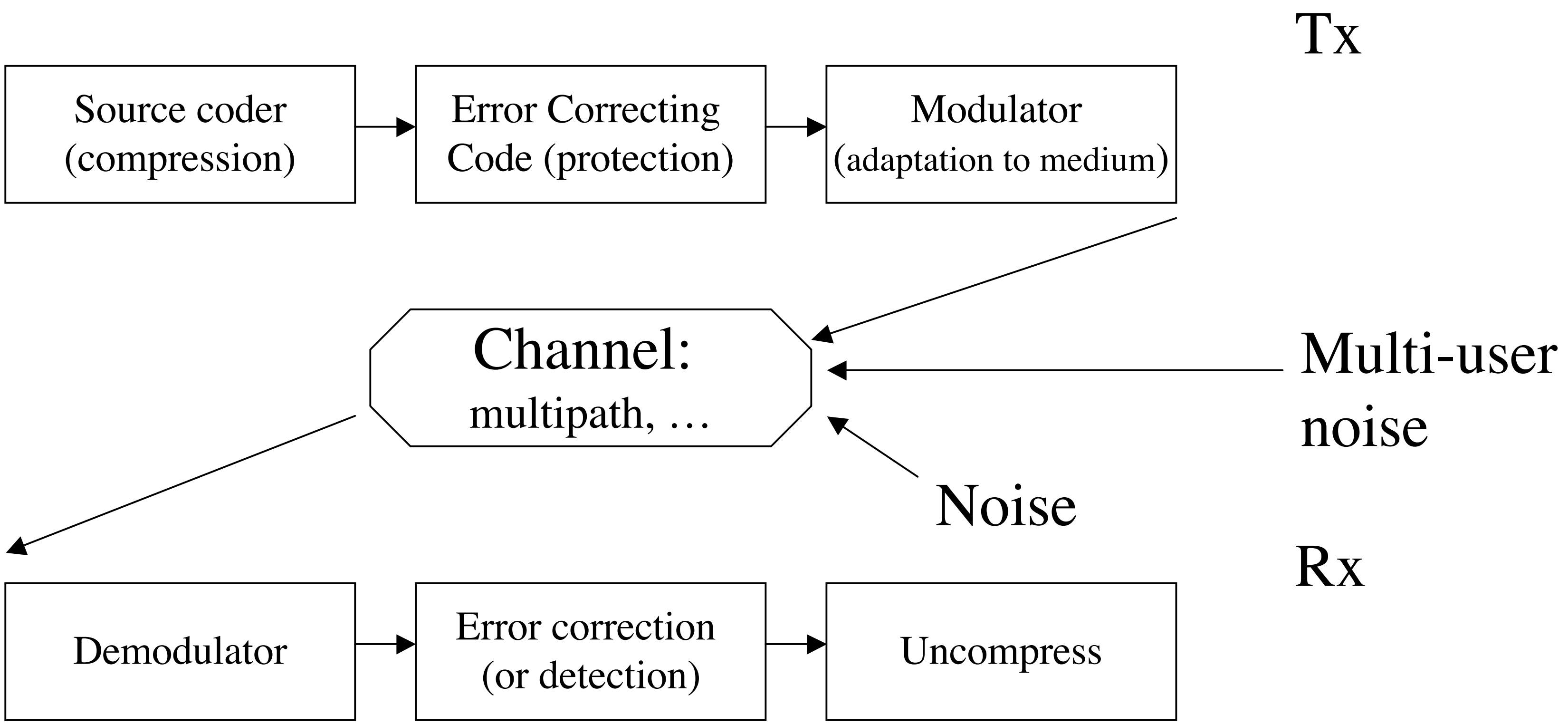
Transmissions numériques avancées

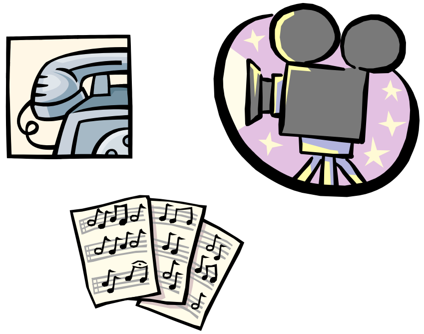
Jean-Marc Brossier

Overview


Tx
Source coder
(compression)
Error Correcting
Code (protection)
Modulator
(adaptation to medium)
Channel:
multipath, …
Multi-user
noise
Noise
Rx
Demodulator
Error correction
(or detection)
Uncompress

Source coding (1)

- (Analog to digital conversion)

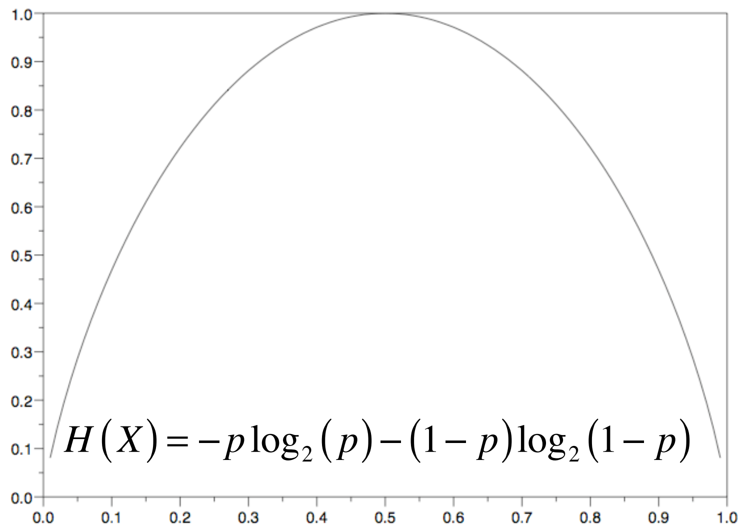


Source message:
0 1 1 0 1 1 1 1 1 …

- Source coding :
  - With loss: mp3, mpeg, divx, xvid (mpeg4), …
  - Lossless: Huffman, Fano-shannon, Lempel-Ziv,

Source coding (2)
Lossless compression : principles

X r.v. with two states: $\{0,1\}$ and $p(X=1)=p=1-p(X=0)$



Shannon Entropy (bits) *vs.* p

1. The source should be rewritten to achieve uniform probability : Huffman, Fano-shannon, …
2. Remove dependence : e n t r o p - e n t r o p y : how do you know "y" is missing?

Channel coding (1)

Transmitter: Receiver:

C O D I N G T H E O R Y C P D I N D T O E O R Y

Redundancy at the transmitter side :

C O D I N G T H E O R Y

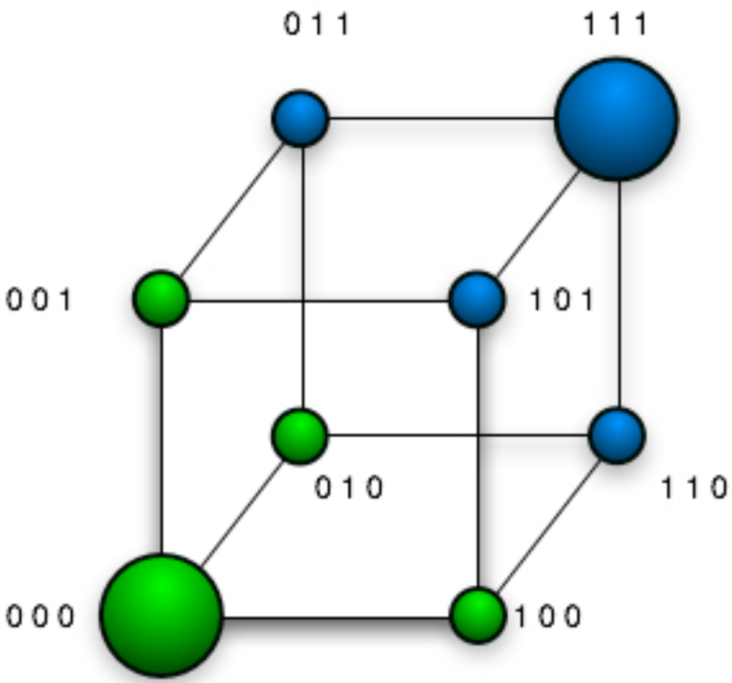
CCC OOO DDD III NNN GGG TTT HHH EEE OOO RRR YYY

CCC OPO DDD III NNN GGD TTT OHO EEE OOO RRR YYY

C O D I N G T O E O R Y

2 corrected, 1 detected

Channel coding (2)



- Tx:
  - 2 code words (length *n=3):* *(000) (111)*
  - *k=1* bit
- Rx:
  - (001),(010),(100): Detection + correction (000)
  - (110),(101),(011) Detection + correction (111)
  - *(000) (111)* *(Probably) right*

Modulation

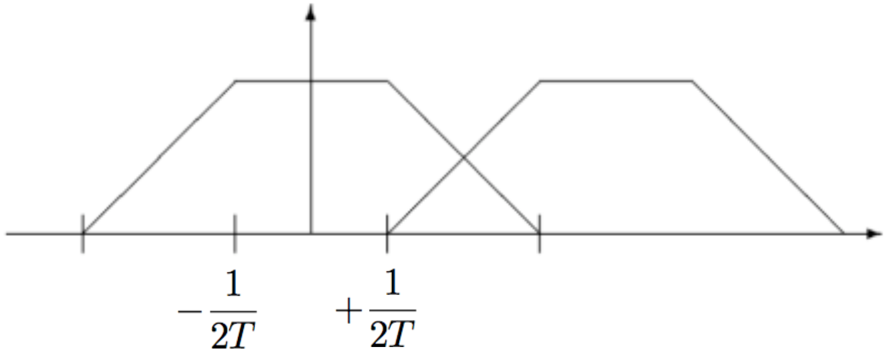
- Modulation: mapping from digital words to continuous time functions
- How many orthogonal functions with support [$0,T$]?
  - Infinity
  - BT with bandwidth B
- *Example: $B=1/T$ if a single function is used.*

Band-limited AWGN channels

- Two facts:
  - Perfection ($g(t) = \delta_0$) means infinite bandwidth
  - Requirement for zero ISI : $g_k = \delta_k$
- The aim:
  - Perfect discrete channel based on perfect finite bandwidth channel

$$g(t)\sum \delta(t-kT) = \delta(t) \leftrightarrow G(\nu) * \frac{1}{T}\sum \delta(\nu - \frac{k}{T}) = 1 \longrightarrow \frac{1}{T}\sum G(\nu - \frac{k}{T}) = 1$$

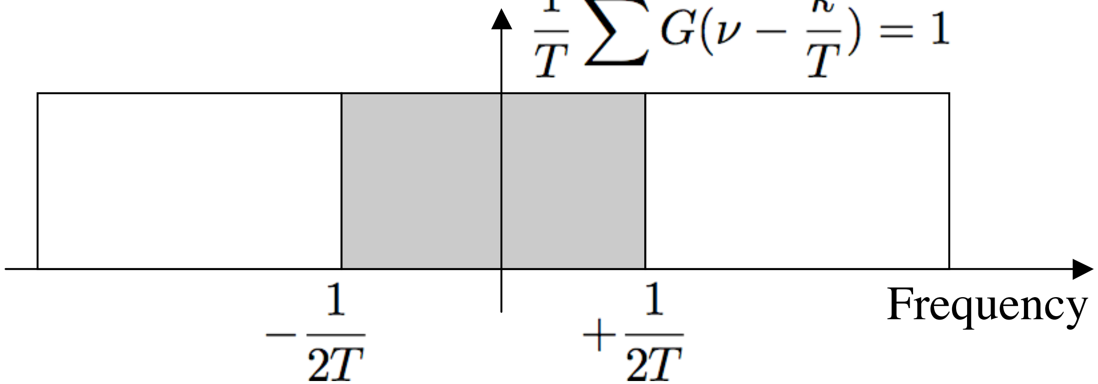
Nyquist criterion:

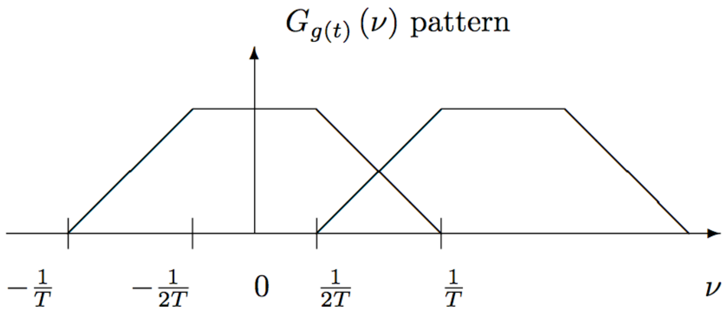


Band-limited AWGN channels

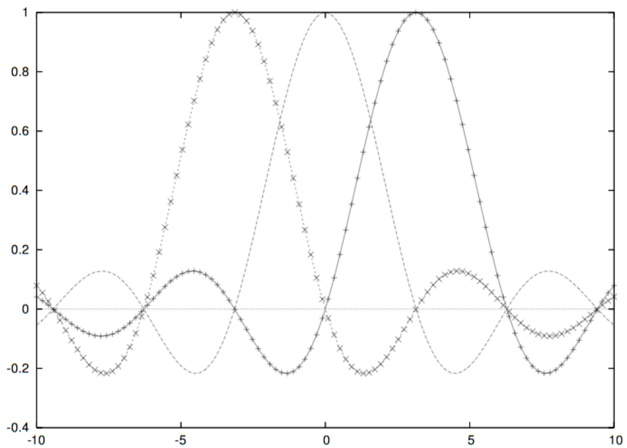
Ideal sinc solution

$$\frac{1}{T}\sum G(\nu - \frac{k}{T}) = 1$$





$[-\frac{1}{2T}, +\frac{1}{2T}]$ is the minimum bandwidth for interference free transmission

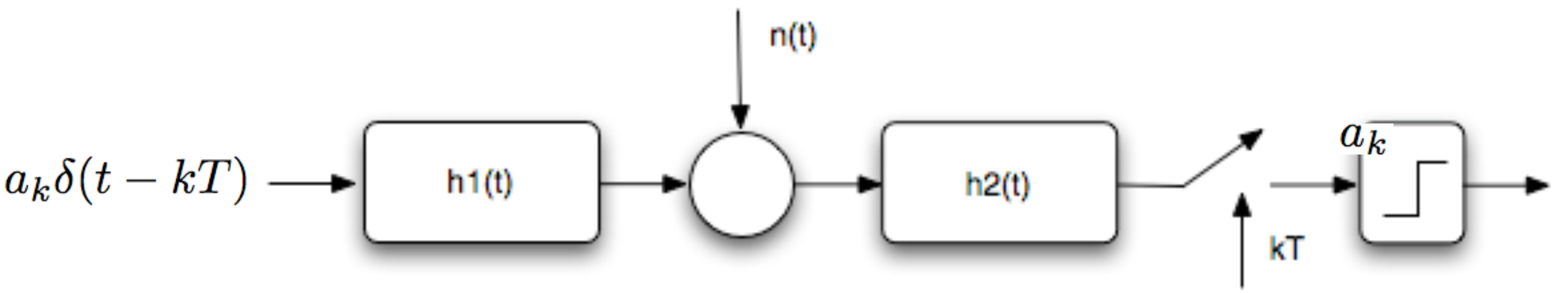


Strictly rectangular spectrum hard to implement:

- slowly decaying, infinitely long, noncausal waveform
- sensitivity to timing errors
- truncation errors (Gibbs phenomenon)

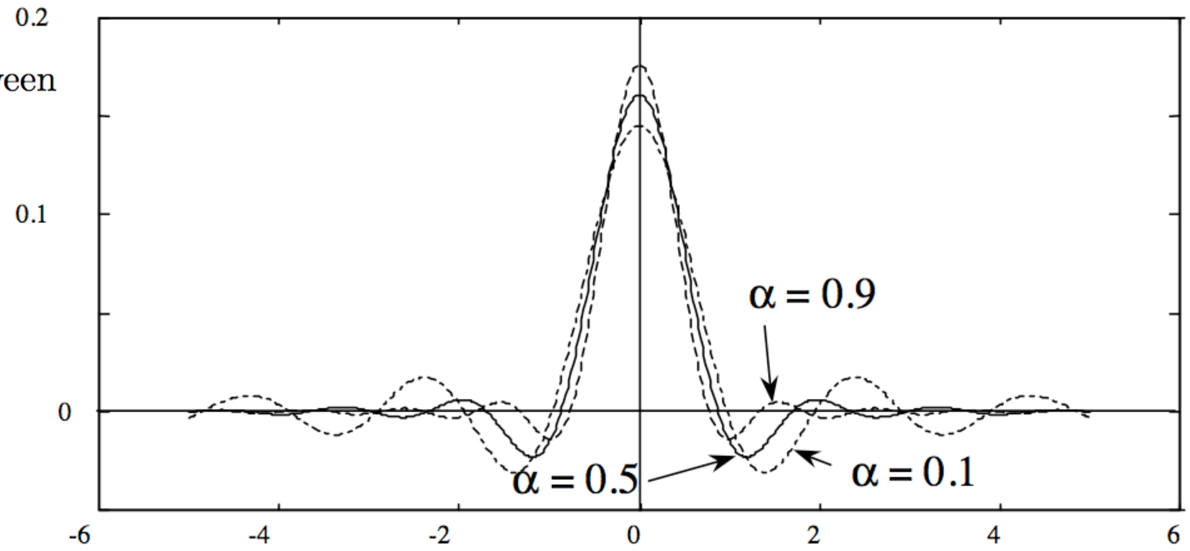
$$y_k = g_0 a_k + \sum_{n \neq k} a_n g_{k-n} + w_k$$

Band-limited AWGN channels
Root-Nyquist filtering



$h_1(t) * h_2(t) = h(t)$ must satisfy Nyquist
→ Best solution : a root-Nyquist filter at each side

The roll-off factor α tunes a tradeoff between bandwidth and decrease rate.



