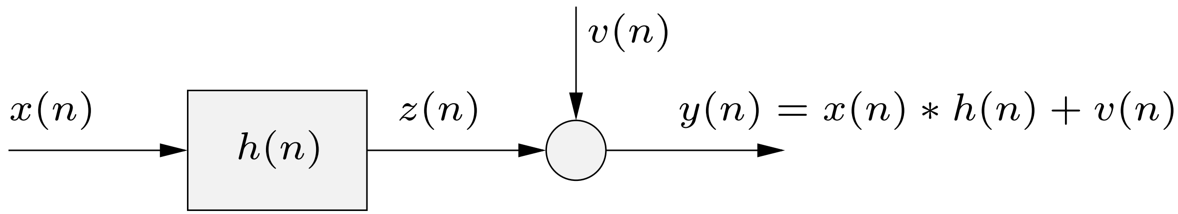
$$a_k(n) = s_k^*(-n) \qquad \text{(time reversed-conjugation)}$$

- É possível construir-se bancos de filtros onde os filtros são FIR. Um exemplo clássico é o caso em que $s_o(n)$ é retangular de largura M e

$$s_k(n) = s_o(n) e^{j\omega_k n} \qquad \text{com } \omega_k = 2\pi k / M$$

Equalização Linear

Modelo discreto de canal:



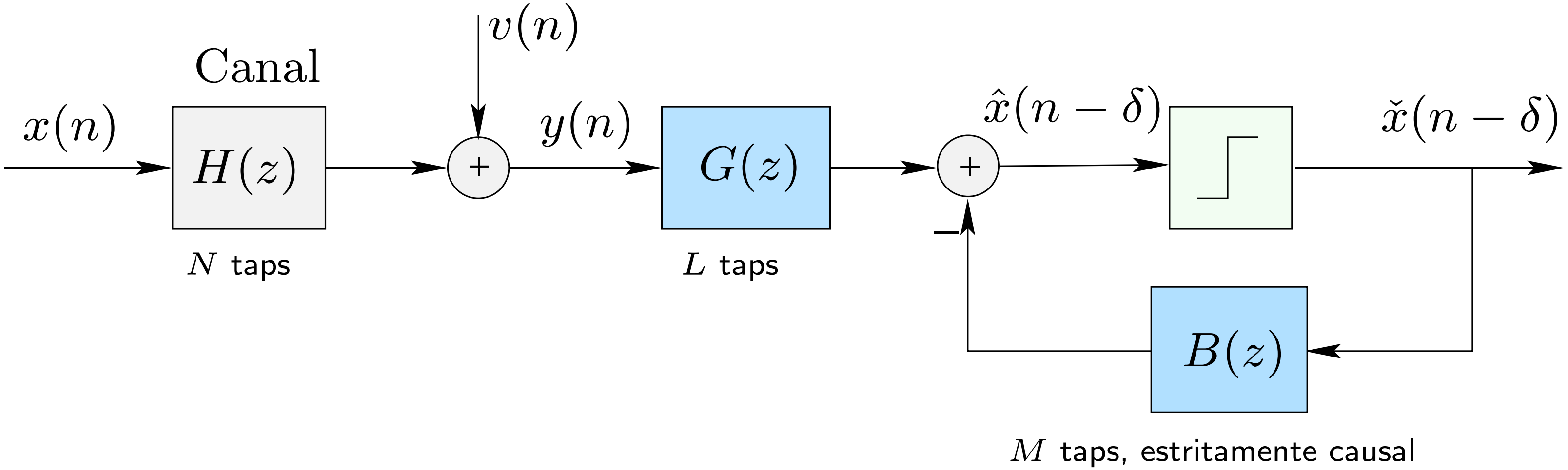
A saída de um canal é dada por

$$y(n) \;=\; h(n) * x(n) \;+\; v(n)$$

ou

$$y(n) \;=\; h(\delta)x(n-\delta) \;+\; \underbrace{\sum_{k=0,k\neq\delta}^{M-1} h(k)x(i-k)}_{\text{ISI}} \;+\; v(n)$$

Equalização não-linear: Decision-Feedback Equalizer (DFE)



$$y(n) \;=\; h(\delta)x(n-\delta) \;+\; \underbrace{\sum_{k=0,k\neq\delta}^{M-1} h(k)x(n-k)}_{\text{ISI}} \;+\; v(n)$$