Channel modeling

- Discrete time baseband model.
- Additive white gaussian noise (AWGN)
- Deterministic linear channel
- Multipath channels
- Flat fading
  - Rayleigh: NLOS
  - Rice: LOS
  - Doppler Spectrum
- Coherence (Time - Frequency)
- Problems and solutions

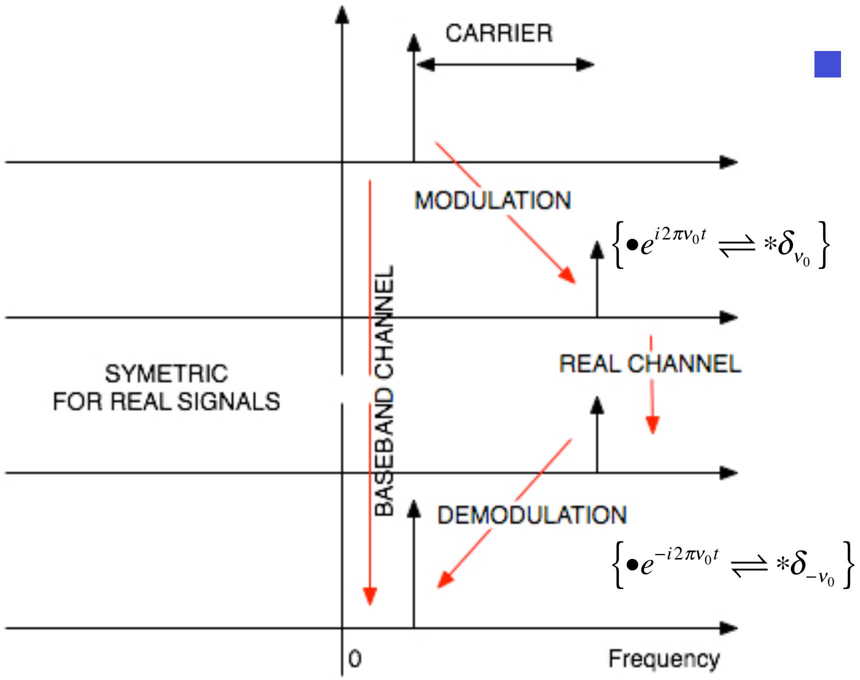
Real world channels

- Continuous time
  - Modulator-demodulator (modem)
- Noise:
  - error correction: algebraic, iterative
- Finite bandwidth:
  - Nyquist criterion
- Non ideal frequency response:
  - Equalization (linear, DFE, Viterbi, OFDM)
- Random or deterministic
  - Diversity
- Multiuser
  - OFDM, CDMA, MUD

Baseband model

- If a system is linear:
  - Superposition: convolution:
    $\exp[i\,\bullet]$ are eigenfunctions
- Full system knowledge:
  - What happens to an eigenfunction: eigenvalue: complex gain.

Baseband model



- Baseband channel:
  - Complex impulse response
  - Image of the real channel (at carrier frequency)

Translation $\rightarrow$

$$s(t)e^{i2\pi\nu_0 t} \rightleftharpoons S(\nu) * \delta_{\nu_0} = S(\nu - \nu_0)$$

$$s_{\mathrm{HF}}(t) = \mathrm{Re}\left[\overbrace{(s_I + is_Q)}^{s(t)} e^{i2\pi\nu_0 t}\right] = s_I \cos(2\pi\nu_0 t) - is_Q \sin(2\pi\nu_0 t) \rightleftharpoons \frac{1}{2}\left[S(\nu-\nu_0) + S^*(-\nu-\nu_0)\right]$$

Translation $\leftarrow$

$$\langle s_{HF} \mid \cos(2\pi\nu_0 t)\rangle \rightarrow s_I$$

$$\langle s_{HF} \mid \sin(2\pi\nu_0 t)\rangle \rightarrow s_Q$$

Baseband model

- Baseband signals:

Tx signal $$s_{\mathrm{HF}}(t) = \mathrm{Re}\left[s(t) e^{i2\pi\nu_0 t} \right] \rightleftharpoons \frac{1}{2}\left[S(\nu - \nu_0) + S^*(-\nu - \nu_0) \right]$$

Real channel $$h_{\mathrm{HF}}(t) = \mathrm{Re}\left[h(t) e^{i2\pi\nu_0 t} \right] \rightleftharpoons \frac{1}{2}\left[H(\nu - \nu_0) + H^*(-\nu - \nu_0) \right]$$

Rx signal $$r_{\mathrm{HF}}(t) = \mathrm{Re}\left[r(t) e^{i2\pi\nu_0 t} \right] \rightleftharpoons \frac{1}{2}\left[R(\nu - \nu_0) + R^*(-\nu - \nu_0) \right]$$

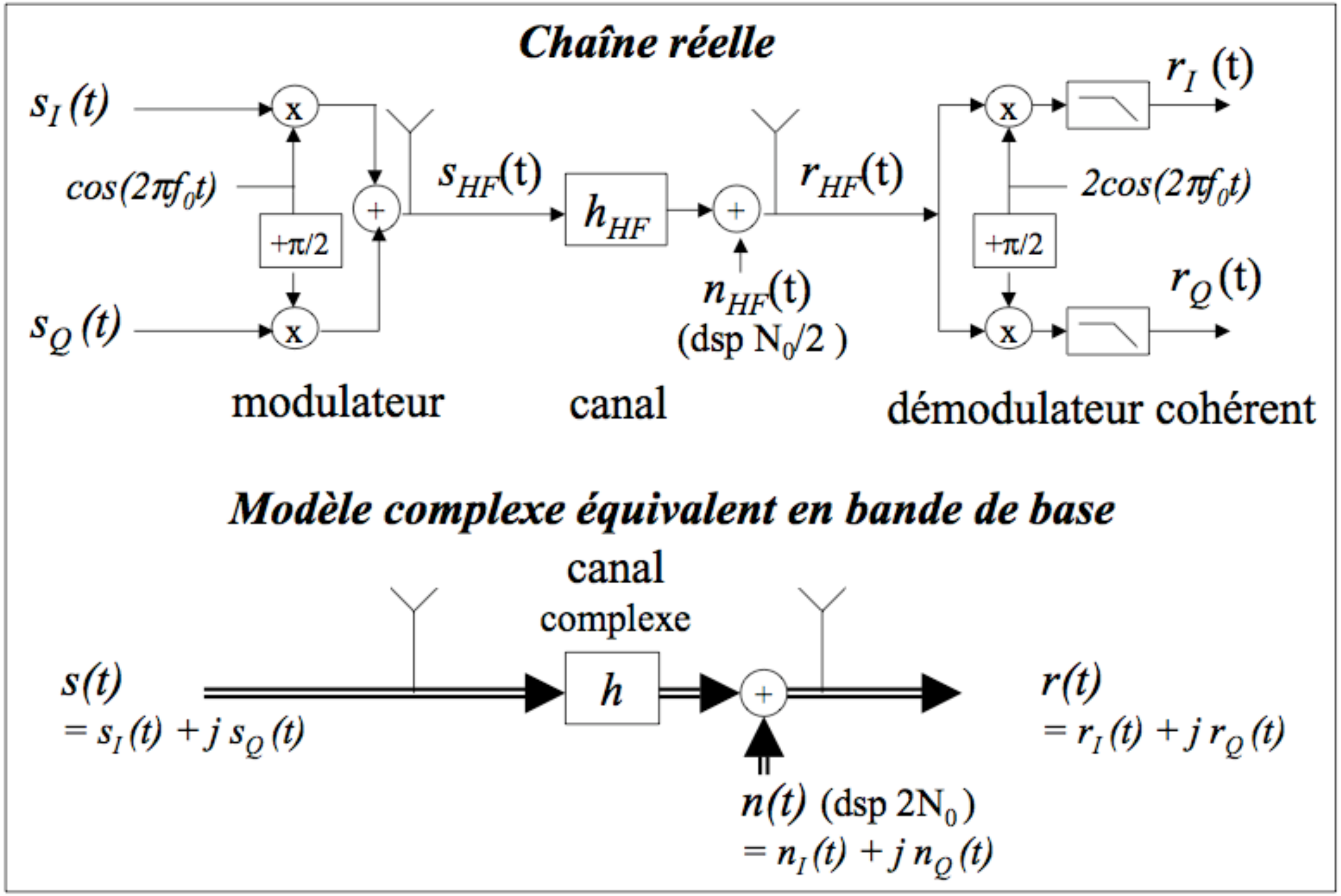
- Baseband channel:

$R(\nu) = H(\nu) S(\nu)$ $H(\nu)$ baseband frequency response

$r(t) = h(t) * s(t)$ $h(t)$ complex baseband impulse response

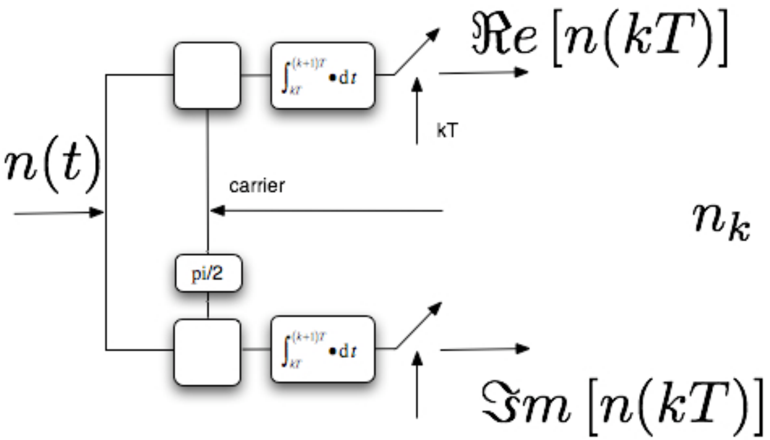
Discrete time baseband model

Baseband impulse response



Discrete time baseband model

Circular gaussian noise



If n(t) is a white gaussian noise, the discrete baseband noise

$$n_k = \int_{kT}^{(k+1)T} n(t) exp(i2\pi\nu_0 t) dt$$

is white, gaussian and circular:

1. Same power on each component

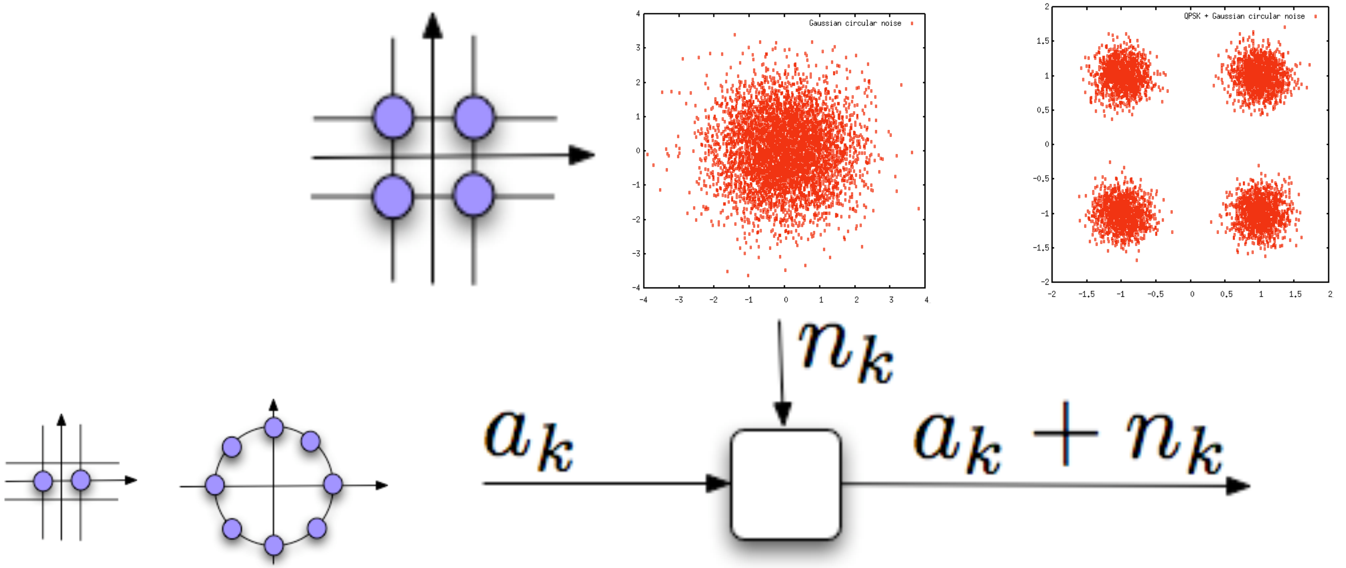
$$E(n_k^{\mathrm{Re}})^2 = E(n_k^{\mathrm{Im}})^2 = \frac{\sigma^2}{2}$$

2. Uncorrelated components

$$E(n_k^{\mathrm{Re}} n_l^{\mathrm{Im}}) = \delta_{kl}$$

Channel modeling

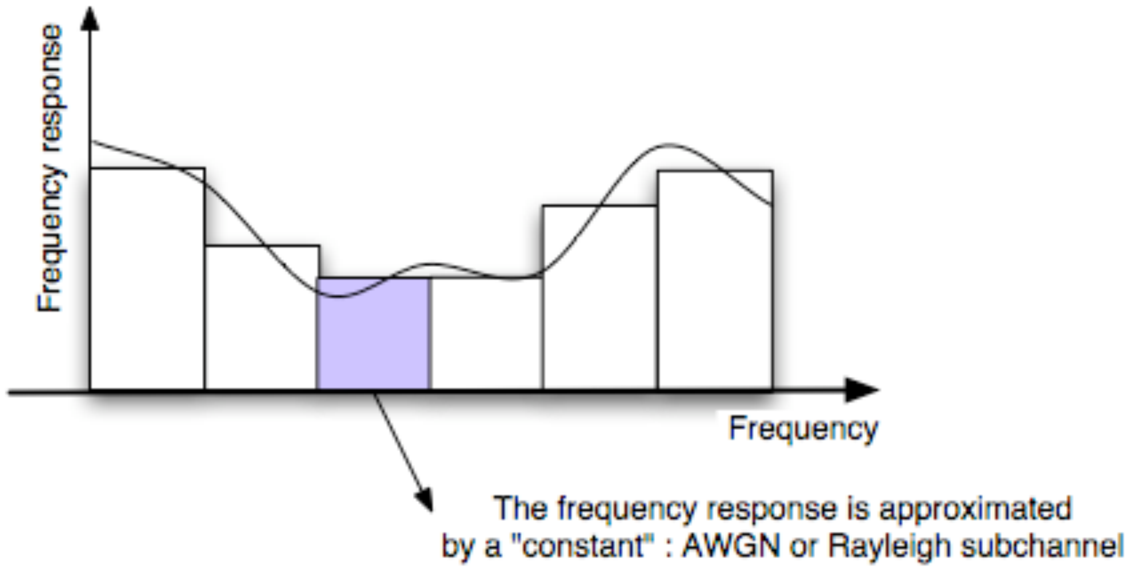
Additive White Gaussian Noise (1)



- a_k belongs to a constellation (complex plan): BPSK, PSK, QAM.
- n_k white gaussian circular noise
- flat frequency response within the Nyquist band

Channel modeling

Additive White Gaussian Noise (2)

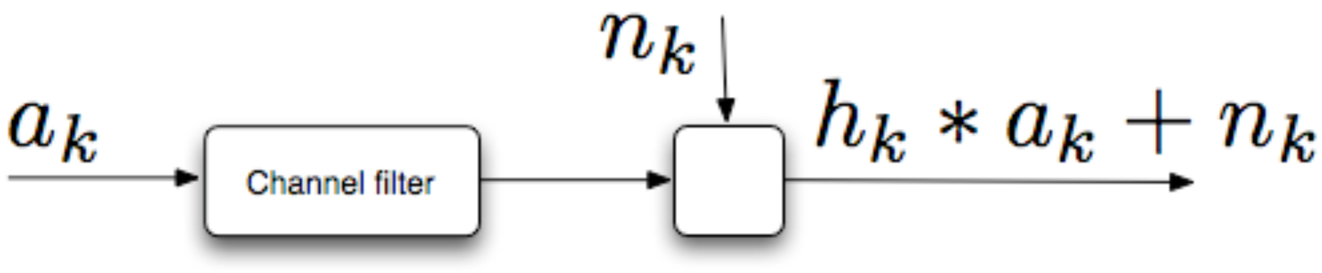


Applications:

- Sub carriers of some multicarrier systems
- Low speed transmissions, ….

Channel modeling

Deterministic linear channel



- a_k belongs to a constellation (complex plan): BPSK, PSK, QAM.
- n_k white gaussian circular noise
- non flat frequency response within the Nyquist band (channel baseband equivalent discrete impulse response h_k, frequency response $H(\nu)$)

Applications:

- xDSL (Digital Subscriber Line) systems, cable modems, ….

Channel modeling

Multipath channels (1)

Number of paths

Amplitude

Delay

Additive (gaussian) noise

$$r_{HF}(t) = \sum_{i=1}^{P} \rho_p s_{HF}(t - \tau_p) + n_{HF}(t)$$

$$s_{HF}(t) = \Re\left[s(t)e^{i2\pi\nu_0 t}\right] \Rightarrow r_{HF}(t) = \Re e \sum_{i=1}^{P} \rho_p e^{-i2\pi\nu_0\tau_p} s(t-\tau_p)e^{i2\pi\nu_0 t} + n_{HF}(t)$$

HF impulse response:

LF equivalent impulse response:

$$h_{HF}(t) = \sum_{i=1}^{P} \rho_p \delta(t - \tau_p) \Rightarrow h(t) = \sum_{i=1}^{P} \rho_p e^{i\phi_p} \delta(t - \tau_p)$$

Channel modeling

Multipath channels (2)

- Parametric baseband representation:

$$h(t) = \sum_{i=1}^{P} \rho_p e^{i\phi_p} \delta(t - \tau_p) = \sum_{i=1}^{P} \alpha_p \delta(t - \tau_p)$$

Each path is characterized by 3 parameters:

- α_p complex amplitude (amplitude ρ_p and phase ϕ_p)
- τ_p delay

Applications:

• UMTS, WiFi (802.11b-g), ….

Fading scales

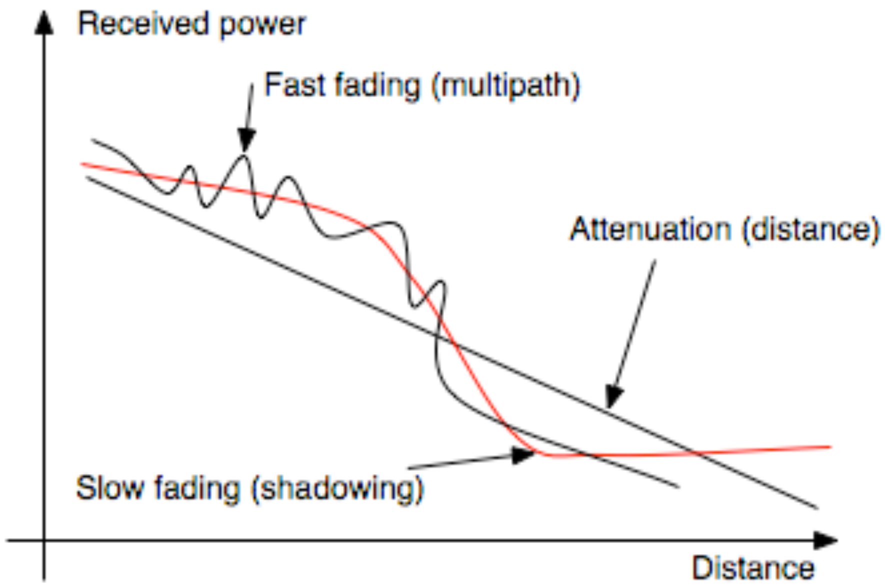
- Distance
  - outdoor, indoor

$1/d^n$ with $n \approx 3$ (indoor) $n \approx 4$ (outdoor)

- Slow fading
  - Log-normal

6-10dB, 5 (indoor)-20m (outdoor)

- Fast fading
  - Multipath propagation



Channel modeling
Flat Rayleigh fading: NLOS

- Path: cluster of micropaths:

$$\alpha_p(t) = \rho_p(t)e^{i\phi_p(t)} = \sum_n \rho_{p,n}(t)e^{i\phi_{p,n}(t)}$$

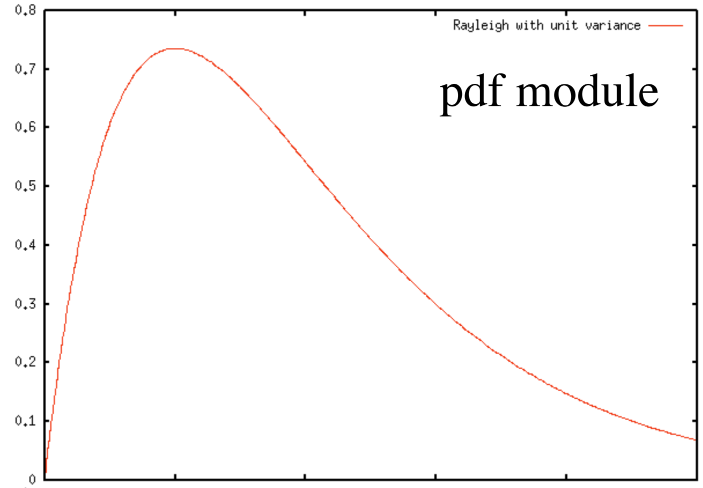
- NLOS (No Line of Sight, urban) : CLT:

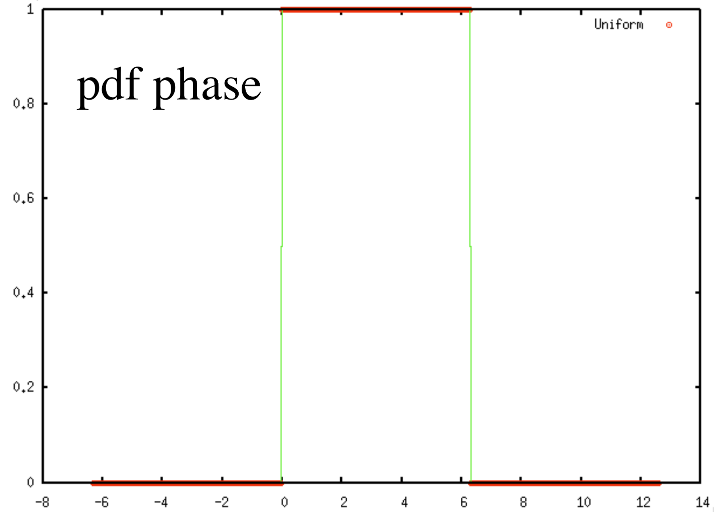
• $\Re e\,[\alpha_p(t)]$ and $\Im m\,[\alpha_p(t)]$ are uncorrelated gaussian with variance $\sigma^2_{\alpha_p}$

• The module $\rho_p(t)$ is Rayleigh :

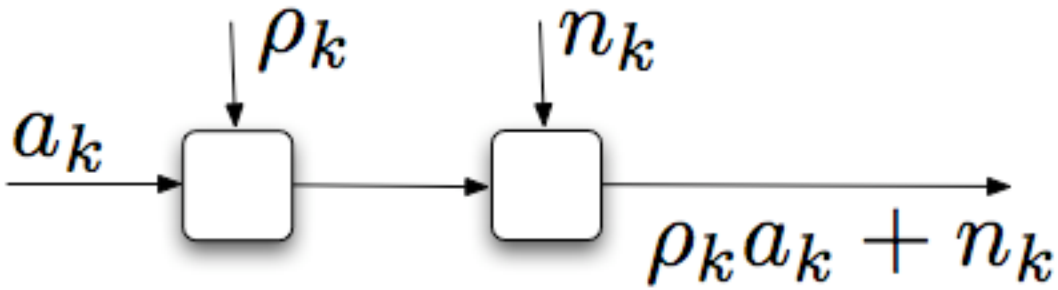
$$p(\rho) = \frac{\rho}{\sigma^2_\alpha} exp\left(-\frac{\rho^2}{2\sigma^2_\alpha}\right) \text{ for } \rho > 0$$

• The phase $\phi_p(t)$ is uniform over $[0, 2\pi)$





Channel modeling
Flat Rayleigh fading



- a_k belongs to a constellation (complex plan): BPSK, PSK, QAM.
- n_k white gaussian circular noise
- flat BUT TIME VARYING frequency response within the Nyquist band

Applications:

- Subcarriers of OFDM RF systems
  - DAB - Digital Audio Broadcast
  - DVB-T - Digital Audio Broadcast - , …
- Low speed transmissions, ….

Channel modeling

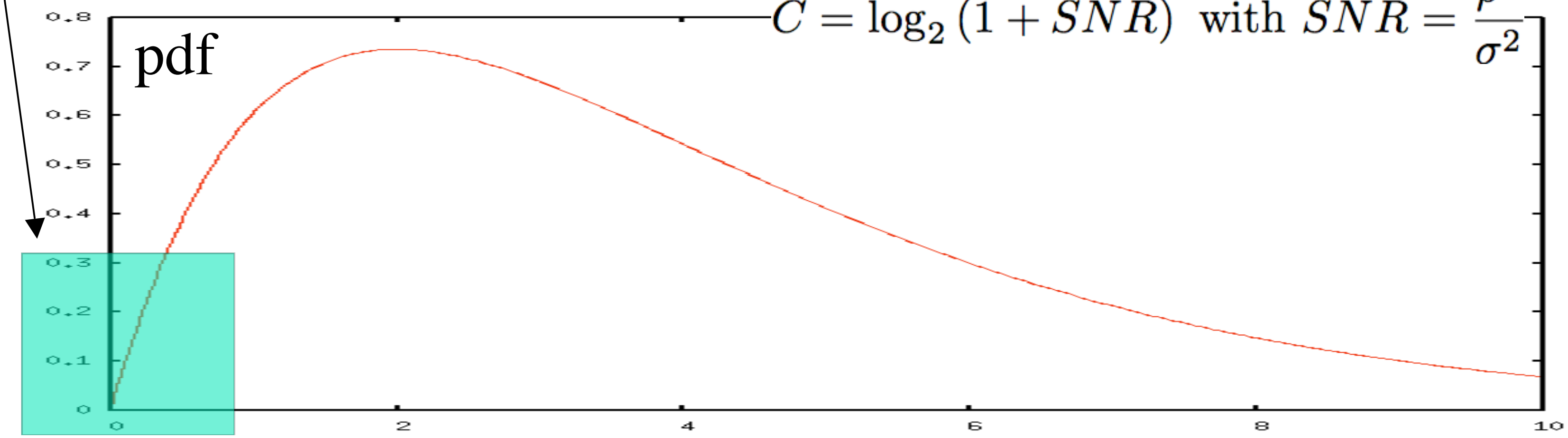
Capacity of a flat Rayleigh channel

- Ergodic channel
- Non ergodic channel:
  - Diversity needed : coding, interleaving, …
  - Outage probability

Random variable

Random capacity close to 0

$$C = \log_2(1 + SNR) \text{ with } SNR = \frac{\rho^2}{\sigma^2}$$



Channel modeling
Flat Rice fading: LOS

- A deterministic component is added:
  - Rice Model

Channel modeling

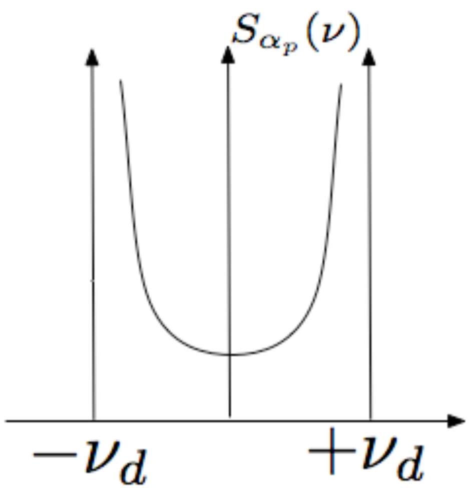
Jakes Doppler spectrum

- Correlation and Doppler spectrum:

$$R_{\alpha_p}(\lambda) = E\left[\alpha_p(t)\alpha_p^*(t-\lambda)\right] \leftrightarrow S_{\alpha_p}(\nu) = \mathcal{F}\left[R_{\alpha_p}(\lambda)\right]$$

- Jakes:

$$R_{\alpha_p}(\lambda) = \sigma_{\alpha_p}^2 J_0\left(2\pi\nu_d\lambda\right) \leftrightarrow S_{\alpha_p}(\nu) = \begin{cases} \dfrac{\sigma_{\alpha_p}^2}{\pi\nu_d\sqrt{1-\left(\frac{\nu}{\nu_d}\right)^2}} \text{ for } |\nu| \leq \nu_d \\ 0 \text{ for } |\nu| > \nu_d \end{cases}$$



Channel modeling

Coherence (time-frequency)

- Wide sense stationary (WSS)
- Uncorrelated scatterers (US)
  - Spaced-time spaced-frequency correlation function

$$R_H(\delta\nu, \delta t) = E\left[H(\nu, t)h^*(\nu - \delta\nu, t - \delta t)\right] \text{ with } h(\tau, t) \leftrightarrow H(\nu, t)$$

  - Coherence bandwidth

$$B_{coh} = \text{supp}\left\{R_H(\delta\nu, 0)\right\} \propto \text{ (temporal spread)}^{-1}$$

  - Coherence duration

$$T_{coh} = \text{supp}\left\{R_H(0, \delta t)\right\} \propto \text{ (Doppler spread)}^{-1}$$

Some useful links

- Wireless:

- http://grouper.ieee.org/groups/802/11/
  - IEEE 802.11TM WIRELESS LOCAL AREA NETWORKS - The Working Group for WLAN Standards
  - IEEE 802.16 Working Group on Broadband Wireless Access Standards
- http://www.3gpp.org/
  - The 3rd Generation Partnership Project Agreement.
  - Partners are ARIB, CCSA, ETSI, ATIS, TTA, and TTC.
- http://www.etsi.org/
  - The European Telecommunications Standards Institute (ETSI)

- Very-high-bit-rate Digital Subscriber Line:

- http://www.vdslalliance.com/
  - DMT (Discrete MultiTone) based VDSL

Problems and solutions

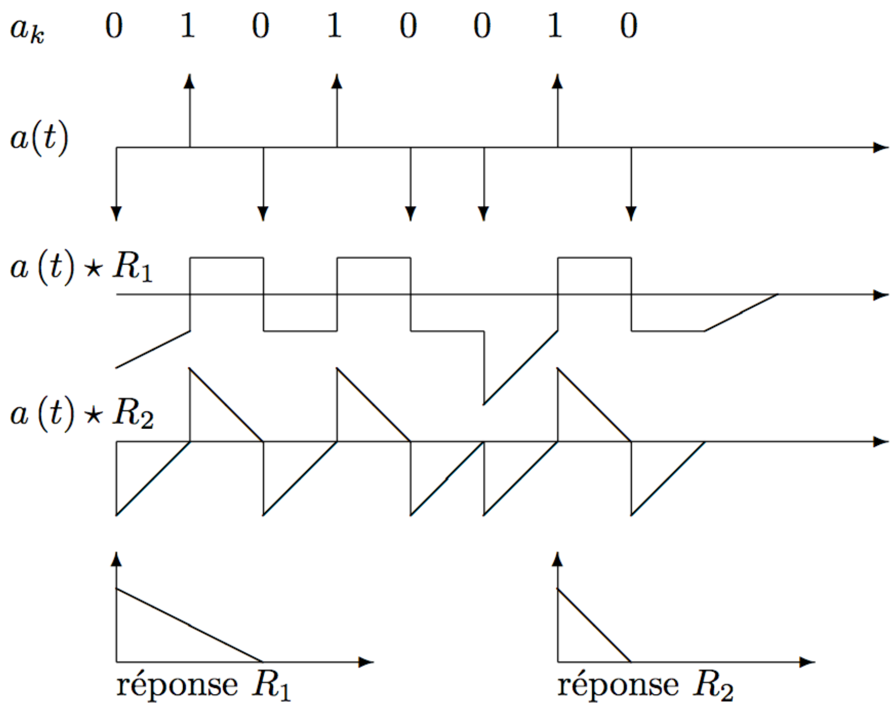
- Parameter estimation:
  - Timing recovery
  - Carrier synchronization
  - Channel estimation
- Channel coding :
  - protection against noise.
- Diversity:
  - protection against fading

Single carrier transmission (selective channel)

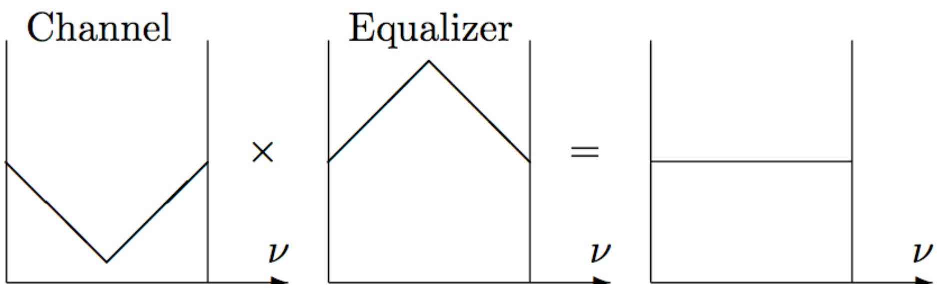
- Optimum receiver
- Structures:
  - Linear equalizers
  - ISI cancellers
  - Decision Feedback Equalizer (DFE)
- Adaptive equalizers
  - LMS
  - RLS
- Maximum Likelihood Sequence Estimation
- Channel estimation

Equalization: principles
ISI : InterSymbol Interference
Spectrum equalization

ISI : InterSymbol Interference

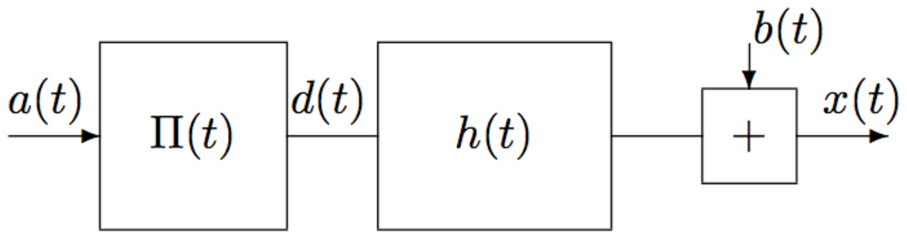


Spectrum equalization

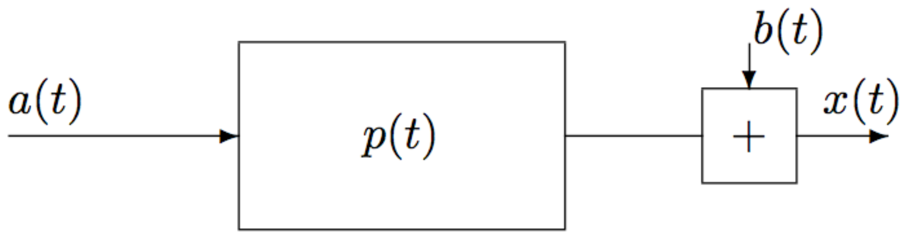


Signal path

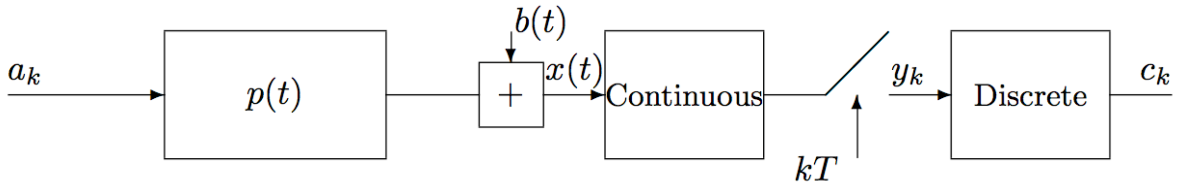
- From the source to the demodulator output