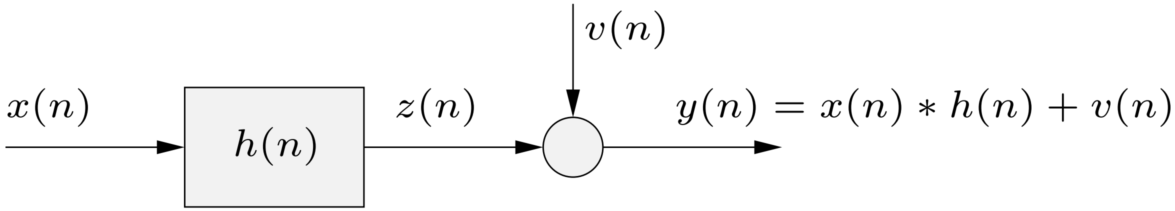
<u>Objetivo</u>: Eliminar a ISI utilizando símbolos detectados anteriormente, isto é, $\{\check{\boldsymbol{x}}(i-\delta-1), \check{\boldsymbol{x}}(i-\delta-2), \ldots\}$.

$\rightarrow$ $G(z)$ faz o chamado "Channel shortening" (elimina o *pre-cursor*).

$\rightarrow$ $B(z)$ cancela a ISI remanescente (*post-cursor*).

Transmissão em Blocos

Modelo discreto de canal:



$$
\begin{bmatrix} y(n) \\ y(n-1) \\ y(n-2) \\ y(n-3) \\ y(n-4) \\ y(n-5) \end{bmatrix}
=
\begin{bmatrix}
h(0) & h(1) & h(2) & & & & & \\
& h(0) & h(1) & h(2) & & & & \\
& & h(0) & h(1) & h(2) & & & \\
& & & h(0) & h(1) & h(2) & & \\
& & & & h(0) & h(1) & h(2) & \\
& & & & & h(0) & h(1) & h(2)
\end{bmatrix}
\begin{bmatrix} x(n) \\ x(n-1) \\ x(n-2) \\ x(n-3) \\ x(n-4) \\ x(n-5) \\ x(n-6) \\ x(n-7) \end{bmatrix}
+ \boldsymbol{v}(n)
$$

- Suponha que $v(n) = 0$. Bloco de saída genérico:

$$
\begin{bmatrix} y(Mn) \\ y(Mn-1) \\ \vdots \\ y(Mn-P+1) \end{bmatrix} = \begin{bmatrix} h(0) & \cdots & h(N-1) & 0 & \cdots & 0 \\ 0 & \cdots & h(N-2) & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & h(0) & \cdots & h(N-1) \end{bmatrix} \begin{bmatrix} x(Mn) \\ x(Mn-1) \\ \vdots \\ x(Mn-P+1) \\ \hline x(Mn-P) \\ \vdots \\ x(Mn-P-N+2) \end{bmatrix}
$$

A saída $\boldsymbol{y}_n$ pode escrita como

$$
\boldsymbol{y}_n \quad = \quad \begin{bmatrix} \mathcal{H}_0 & \mathcal{H}_1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_n \\ \boldsymbol{x}_{n-1} \end{bmatrix}
$$

ou

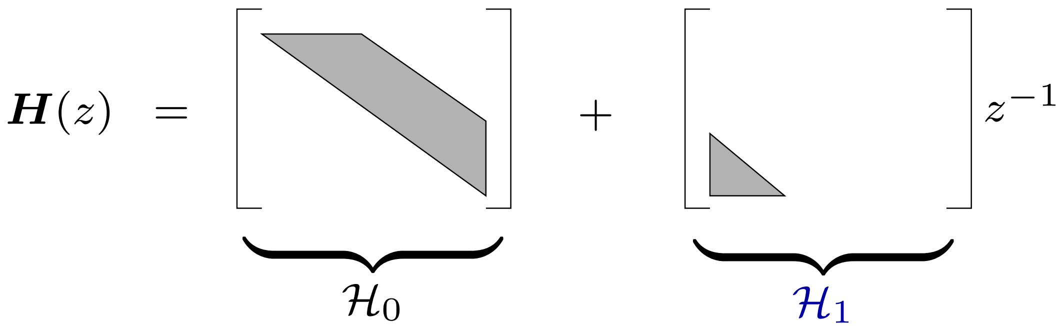
$$
\boldsymbol{y}_n \quad = \quad \mathcal{H}_o \boldsymbol{x}_n + \mathcal{H}_1 \boldsymbol{x}_{n-1}
$$

Tomando-se a transformada-z, podemos escrever

$$\boldsymbol{Y}(z) \;=\; \boldsymbol{H}(z)\boldsymbol{X}(z)$$

onde

$$\boldsymbol{H}(z) \;=\; \boldsymbol{H}_o(z) \;+\; z^{-1}\boldsymbol{H}_1(z)$$



$\mathcal{H}_0 \Longrightarrow$ Representa ISI

$\mathcal{H}_1 \Longrightarrow$ Representa IBI

Precodificadores

- IBI pode ser eliminada através da introdução de redundâncias, mas que diminui o *throughput* do sistema.