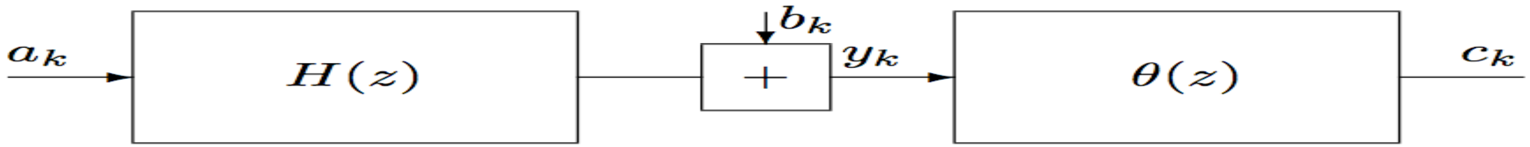


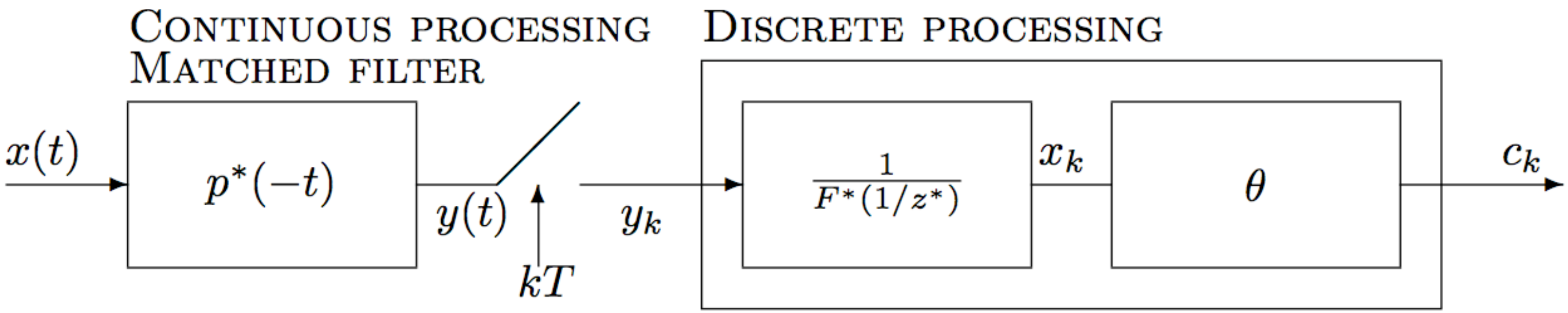
$$a(t) = \sum a_k \delta(t - kT) \qquad \Updownarrow \qquad x\left(t\right) = \sum_{n=-\infty}^{+\infty} a_n \; p\left(t - nT\right) + b\left(t\right)$$



Optimum receiver structure







$$y_k = \sum_{n=-\infty}^{+\infty} a_n g_{k-n} + w_k$$

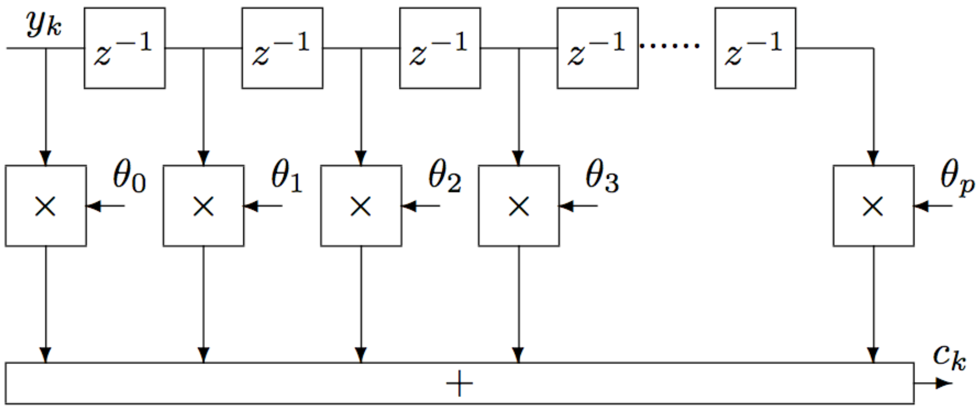
$$x(t) = \sum_{n=-\infty}^{+\infty} a_n \, p(t - nT) + b(t)$$

$$x_k = \sum_{n=-\infty}^{+\infty} a_n f_{k-n} + b_k$$

FIR Structure. Finite Impulse Response (a.k.a. MA - Moving average)

Linear structure

- The output is a linear combination of a finite set of inputs. $c_k = \sum_{n=N}^{n=M} \theta_n y_{k-n}$
- Stable



$\Theta = (\theta_N, \ldots, \theta_M)^T$ and
$Y_k = (y_{k-N}, \ldots, y_{k-M})^T$.

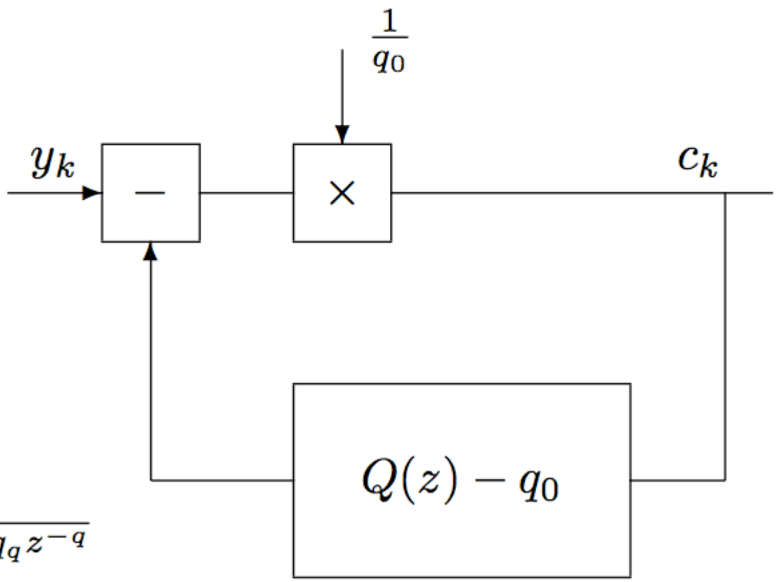
The output writes: $c_k = \Theta^T Y_k$

IIR structure. Infinite Impulse Response

Autoregressive (AR) structure.

Linear structure

- Inverse of a FIR filter: the output is a linear combination of a finite set of outputs
- Stable or NOT



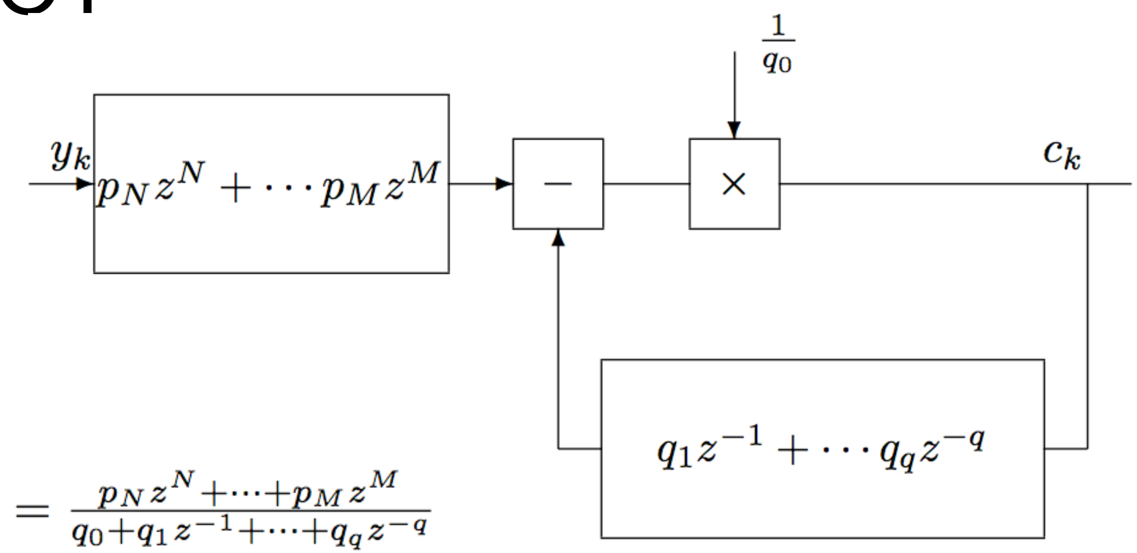
$$Q(z) = q_0 + q_1 z^{-1} + \cdots + q_q z^{-q}$$

$$c_k = \left[\frac{1}{Q(z)}\right] y_k \text{ with } \frac{1}{Q(z)} = \frac{1}{q_0 + q_1 z^{-1} + \cdots + q_q z^{-q}}$$

ARMA structure

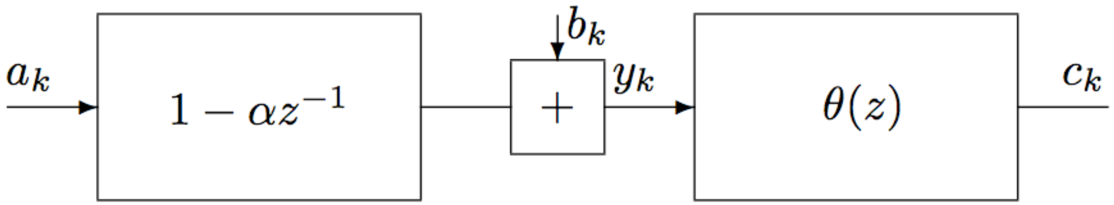
Linear structure

- The output depends on outputs and inputs
- Stable or NOT



$$c_k = \left[\frac{P(z)}{Q(z)}\right] y_k \text{ with } \frac{P(z)}{Q(z)} = \frac{p_N z^N + \cdots + p_M z^M}{q_0 + q_1 z^{-1} + \cdots + q_q z^{-q}}$$

Example: channel with 1 zero



- Minimum phase $1-\alpha z^{-1}$ avec $|\alpha|<1.$
  - Causal, stable inverse: MA or AR

$$1/\left(1-\alpha z^{-1}\right)=1+\alpha z^{-1}+\cdots\alpha^q z^{-q}+\cdots\rightarrow c_k=a_k+\alpha a_{k-1}+\cdots$$

$$c_k=\left[\frac{1}{1-\alpha z^{-1}}\right]y_k\rightarrow c_k=y_k+\alpha c_{k-1}$$

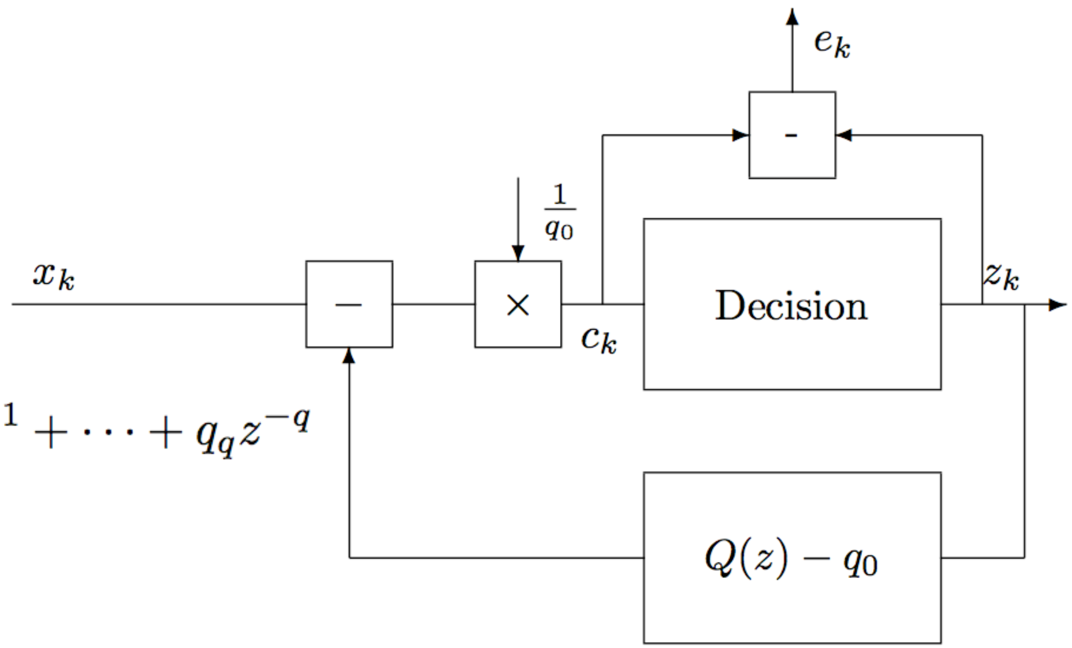
- Maximum phase $1-\alpha z^{-1}$ avec $|\alpha|>1.$
  - Non causal, stable inverse: MA

$$1/\left(1-\alpha z^{-1}\right)=-\alpha^{-1}z/\left(1-\alpha^{-1}z\right)$$

$$=-\alpha^{-1}z\left(1+\alpha^{-1}z+\cdots+\alpha^{-q}z^q+\cdots\right)$$

ISI canceller

- Remove contribution of undesired symbols

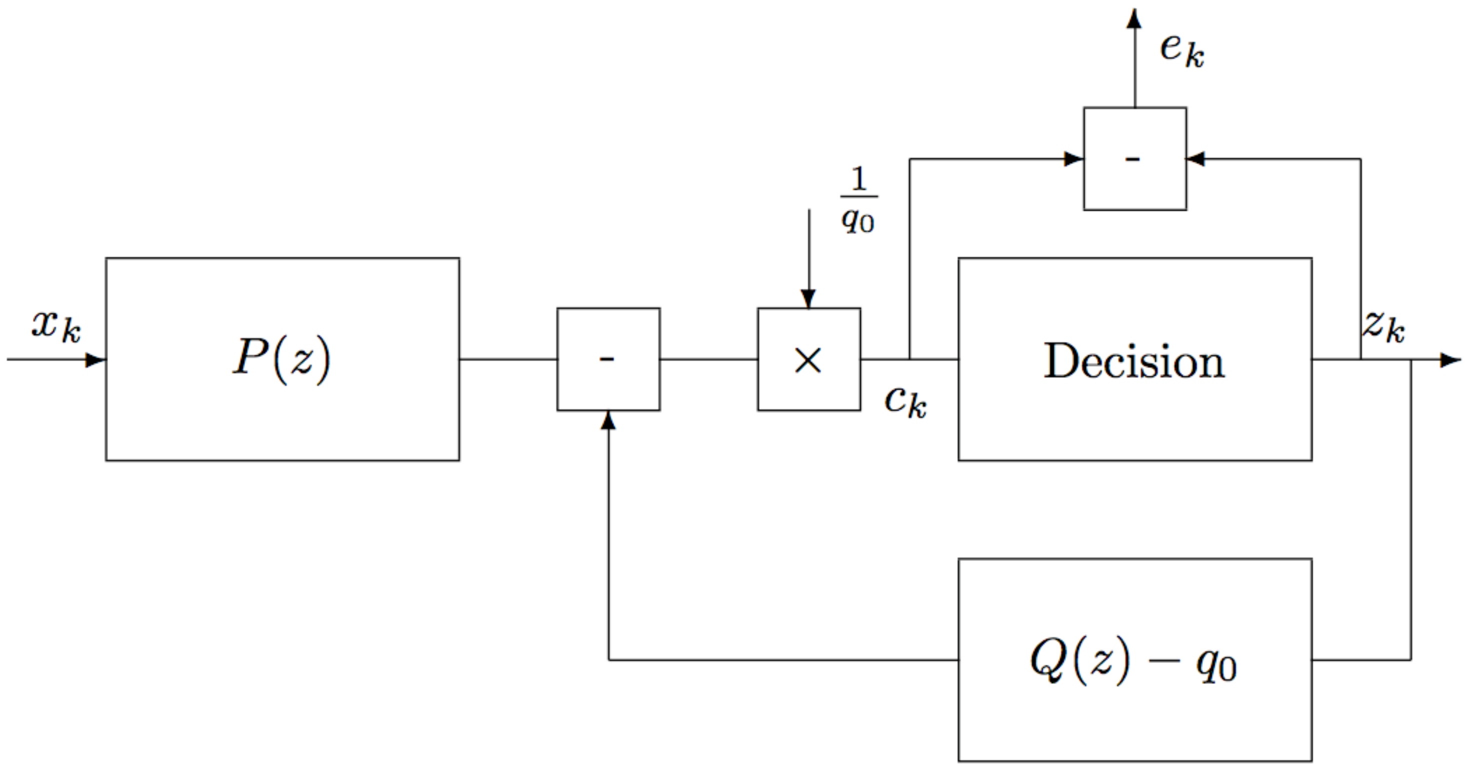


$$Q(z) = q_0 + q_1 z^{-1} + \cdots + q_q z^{-q}$$

Decision Feedback Equalizer (DFE)

Non linear structure

- Division "causal or not"



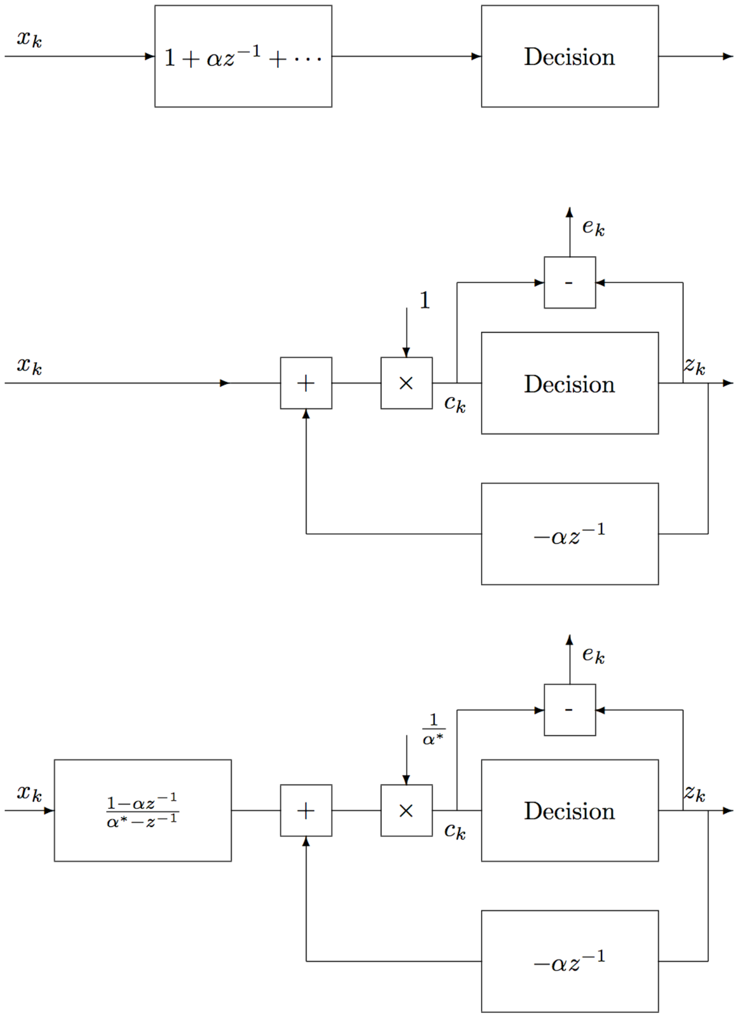
Comparison of 3 structures

Non linear structure

- Equalization of

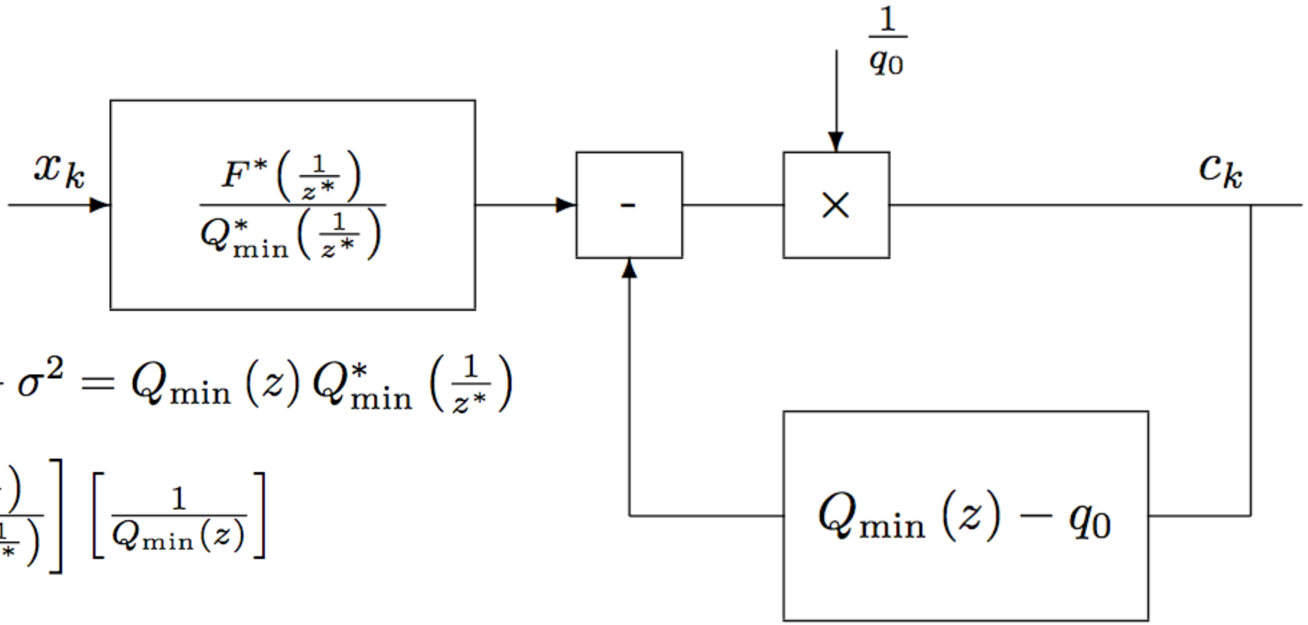
$$1 - \alpha z^{-1} \text{ avec } |\alpha| < 1.$$

$$1 - \alpha z^{-1} \text{ avec } |\alpha| > 1.$$



ARMA Wiener filter

- ARMA implementation of the optimal Wiener filter $J(\Theta) = E\left|e_k(\Theta)\right|^2$ with $e_k(\Theta) = a_k - c_k(\Theta)$



$$S_{xx}(z) = F(z)\,F^*\left(\frac{1}{z^*}\right) + \sigma^2 = Q_{\min}(z)\,Q^*_{\min}\left(\frac{1}{z^*}\right)$$

$$\rightarrow$$

$$\theta(z) = \frac{F^*\left(\frac{1}{z^*}\right)}{S_{xx}(z)} = \left[\frac{F^*\left(\frac{1}{z^*}\right)}{Q^*_{\min}\left(\frac{1}{z^*}\right)}\right]\left[\frac{1}{Q_{\min}(z)}\right]$$

Adaptive equalization:
FIR LMS - Least Mean Squares

- Objective function:

$$J(\Theta) = E\left|e_k(\Theta)\right|^2 \text{ with } e_k(\Theta) = a_k - c_k(\Theta)$$

- Minimization:

$$\nabla_\Theta J(\Theta) = -2E\left(Y_k^* e_k\right) = 0 \text{ with } Y_k^T = \left[y_k, y_{k-1}, \cdots, y_{k-p}\right]^T \text{ and } c_k(\Theta) = Y_k^T\Theta$$

- Gradient: $\Theta_k = \Theta_{k-1} - \mu_k \nabla_\Theta J(\Theta)\big|_{\Theta=\Theta_{k-1}}$
- Stochastic approximation:

$$\left\{\begin{array}{rcl} e_k & = & a_k - Y_k^T\Theta_{k-1} \\ \Theta_k & = & \Theta_{k-1} + \mu_k Y_k^* e_k \end{array}\right.$$

Adaptive equalization:

RLS - Recursive Least Squares

- Newton algorithm:

$$\Theta_k = \Theta_{k-1} - \mu_k \left\{ \left[\nabla^2_\Theta J(\Theta) \right]^{-1} \left[\nabla_\Theta J(\Theta) \right] \right\}_{\Theta=\Theta_{k-1}}$$

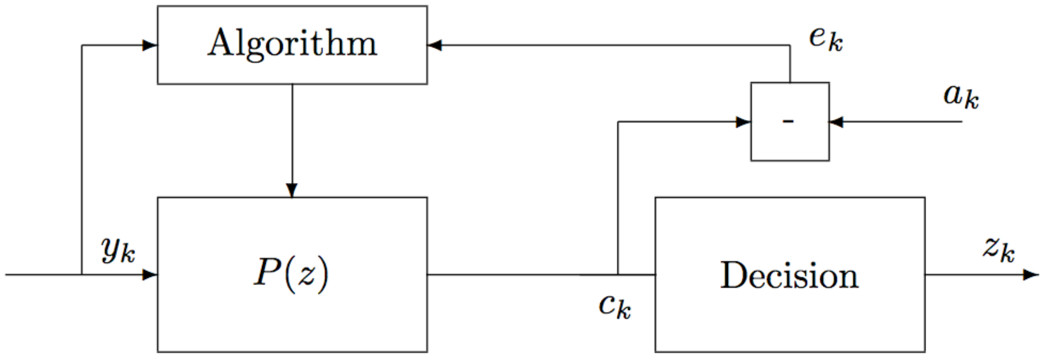
- For a quadratic function:

$$\begin{aligned} \frac{1}{2}\nabla^2_\Theta J(\Theta) &= \frac{1}{2}\nabla^2_\Theta J(\Theta_*) \\ &= E\frac{\partial}{\partial\Theta}\left\{ Y_k^* \left(Y_k^T \Theta - a_k \right) \right\} \\ &= E\left(Y_k^* Y_k^T \right) \end{aligned}$$

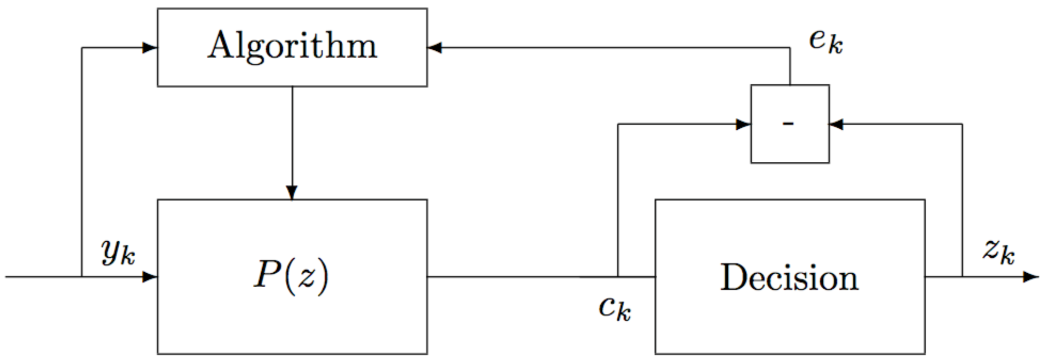
$$P_k = \frac{1}{k}\sum_{n=1}^{k} Y_n^* Y_n^T \leftrightarrow P_k = P_{k-1} + \gamma_k \left(Y_k^* Y_k^T - P_{k-1} \right)$$

Adaptive equalization:
Learning and tracking

- Learning



- Tracking



IIR LMS

- IIR Structure: $c_k(\Theta) = \Theta^T \varphi_k(\Theta)$
  with $\varphi_k^T(\Theta) = [y_k, y_{k-1}, \cdots, y_{k-p}; c_{k-1}, \cdots, c_{k-q}]$
  and $\Theta^T = \left[\Theta_0^{(1)}, \Theta_1^{(1)}, \cdots, \Theta_p^{(1)}; \Theta_1^{(2)}, \cdots, \Theta_q^{(2)}\right]$

$$c_k = \sum_{n=0}^{p} \Theta_n^{(1)} y_{k-n} + \sum_{n=1}^{q} \Theta_n^{(2)} c_{k-n}$$

- Objective function:

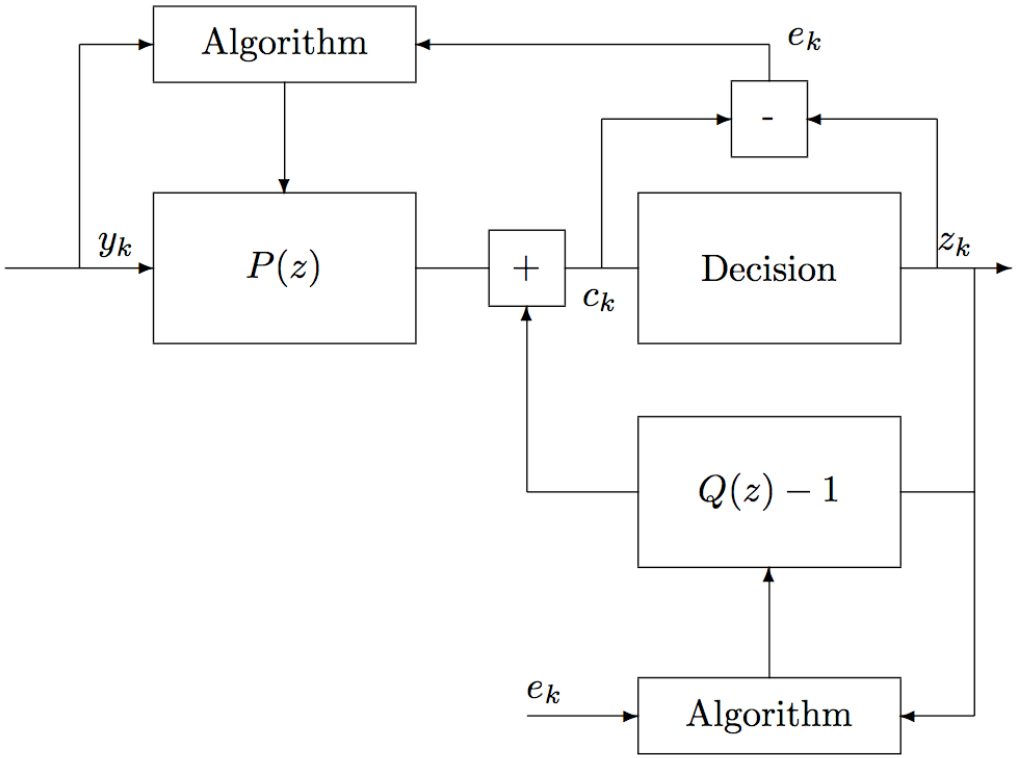
$$J(\Theta) = E|e_k|^2 \text{ avec } e_k = a_k - \varphi_k^T \Theta$$

- Adaptive algorithm:

$$\begin{aligned} e_k &= a_k - \Theta_{k-1}^T \varphi_k \\ \Theta_k &= \Theta_{k-1} + \mu_k e_k \varphi_k^* \\ \varphi_k^T(\Theta) &= [y_k, y_{k-1}, \cdots, y_{k-p}; c_{k-1}, \cdots, c_{k-q}]^T \end{aligned}$$

IIR LMS with decision

- The same idea can be applied to DFE



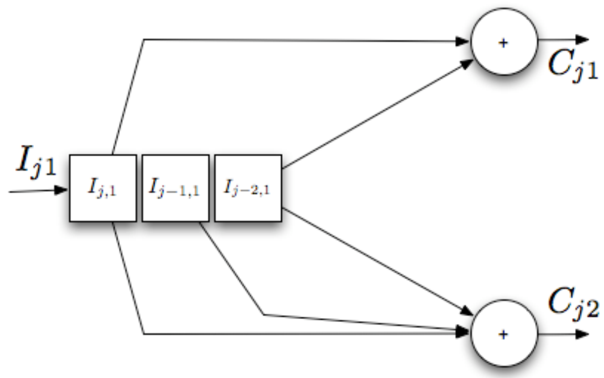
Maximum Likelihood Sequence Estimation

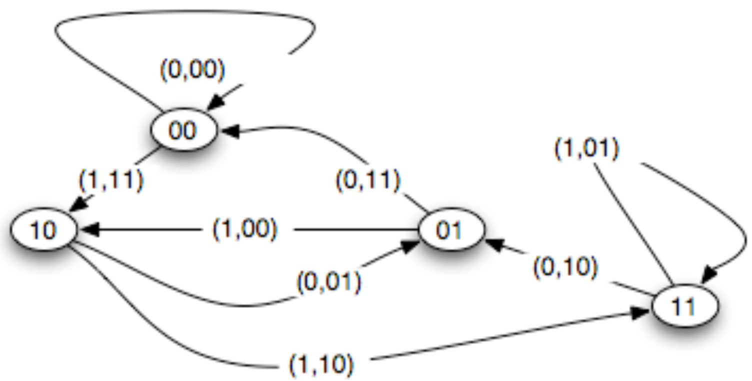
- Calculation of the transmitted sequence that maximizes the likelihood of observations.
- Efficient implementation:
  - Viterbi Algorithm

Fonctionnement de l’algorithme de Viterbi

Exemple d’un code convolutif

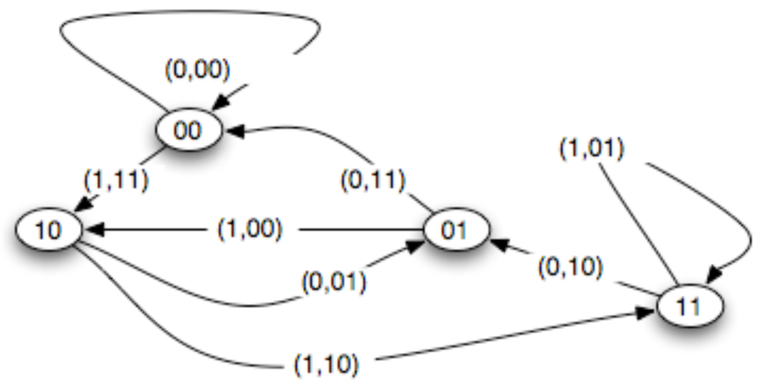
Codeur convolutif Machine à état fini

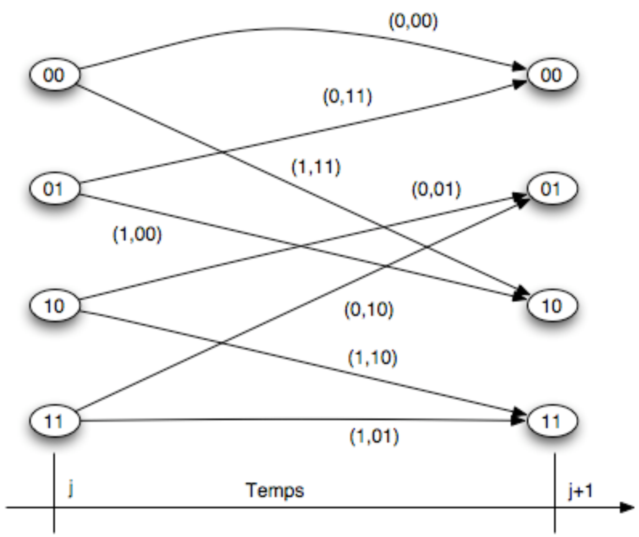




- Le codeur effectue une convolution sur les données transmises.
- Son état interne est déterminé par
  - L'entrée présente
  - Des (ici 2) variables internes
- Les transitions sont pilotées par l'entrée.

L'automate évolue au cours du temps





- Machine à état fini
  - l'état suivant ne dépend que de l'entrée et de l'état actuel.
- Tranche du treillis
  - chaque état peut transiter vers un autre état à l'instant suivant en fonction de l'entrée.