- Dentre as diversas formas de introdução de redundâncias, existem duas que tornam o modelo entrada-saída bastante simples:
  - Método *Zero Padding* ou *Transmit-Zeros* (TZ)

    $\hookrightarrow$ Overlap-and-Save
  - Método *Leading-zeros* (LZ)

    $\hookrightarrow$ Overlap-and-Add

Eliminação de IBI

Seja $\delta \in \{0, L-1\}$. Vamos definir 2 matrizes:

$$\begin{aligned} \boldsymbol{\mathcal{I}}_\delta &= \begin{bmatrix} \mathbf{0}_{\delta \times M} \\ \boldsymbol{I}_M \end{bmatrix} \\ \bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}} &= \begin{bmatrix} \boldsymbol{I}_{P-L+1+\delta} & \mathbf{0}_{(P-L+1+\delta)\times(L-1-\delta)} \end{bmatrix}, \end{aligned}$$

De forma que

$$\boldsymbol{x}_n = \boldsymbol{\mathcal{I}}_\delta \boldsymbol{s}_n , \qquad \boldsymbol{y}'_n = \bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}} \boldsymbol{y}_n$$

$\therefore$

$$\boldsymbol{y}'_n = \underbrace{(\bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}} \mathcal{H}_0 \boldsymbol{\mathcal{I}}_\delta)}_{\triangleq \mathbf{H}_o} \boldsymbol{s}_n + \underbrace{(\bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}} \mathcal{H}_1 \boldsymbol{\mathcal{I}}_\delta)}_{\equiv 0} \boldsymbol{s}_{n-1} + \bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}} \boldsymbol{v}_n$$

Modelo Equivalente

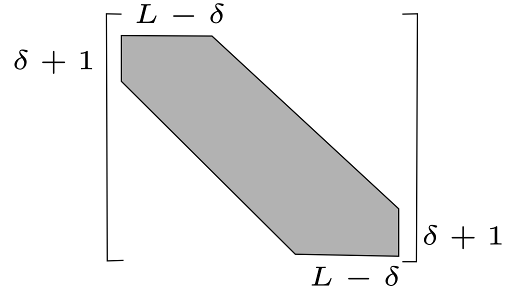
$$\boxed{\boldsymbol{y}_n' \;=\; \mathbf{H}_o \boldsymbol{s}_n \;+\; \boldsymbol{v}_n'}$$

onde

$$\mathbf{H}_o \;=\; \bar{\boldsymbol{\mathcal{I}}}_\delta^{\mathsf{T}} \cdot \mathcal{H}_0 \cdot \boldsymbol{\mathcal{I}}_\delta \qquad (M + 2\delta - L + 1) \times M$$

$\mathbf{H}_o$ é do tipo Toeplitz:

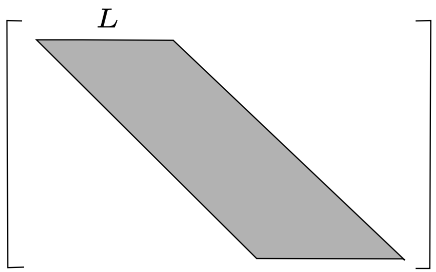
$$\mathbf{H}_o \quad =$$



Casos Extremos

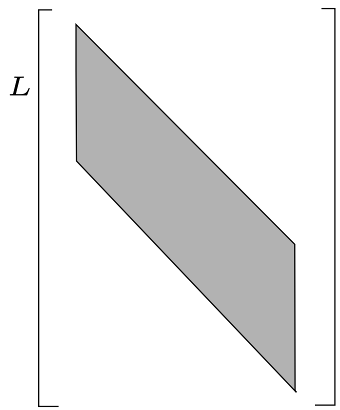
- $\delta = 0$ (Leading-Zeros)

$$\mathbf{H}_o =$$



- $\delta = L - 1$ (Transmit-Zeros)