Maximum Likelihood Sequence Estimation (MLSE)

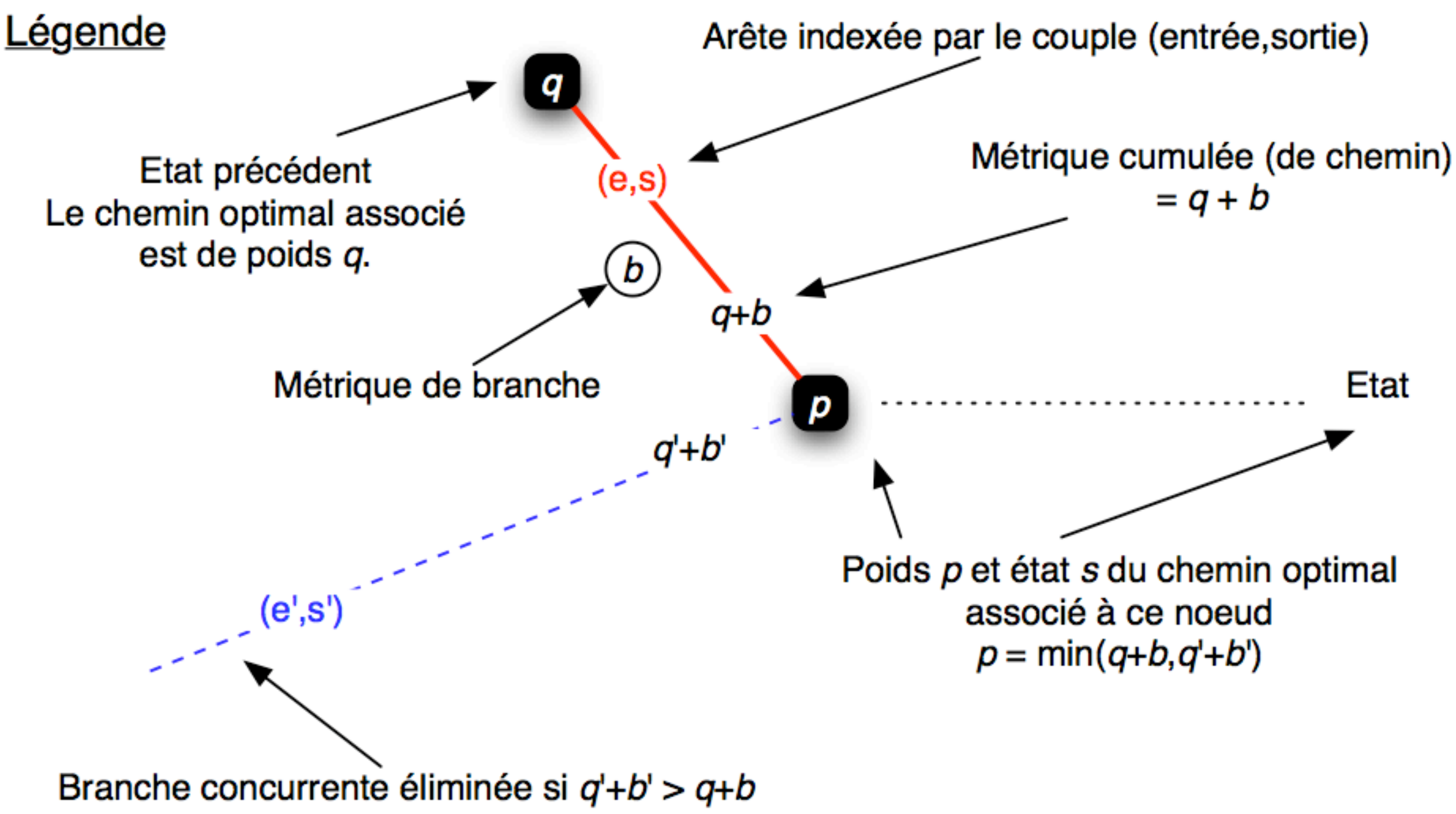
- L mots (de 1 bit) transmis engendre L+2 mots (de 2 bits) reçus.
- On ferme le treillis (2 zéros finaux pour ramener son état à 00)

- Le message traverse un canal à bruit additif blanc

- On observe *y,* version bruitée de la sortie du codeur
- Les transitions de la machine à état fini peuvent être indexées par :
  - la sortie du codeur C_j
  - les 2 états $s_j s_{j+1}$ entre lesquels s'effectue la transition
- On recherche la séquence qui maximise la vraisemblance des observations :

$$\log P(\boldsymbol{y}|\boldsymbol{C}) = \sum_{j=0}^{L+M-1} \log P\left(\boldsymbol{y}_j|\boldsymbol{C}_0 \cdots \boldsymbol{C}_j\right) = \sum_{j=0}^{L+M-1} \log P\left(\boldsymbol{y}_j|s_j s_{j+1}\right)$$

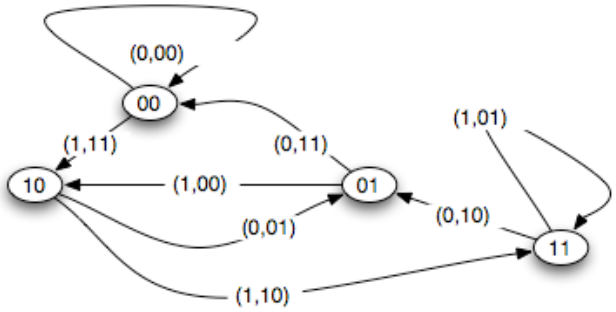
Notations des transitions sur le treillis



Canal binaire symétrique

- La métrique de branche est donnée par la distance de Hamming entre l'observation et la sortie du codeur.

- La métrique d'un chemin, métrique cumulée, est la somme des métriques des branches qui le compose.



Algorithme de Viterbi

Etat initial
00

Information
001

Séquence Transmise
00 00 11 01 11

Séquence Reçue
10 01 11 01 11

Algorithme de Viterbi

Illustration du fonctionnement

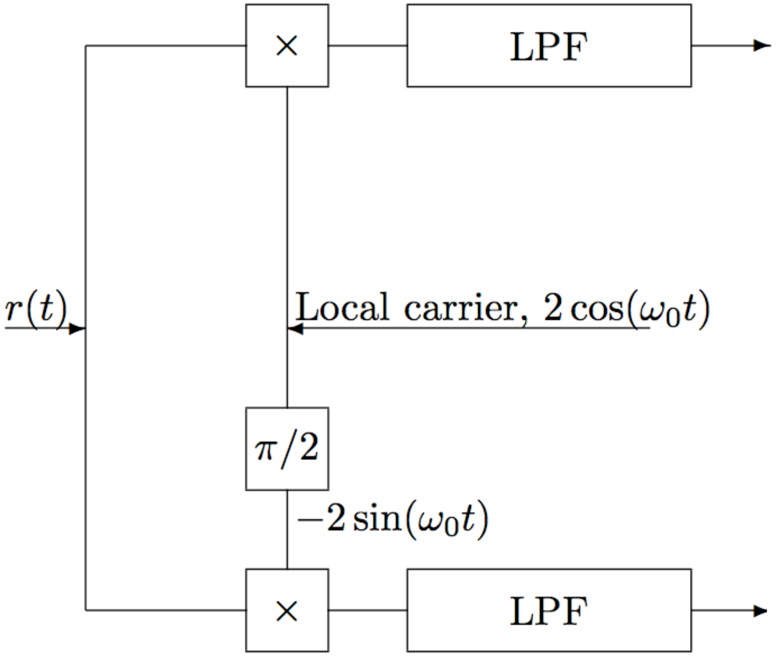
Channel estimation

- Viterbi for equalization required the knowledge of the channel impulse response:
  - Pilot sequence (midamble)
  - Blind estimation
  - Semi-blind techniques

Synchronization

- Carrier synchronization
- Timing recovery
- Strongly related to channel estimation

Origin and consequences of a phase misadjustment



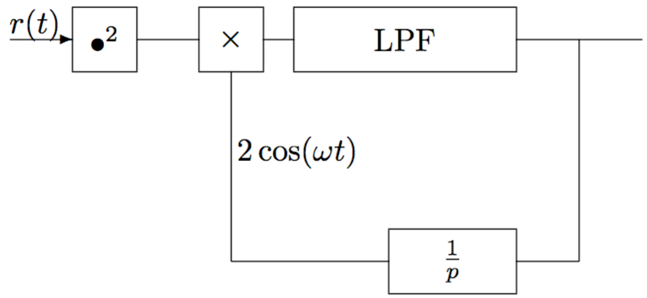
Problems:
- Doppler
- Electronics

$$r(t) = \mathrm{Re}\left[x(t)\, e^{i(\omega_0 t + \varphi)}\right]$$

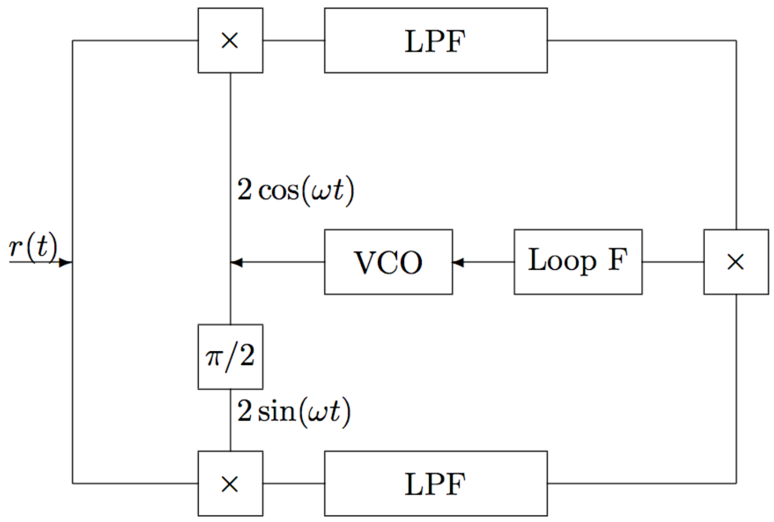
$$\begin{cases} P(t) &= \mathrm{Re}\left[x(t)\, e^{i\varphi}\right] &= \mathrm{Re}\left[x(t)\right]\cos(\varphi) &- &\mathrm{Im}\left[x(t)\right]\sin(\varphi) \\ Q(t) &= \mathrm{Im}\left[x(t)\, e^{i\varphi}\right] &= \mathrm{Re}\left[x(t)\right]\sin(\varphi) &+ &\mathrm{Im}\left[x(t)\right]\cos(\varphi) \end{cases}$$

Carrier synchronization
BPSK (Binary PSK)

- Squared loop:



- Costas loop:



Carrier synchronization

Problem and notations

- Basic model:

$$y_k = a_k e^{i\xi_k} + b_k$$

- Notations:

  - a_k belongs to a constellation
  - b_k is a complex, circular, gaussian white noise
  - ξ_k the time varying phase to be estimated

Digital Costas Loop (1)
BPSK

Error function:

$$\operatorname{Im}\left(y_k^2 e^{-i2\varphi_{k-1}}\right)$$

In a noise free context:

$$\begin{aligned}\operatorname{Im}\left(e^{i2(\xi-\varphi_{k-1})}\right) &= \sin\left(2\left(\xi-\varphi_{k-1}\right)\right)\\ &\simeq 2\left(\xi-\varphi_{k-1}\right)\ \text{pour}\ \xi\simeq\varphi_{k-1}\end{aligned}$$

Objective function:

$$\begin{aligned}J(\varphi) &= \mathrm{E}\left|y_k^2 e^{-i2\varphi}-1\right|^2\\ &= 4\sin^2\left(\xi-\varphi\right)+2\sigma^2\left(1+\sigma^2\right)\end{aligned}$$

Digital Costas Loop (2)
BPSK

The **cost function**:

$$J(\varphi) = \mathrm{E}\left|y_k^2 e^{-i2\varphi} - 1\right|^2$$

is minimum for $\varphi = \xi + k\pi$.

Stochastic gradient algorithm:

$$\varphi_k = \varphi_{k-1} + \gamma \mathrm{Im}\left(y_k^2 e^{-i2\varphi_{k-1}}\right)$$

A priori knowledge can be taken into account using a loop filter.
For example: second order digital Costas loop:

$$\varphi_k = \varphi_{k-1} + \left(\gamma^1 + \frac{\gamma^2}{1 - z^{-1}}\right) \aleph_k^{\mathrm{Im}}$$

Decision Feedback Loop (1)

DFL - BPSK

The likelihood writes:

$$P\left(Y_k|\xi,a_k\right)=\prod_{n=0}^{k}\frac{1}{\pi\sigma_b^2}\exp\left(-\frac{\left|y_n-a_ne^{i\xi}\right|^2}{\sigma_b^2}\right)$$

$$P\left(Y_k|\xi\right)=\left(\frac{1}{\pi\sigma_b^2}\right)^{k+1}\prod_{n=0}^{k}\left\{\begin{array}{c}\exp\left(-\frac{|y_n|^2-2\mathrm{Re}\left(y_ne^{-i\xi}\right)+1}{\sigma_b^2}\right)\\+\exp\left(-\frac{|y_n|^2+2\mathrm{Re}\left(y_ne^{-i\xi}\right)+1}{\sigma_b^2}\right)\end{array}\right\}$$

Optimum is achieved for:

$$\begin{aligned}\partial_\xi\log P\left(Y_k|\xi\right)&\propto\sum_{n=0}^{k}\partial_\xi\log\cosh\left(\frac{2}{\sigma_b^2}\mathrm{Re}\left(y_ne^{-i\xi}\right)\right)\\&=\sum_{n=0}^{k}\frac{2}{\sigma_b^2}\mathrm{Im}\left(y_ne^{-i\xi}\right)\tanh\left(\frac{2}{\sigma_b^2}\mathrm{Re}\left(y_ne^{-i\xi}\right)\right)\end{aligned}$$

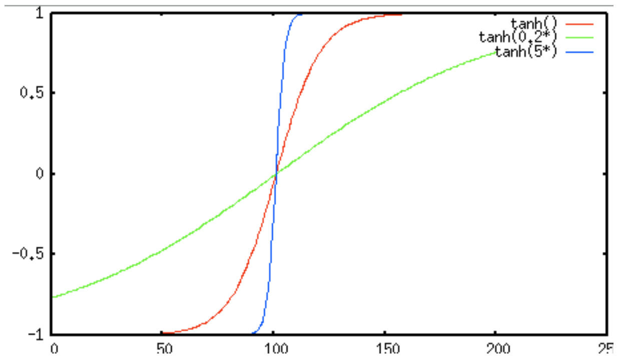
Adaptive approximation:

$$\varphi_{k+1}=\varphi_k+\frac{\gamma'}{\sigma_b^2}\mathrm{Im}\left(y_ke^{-i\varphi_k}\right)\tanh\left(\frac{2}{\sigma_b^2}\mathrm{Re}\left(y_ke^{-i\varphi_k}\right)\right)$$

Decision Feedback Loop (2)

DFL - BPSK

[High SNR] $\frac{2}{\sigma_b^2}\mathrm{Re}\left(y_k e^{-i\varphi_k}\right)$ is large and tanh is almost a sign function:



$\tanh\left(\frac{2}{\sigma_b^2}\mathrm{Re}\left(y_k e^{-i\varphi_k}\right)\right)$ is a decision $\widetilde{a_k}$ about symbol a_k.

The approximate ML solution is a *decision feedback loop* :

$$\varphi_{k+1} = \varphi_k + \gamma \mathrm{Im}\left(\widetilde{a_k} y_k e^{-i\varphi_k}\right)$$

[Low SNR] $\frac{2}{\sigma_b^2}\mathrm{Re}\left(y_k e^{-i\xi}\right)$ is small and tanh is almost linear:

$$\varphi_{k+1} = \varphi_k + \frac{\gamma'}{\sigma_b^2}\mathrm{Im}\left(y_k e^{-i\varphi_k}\right)\mathrm{Re}\left(y_k e^{-i\varphi_k}\right)$$

is a *Costas loop* :

$$\varphi_{k+1} = \varphi_k + \gamma \mathrm{Im}\left(y_k^2 e^{-i2\varphi_k}\right)$$

Decision Feedback Loop (3)

DFL - Generalization

- Cost function:

$$J(\varphi) = \mathrm{E}\left|a_k - y_k e^{-i\varphi}\right|^2$$

- Adaptive algorithm:

$$\varphi_{k+1} = \varphi_k + \gamma \mathrm{Im}\left(\widetilde{a_k^*} y_k e^{-i\varphi_k}\right)$$

Fourth power loop
QAM Phase estimation

$$\varphi_n^{\mathrm{FP}} = \frac{1}{4}\arg\left[E\left(a_k^{*4}\right)\frac{1}{n}\sum_{k=0}^{n-1} y_k^4\right] \longrightarrow \xi \bmod \frac{\pi}{2}$$

The expression $\sum_{k=0}^{n-1}\left|y_k^4 e^{-i4\varphi_n} - 1\right|^2$ has global minima corresponding to

$$\sum_{k=0}^{n-1}\Im\left(y_k^4 e^{-i4\varphi_n}\right) = 0 \rightarrow \tan\left(4\varphi_n\right) = \frac{\sum_{k=0}^{n-1}\Im\left(y_k^4\right)}{\sum_{k=0}^{n-1}\Re\left(y_k^4\right)} = \tan\left[4\left(\varphi_n^{\mathrm{FP}} - \frac{\pi}{4}\right)\right]$$

φ_n is the fourth power estimator up to a $\pi/4$ phase shift:

$$\varphi_n^{\mathrm{FP}} - \frac{\pi}{4} = \varphi_n \bmod \frac{\pi}{2}$$

Adaptive implementation:

Stochastic gradient scheme to minimize the objective function $J\left(\varphi\right) = E\left|y_k^4 e^{-i4\varphi} - 1\right|^2$

$$\varphi_k = \varphi_{k-1} + \gamma_k \mathrm{Im}\left(y_k^4 e^{-i4\varphi_{k-1}}\right)$$

Timing synchronization
Basic algorithm

- Adaptive algorithm approach

Cost function:

$$J_{\mathrm{PseudoEQM}}(\tau) = \mathrm{E}\left|x(kT+\tau) - \widetilde{a}_k\right|^2$$

Stochastic gradient algorithm:

$$\tau_{k+1} = \tau_k - \gamma\, \nabla_\tau \left|x(kT+\tau_k) - \widetilde{a}_k\right|^2\Big|_{\tau=\tau_k}$$

$$\longrightarrow$$

$$\tau_{k+1} = \tau_k - \gamma \mathrm{Re}\left\{ \left[x(kT+\tau_k) - \widetilde{a}_k\right]^* \left.\frac{d}{dt}x(t)\right|_{t=kT+\tau_k} \right\}$$

Timing synchronization
Basic algorithm implementation

- Estimation of the derivative:
  - Shannon interpolation
  - Finite difference approximation
- Fully digital implementation requires:
  - Digital interpolation (multirate filters, …)
  - 2 samples/symbol

Related to channel estimation

- Phase and timing are special cases of channel estimation.
- Separation, why?
  - Several time scales
  - Simplicity
  - History

Advanced modulation schemes

- OFDM
  - Principles
  - Discrete time formulation
  - DMT for deterministic channels:
    - Bit loading
    - Water filling
  - OFDM over random channels:
    - COFDM

OFDM: Principles (1)
Why and how

- Why? Equalizers face many problems:
  - Non minimum phase channels.
  - Spectral nulls.
  - Complexity (Viterbi) or suboptimality.
- How? Multicarrier approach:
  1. Several (for speed) low rate (to avoid ISI) transmissions.
  2. Non interfering sub transmissions.
  3. The Transmitter helps.
  4. Low complexity (FFT)

OFDM: Principles (2) Analog point of view (f)

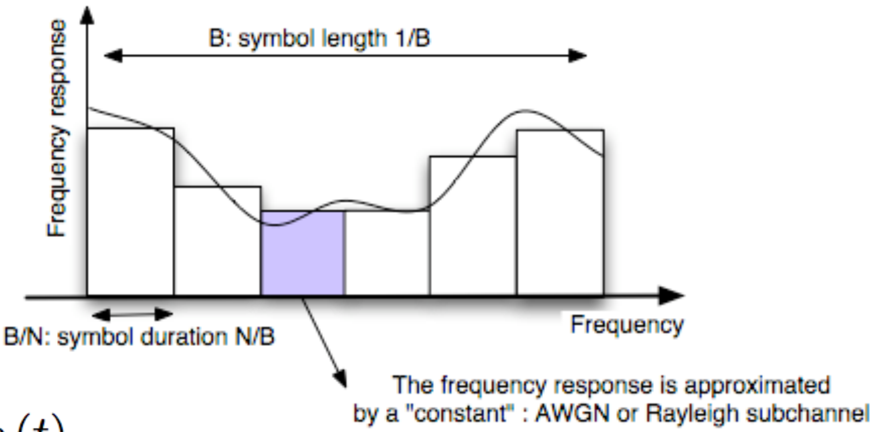
1. Several low rate transmissions.