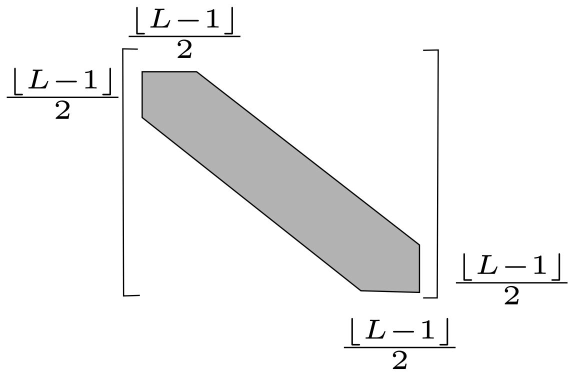
$$\mathbf{H}_o =$$



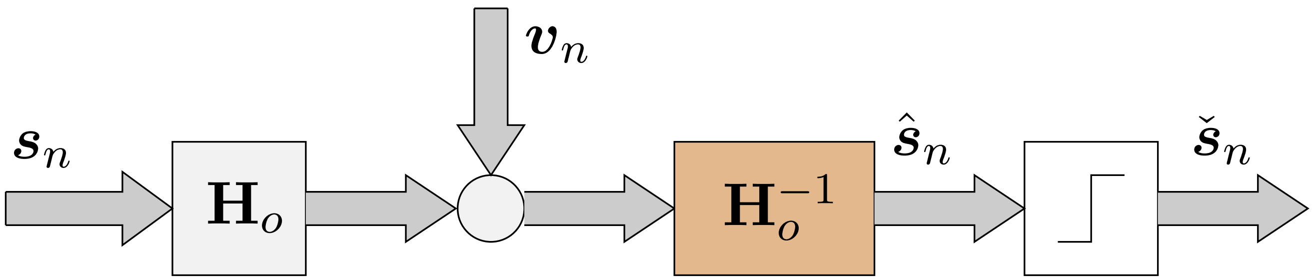
Caso de Redundância Mínima

- $\delta = \frac{\lfloor L-1 \rfloor}{2}$

$$\mathbf{H}_o =$$

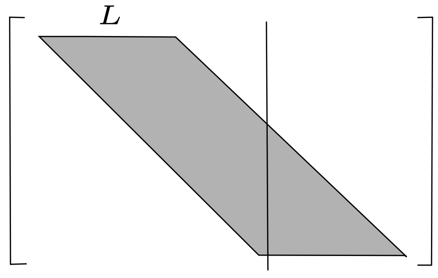


– Detecção :



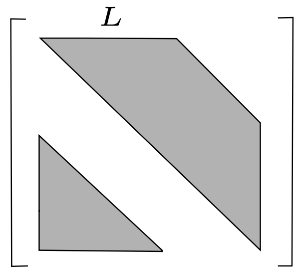
Método Overlapp-and-Save (LZ + Prefixo Cíclico)

Para $\delta = 0$, observe que



$$\left[\begin{array}{cc} \multicolumn{2}{c}{\boldsymbol{I}_M} \\ \hline \boldsymbol{I}_{L-1} & \mathbf{0} \end{array}\right]$$

$=$



$\triangleq \quad \mathbf{C}$

$\Longrightarrow$ $\mathbf{C}$ é *Circulante*, e portanto diagonalizada pela matriz DFT:

$$
\mathbf{C} = \begin{bmatrix}
h(0) & h(1) & \cdots & h(N-1) & 0 & \cdots & 0 \\
0 & h(0) & h(1) & \ddots & h(N-1) & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\
0 & \ddots & \ddots & \ddots & \ddots & \ddots & h(N-1) \\
h(N-1) & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & & & \ddots & h(1) \\
h(1) & \cdots & h(N-1) & 0 & 0 & \cdots & h(0)
\end{bmatrix}
$$

$$
\boxed{\mathbf{C} = \boldsymbol{F}^{*} \boldsymbol{\Lambda} \boldsymbol{F}}
$$

onde $\boldsymbol{\Lambda} \triangleq \mathrm{diag}(\boldsymbol{\lambda})$,

$$
\boldsymbol{\lambda} = \sqrt{M} \boldsymbol{F} \begin{bmatrix} \boldsymbol{h} \\ \mathbf{0}_{M-L,1} \end{bmatrix}
$$

$$
\boldsymbol{h} = \begin{bmatrix} h(0) & h(1) & \cdots & h(N-1) \end{bmatrix}^{\mathsf{T}}
$$