   Analog point of view, frequency-domain vision.

   - Divide the bandwidth
   - Parallelize low band transmissions

$$x_k(t) = \sum_{n=-\infty}^{+\infty} a_n^k \, p(t - nT) \, e^{i2\pi\nu_k t} + b(t)$$

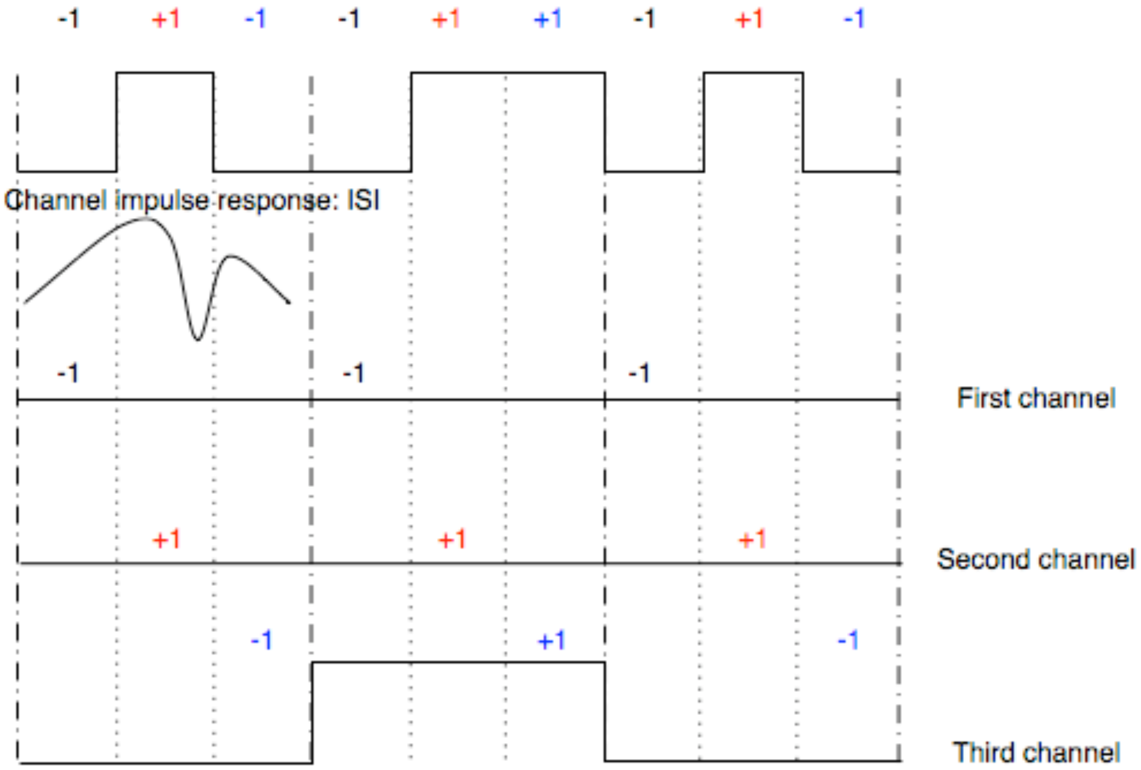
$$x(t) = \sum_{n=-\infty}^{+\infty} \sum_{k=0}^{N-1} a_n^k \, p(t - nT) \, e^{i2\pi\nu_k t} + b(t)$$



OFDM: Principles (3)
Analog point of view (t)

1. Several low rate transmissions.

   Analog point of view, time-domain vision.



- Short symbols: ISI
- Long symbols: ISI free transmission

OFDM: Principles (4)
Non interfering transmissions

2. Non interfering sub transmissions. Sub streams orthogonality

$$\langle x(t) \mid y(t) \rangle = \langle X(\nu) \mid Y(\nu) \rangle$$

- Distinct supports (too strong)
- Orthogonal carriers:
  - Infinite length: frequencies are always orthogonal.

$$\langle e^{i2\pi\nu_1 t} \mid e^{i2\pi\nu_2 t} \rangle = \delta(\nu_1 - \nu_2) \Rightarrow \text{ Orthogonality for } \nu_1 \neq \nu_2$$

  - Finite length: minimum carrier spacing: 1/T

$$\langle e^{i2\pi\nu_1 t} \mid e^{i2\pi\nu_2 t} \rangle_T = \frac{e^{i2\pi(\nu_1-\nu_2)T} - 1}{i2\pi(\nu_1 - \nu_2)} = 0 \Leftrightarrow \nu_1 - \nu_2 = \frac{k}{T}, k \in \mathbb{N}$$

OFDM:
Discrete time formulation

Analog formulation: $$\sum_{n=0}^{+\infty}\sum_{k=0}^{N-1} a_n^k p(t-nT)\mathrm{e}^{i2\pi\nu_k t}$$

For symbol *n*: $$\sum_{k=0}^{N-1} a_n^k \exp(i2\pi\nu_k t)$$

Discrete time formulation: $$\begin{cases} \nu_k = \dfrac{k}{T}, k = 0\cdots N-1 \Rightarrow \text{Shannon: } \nu_S = \dfrac{N}{T} \Rightarrow T_S = \dfrac{T}{N} \\ t_n = nT_S = n\dfrac{T}{N} \end{cases}$$

$$\exp(i2\pi\nu_k t_n) = \exp\left(i2\pi\left(\frac{k}{T}\right)\left(n\frac{T}{N}\right)\right) = \exp\left(\frac{i2\pi kn}{N}\right)$$

$$\left.\sum_{k=0}^{N-1} a_n^k \exp(i2\pi\nu_k t)\right|_{t_n = n\frac{T}{N}} = \sum_{k=0}^{N-1} a_n^k \exp\left(\frac{i2\pi kn}{N}\right)$$

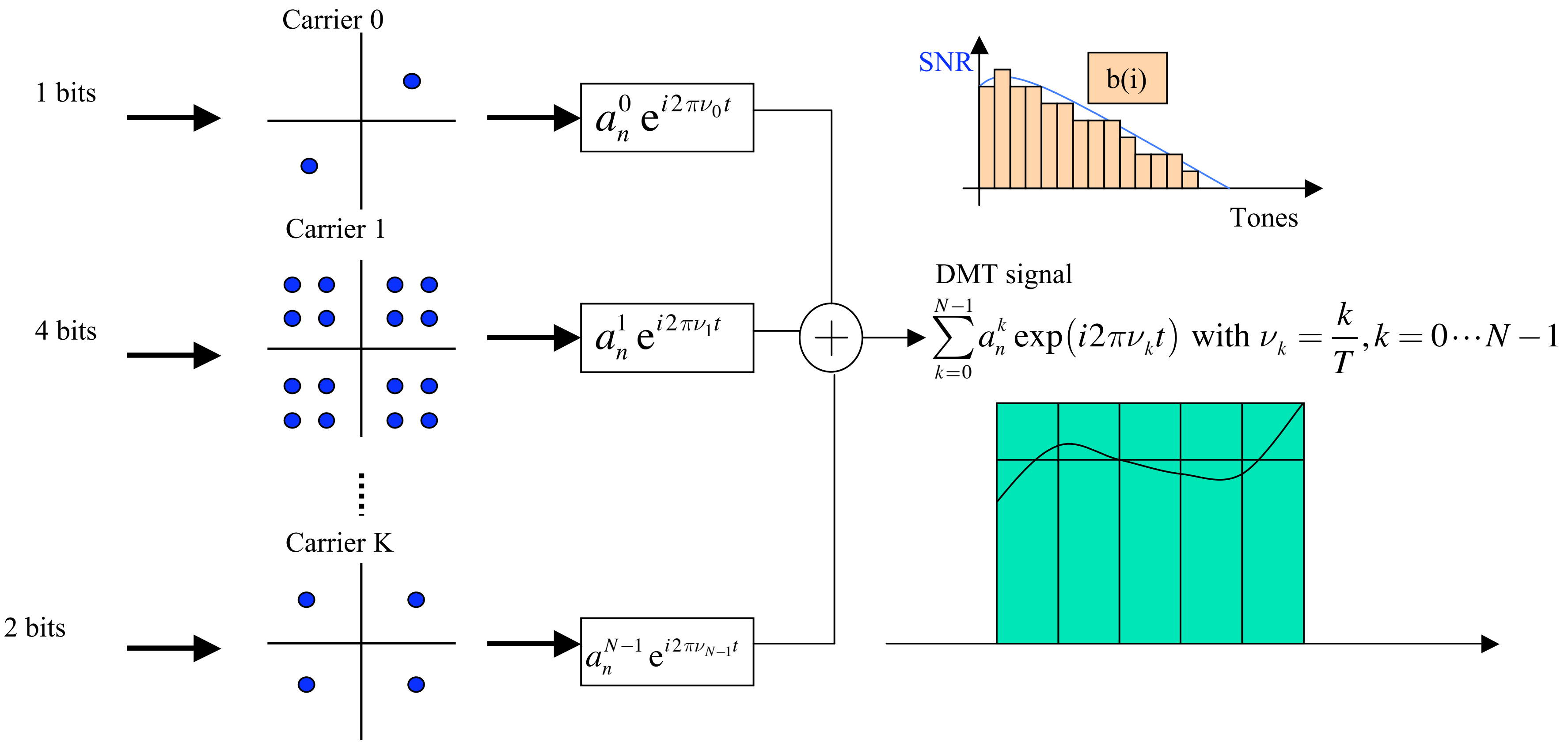
Efficient implementation:

$$TF^{-1}\left\{a_n^0, a_n^1, \cdots, a_n^{N-1}\right\} = TF\left\{\underline{a}_n\right\}$$

DMT:

DMT=OFDM + bit loading

Discrete Multi Tones



DMT:
Bit loading (1)

- Channel: $y = Hx + n$ with $\begin{cases} H \text{ complex gain of a flat channel} \\ x \in \text{QAM} \\ n \text{ gaussian noise with } E|n|^2 = \sigma^2 = N_0 \end{cases}$
- Energy of an even QAM:

$$E_Q = E\left(|x|^2\right) = \frac{M-1}{6} d_Q^2 \text{ where } \begin{cases} d_Q \text{ minimum distance of the QAM} \\ M \text{ number of states of the QAM} \end{cases}$$

- Symbol error rate:

$$P_E \text{ for the real component: } 2Q\left(\frac{d_{out}}{\sqrt{2N_0}}\right) \text{ with } Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{+\infty} e^{-t^2/2} dt$$

$$d_{out} \text{ minimum distance at the noise free channel output: } d_{out} = |H| d_Q$$

$$P_Q \leq 4Q\left(\sqrt{\frac{d_{out}^2}{2N_0}}\right) = 4Q\left(\sqrt{3\Gamma}\right) \text{ with } \Gamma = \frac{d_{out}^2}{6N_0} = \frac{|H|^2 d_Q^2}{6N_0} \qquad P_{bit} = K_b Q\left(\sqrt{\frac{d_{out}^2}{2N_0}}\right)$$

DMT:
Bit loading (2)

- Gap for a given P_e :

$$P_e / \dim < \frac{P_Q}{2} \Rightarrow \Gamma > \Gamma_0 \text{ with } \Gamma_0 = \frac{1}{3}\left[Q^{-1}\left(\frac{P_Q}{4}\right)\right]^2$$

- Bit loading:

$$E_Q = \frac{M-1}{6} d_Q^2 \Rightarrow M = 1 + \frac{6E_Q}{d_Q^2} = 1 + \left(\frac{E_Q |H|^2}{N_0}\right)\left(\frac{6N_0}{d_{out}^2}\right)$$

$$\boxed{\begin{aligned} b &= \log_2(M) = \log_2\left(1 + \frac{\rho_{\text{out}}}{\Gamma}\right) \\ \Gamma &= \Gamma_0 + \gamma_m - \gamma_c (dB) \end{aligned}}$$

$$\rho_{out} = \frac{E_Q |H|^2}{N_0}$$

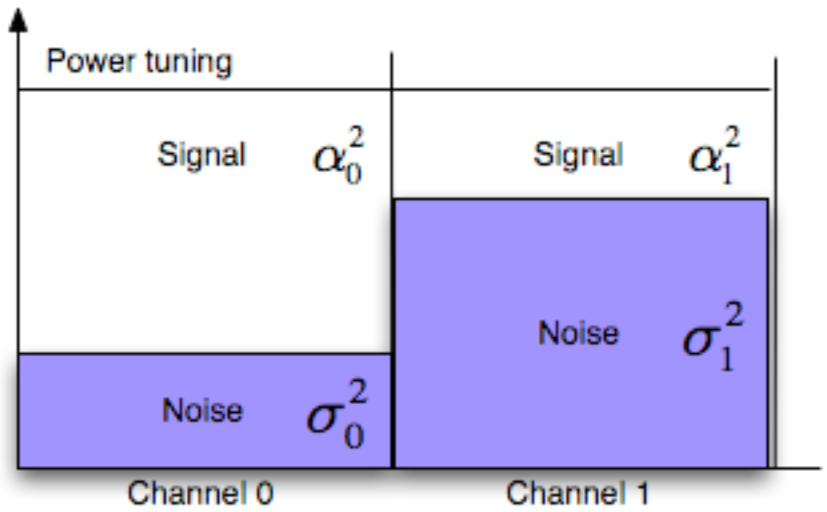
Margin

Coding gain

1 bit = 3 dB

DMT:
Water filling

- Two channels example:
  - Total power: P_T

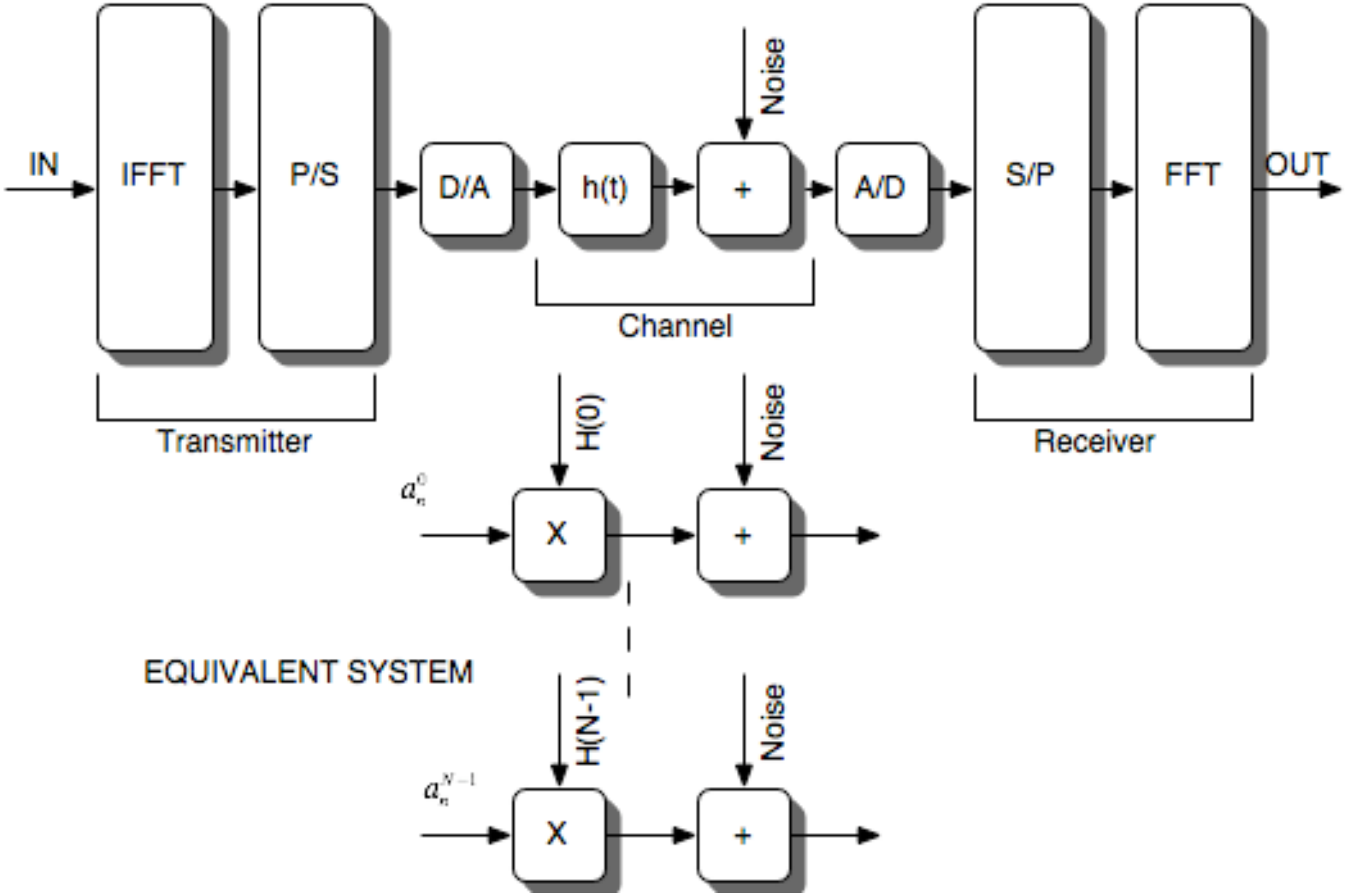


$$C_j = \log_2\left(1+\rho_j\right) = \log_2\left(1+\frac{\alpha_j^2}{\sigma_j^2}\right)$$

$$C = C_0 + C_1$$

$$\begin{cases} \alpha_0^2 = \dfrac{P_T}{2} + \left(\dfrac{\sigma_1^2 - \sigma_0^2}{2}\right) \\ \alpha_1^2 = \dfrac{P_T}{2} - \left(\dfrac{\sigma_1^2 - \sigma_0^2}{2}\right) \end{cases}$$