Modelo Equivalente Circulante

Assim, ao invés de transmitirmos $\boldsymbol{s}_n$, transmitimos

$$
\begin{aligned}
\boldsymbol{u}_n &\triangleq \begin{bmatrix} \boldsymbol{I}_M \\ \begin{bmatrix} \boldsymbol{I}_{L-1} & \mathbf{0}_{L-1,M-L+1} \end{bmatrix} \end{bmatrix} \boldsymbol{s}_n \\
&= \begin{bmatrix} \boldsymbol{s}_n \\ \boldsymbol{s}_n(0:L-2) \end{bmatrix}
\end{aligned}
$$

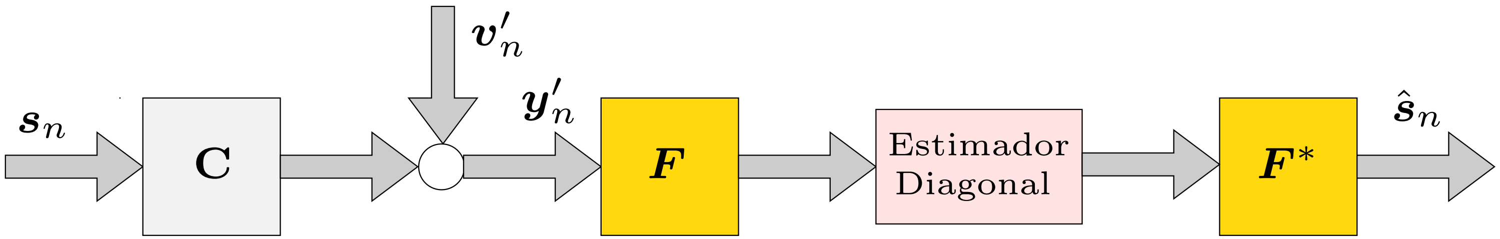
de forma que o modelo equivalente de canal se torna

$$\boxed{\boldsymbol{y}'_n = \mathbf{C}\boldsymbol{s}_n + \boldsymbol{v}'_n}$$

Equalização

Dado o modelo circulante, podemos ignorar o ruído de canal, e tentar recuperar $\boldsymbol{s}_n$ simplesmente invertendo o modelo:

$$\boxed{\hat{\boldsymbol{s}}_n \;=\; \mathbf{C}^{-1}\boldsymbol{y}'_n \;=\; \boldsymbol{F}^*\boldsymbol{\Lambda}^{-1}\boldsymbol{F}\,\boldsymbol{y}'_n}$$



OFDM — Orthogonal-Frequency-Division-Multiplexing

No padrão OFDM, de forma a diagonalizar a matriz de canal, ao invés de transmitirmos $\boldsymbol{s}_n$, antes de adicionar o prefixo cíclico, transmitimos

$$\bar{\boldsymbol{s}}_n = \boldsymbol{F}^* \boldsymbol{s}_n$$

e no receptor, após descartarmos as amostras correspondentes, observamos

$$\boldsymbol{y}_n'' = \boldsymbol{F} \boldsymbol{y}_n'$$

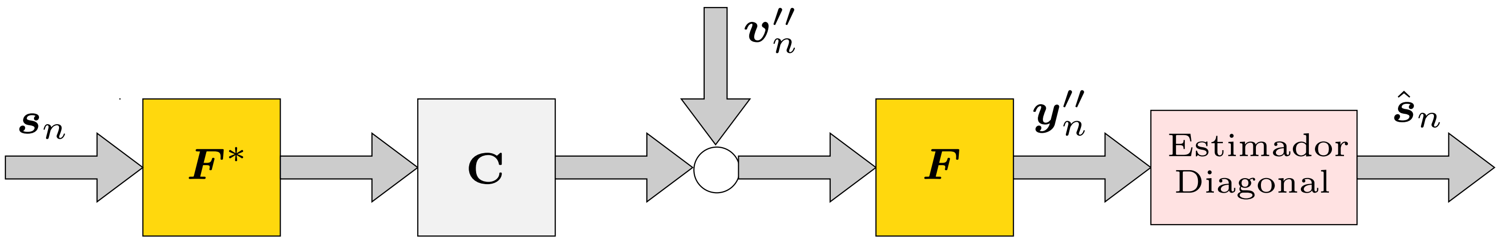
Desta forma, o modelo equivalente se torna

$$\boxed{\boldsymbol{y}_n'' = \boldsymbol{\Lambda} \boldsymbol{s}_n + \boldsymbol{v}_n''}$$

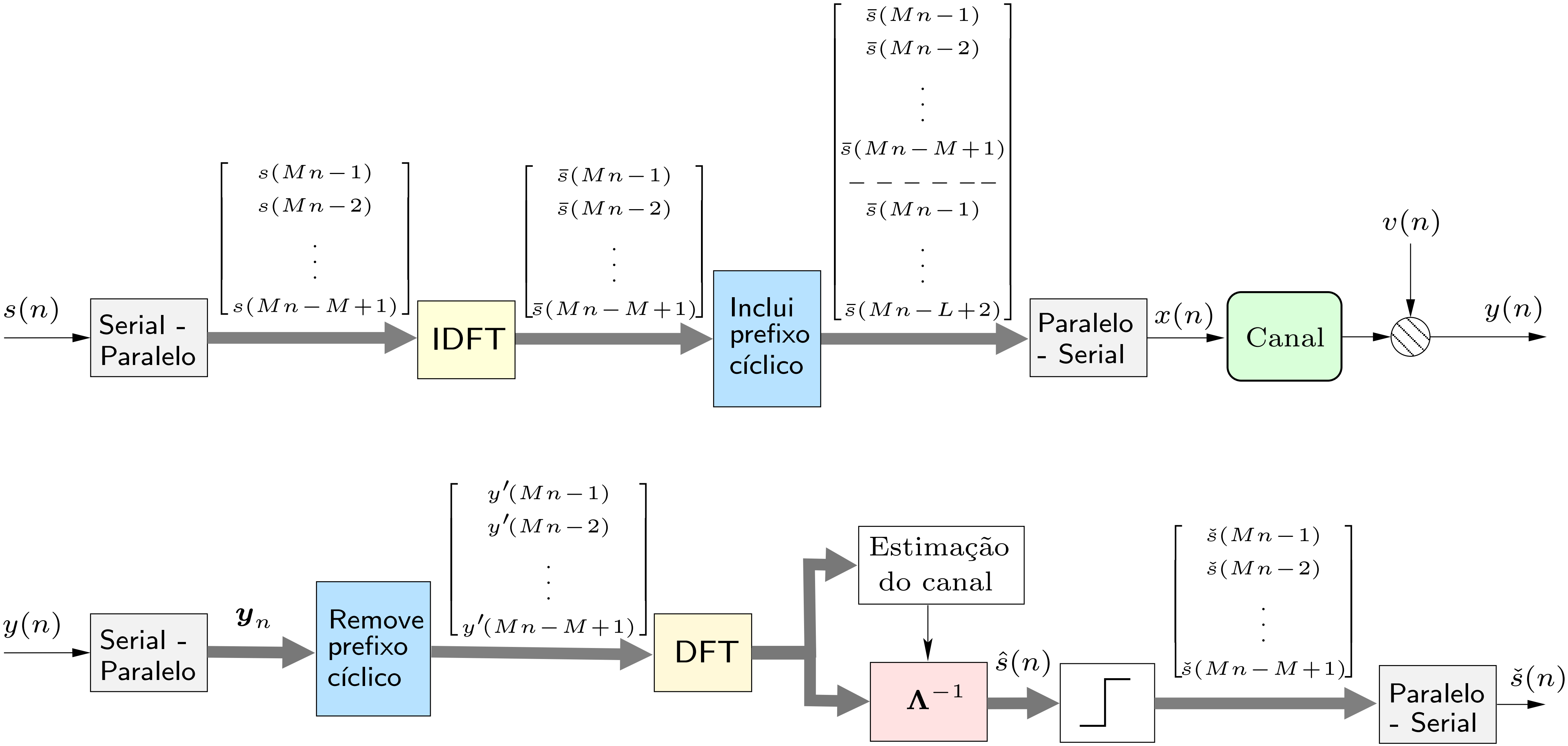
e o estimador de $\boldsymbol{s}_n$ fica

$$\boxed{\hat{\boldsymbol{s}}_n = \boldsymbol{\Lambda}^{-1} \boldsymbol{y}_n''}$$

Equalização OFDM



Esquema OFDM completo:



Método Overlapp-and-Add (TZ + Sufixo Cíclico)

Para fazer em casa...