DMT: implementation



COFDM

- Random channels:
  - Repetitions:
    - > Coherence time
    - > Coherence bandwidth
- Interleaving
- Channel coding

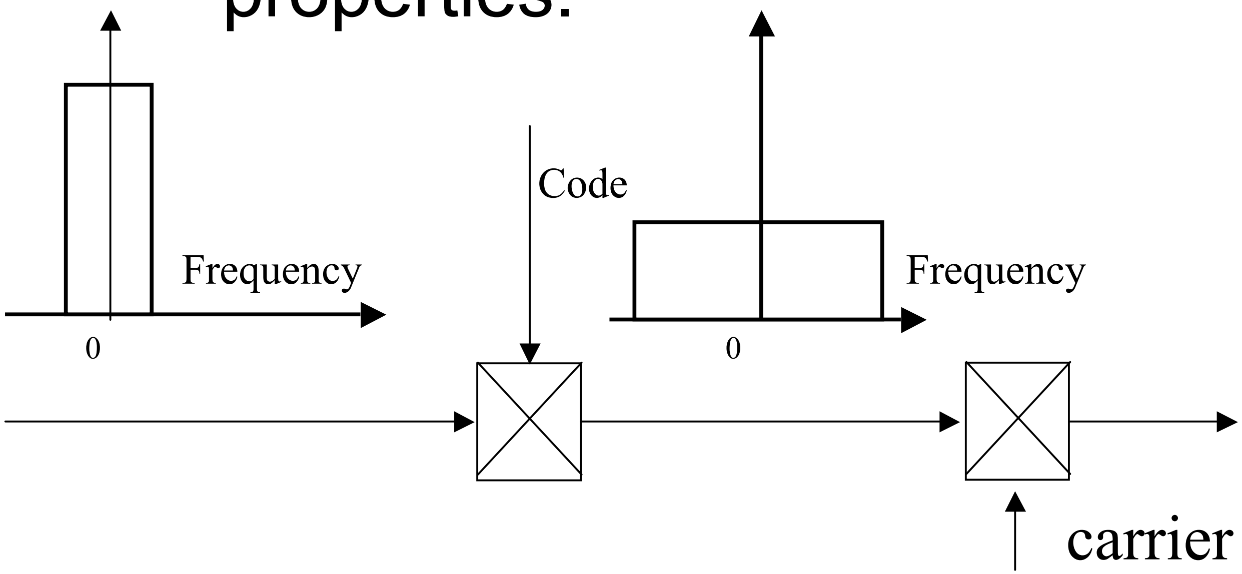
Advanced modulation schemes

- CDMA (Direct sequence)
  - Conventional receiver
  - Multi user
  - Multiuser detection

DS-SS: Principle

Direct Sequence Spread Spectrum

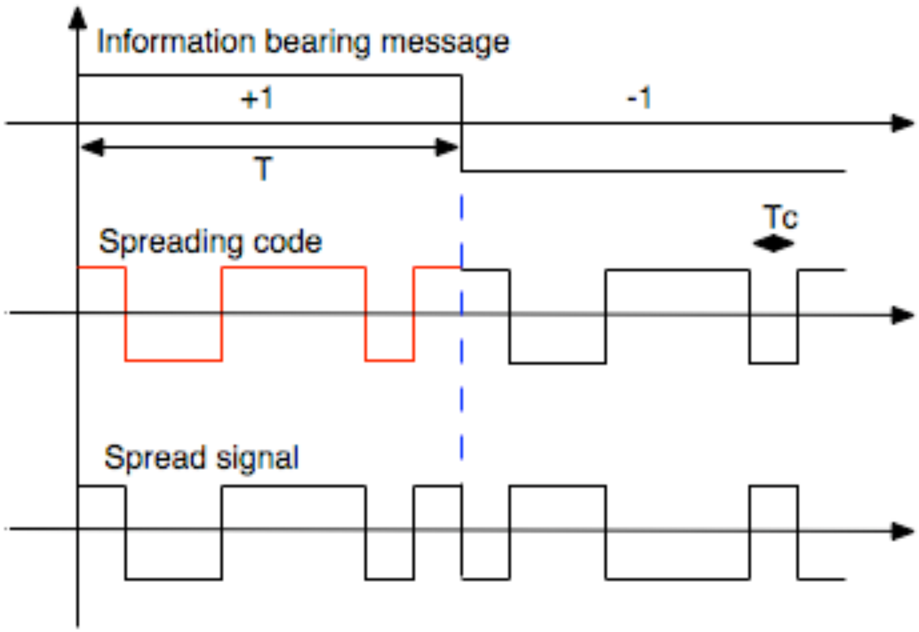
- A code is a pulse shaping with:
  - A wide band.
  - Good correlation and cross correlation properties.



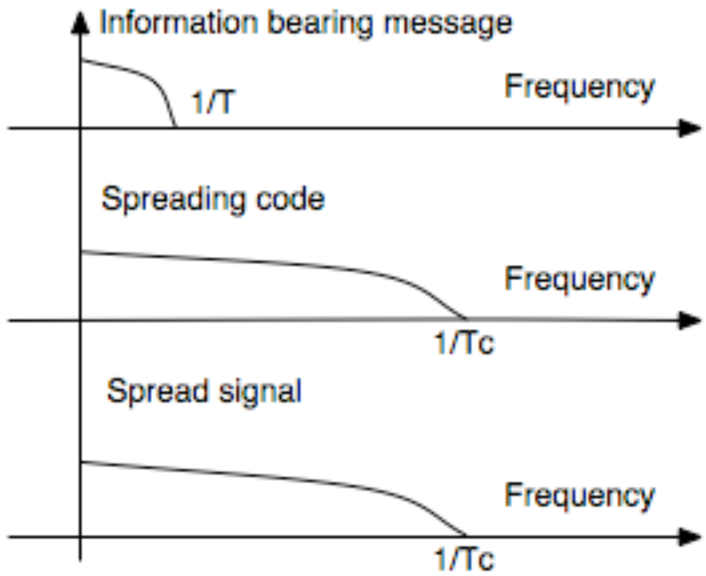
DS-SS

Modulation

Time Domain

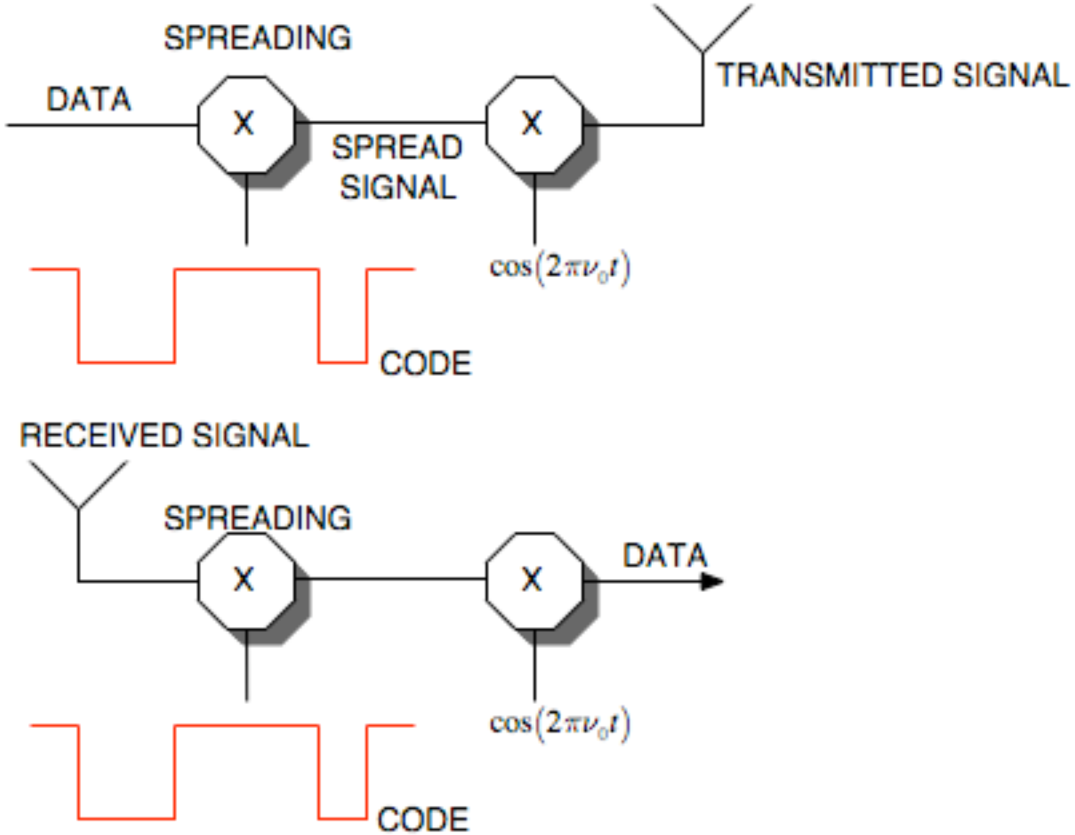


Frequency Domain



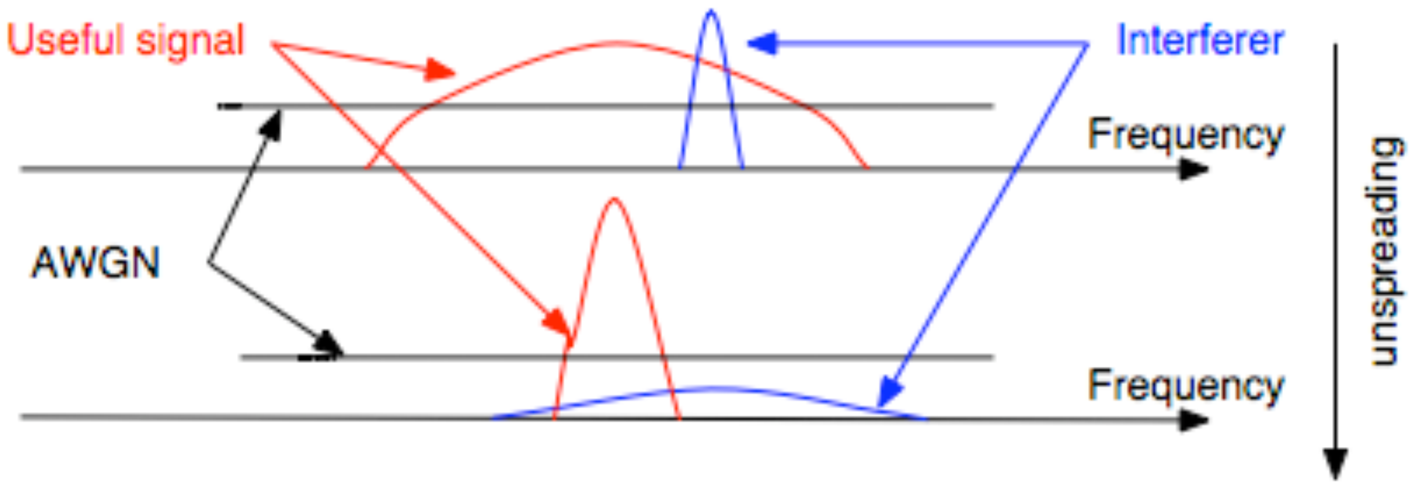
DS-SS

Transmitter and Receiver



DS-SS
Demodulation

Robustness, low band interferers



Multi User Detection

Symbols, user k

K users

Asynchronism

$k \in [0, T_S]$

$$y(t) = \sum_{k=1}^{K} \sum_{j=-M}^{M} A_k b_k[j] s_k(t - jT_S - \tau_k) + n(t)$$

Code, user k

Noise

Multi User Detection

K users

Symbols, user k

Code, user k

Noise

$$y(t)=\sum_{k=1}^{K}\sum_{j=-M}^{M}A_k b_k[j]s_k(t-jT_S-\tau_k)+n(t)$$

Asynchronism, $k\in[0,T_S]$

$$y_k=A_k b_k+\sum_{j\neq k}A_j b_j\rho_{jk}+n_k \text{ with } y_k=\int_0^{T_S}y(t)s_k(t)dt,\ k=1\cdots K$$

Synchron: $\left[\rho_{i,j}\right]=\left[\int_0^{T_S}s_i(t)s_j(t)dt\right]_{i,j}$

$$\mathbf{y}^T=(y_1,\cdots,y_K),\ \mathbf{b}^T=(b_1,\cdots,b_K),\ \mathbf{A}=diag(A_1,\cdots,A_k)$$

$$n_k=\int_0^{T_S}n(t)s_k(t)dt$$

$\mathbf{y}=\mathbf{RAb}+\mathbf{n}$, Noise covariance $\Gamma_\mathrm{n}=\sigma^2R$

Mono User performance

$$y(t) = Abs(t) + n(t), t \in [0, T_S[$$

1 user

$$P = Q\left(\frac{A}{\sigma}\right) \text{ for device: } \hat{b} = sign\left(\int_0^{T_S} y(t)s(t)dt\right)$$

2 users

$$P_1 = \frac{1}{2}Q\left(\frac{A_1 - A_2\rho}{\sigma}\right) + \frac{1}{2}Q\left(\frac{A_1 + A_2\rho}{\sigma}\right) \leq Q\left(\frac{A_1 - A_2|\rho|}{\sigma}\right)$$

K users

$$P_k \leq Q\left(\frac{A_k}{\sigma} - \sum_{j \neq k} \frac{A_j}{\sigma} |\rho_{jk}|\right)$$

Near-Far

Multi User Detection

- Optimum receiver

$$y(t) = \sum_{k=1}^{K} A_k b_k s_k(t) + n(t), t \in [0,T)$$

$$P(b_k | \{y(t), t \in [0,T)\})$$

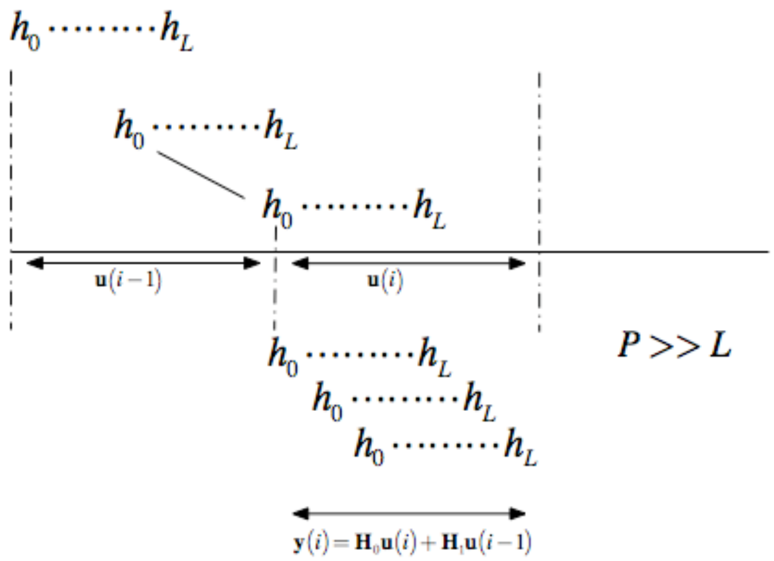
$$y_k = A_k b_k + \sum_{j \neq k} A_j b_j \rho_{jk} + n_k$$

- Maximization:

$$\exp\left(-\frac{1}{2\sigma^2}\int_0^T \left[y(t) - \sum_{k=1}^{K} b_k A_k s_k(t)\right]^2 dt\right)$$

$$\Omega(\underline{b}) = 2\underline{b}^T \underline{\underline{A}}\,\underline{y} - \underline{b}^T \underline{\underline{H}}\,\underline{b} \qquad \underline{\underline{H}} = \underline{\underline{A}}\,\underline{\underline{R}}\,\underline{\underline{A}}$$

Bloc transmission (1)
General model



Input $\quad \mathbf{u}(i)^T = [u(iP),\; u(iP+1), \cdots,\; u(iP+P-1)]$

Output $\quad \mathbf{y}(i)^T = [y(iP),\; y(iP+1), \cdots,\; y(iP+P-1)]$

$$\mathbf{y}(i) = \mathbf{H}_0 u(i) + \mathbf{H}_1 u(i-1) + \text{ noise.}$$

with

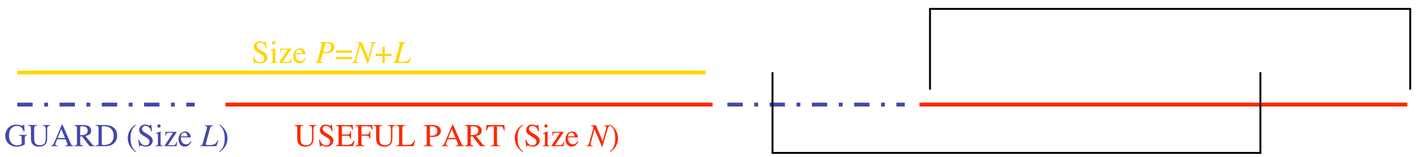
$$\mathbf{H}_0 = \begin{bmatrix} h(0) & 0 & 0 & \cdots & 0 \\ \vdots & h(0) & 0 & \cdots & 0 \\ h(L) & & \ddots & & \\ & \ddots & & \ddots & \\ & & h(L) & & h(0) \end{bmatrix} \text{ and } \mathbf{H}_1 = \begin{bmatrix} 0 & \cdots & h(L) & \cdots & h(1) \\ \vdots & \ddots & 0 & \ddots & \vdots \\ 0 & \cdots & \ddots & \cdots & h(L) \\ \vdots & \vdots & & \ddots & \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix}$$

Bloc transmission (2)
IBI Free transmission

- Guard interval to prevent IBI and …

$$\underline{u}(i) \rightarrow \underline{\underline{T}}\,\underline{u}(i)$$

$$\underline{x}(i) = \underline{\underline{H}}_0\,\underline{\underline{T}}\,\underline{u}(i) + \underline{\underline{H}}_1\,\underline{\underline{T}}\,\underline{u}(i-1) + noise$$



Two usual choices: Cyclic Prefix: frequency domain equalizer *but* sensitivity to zeros near the FFT grid.

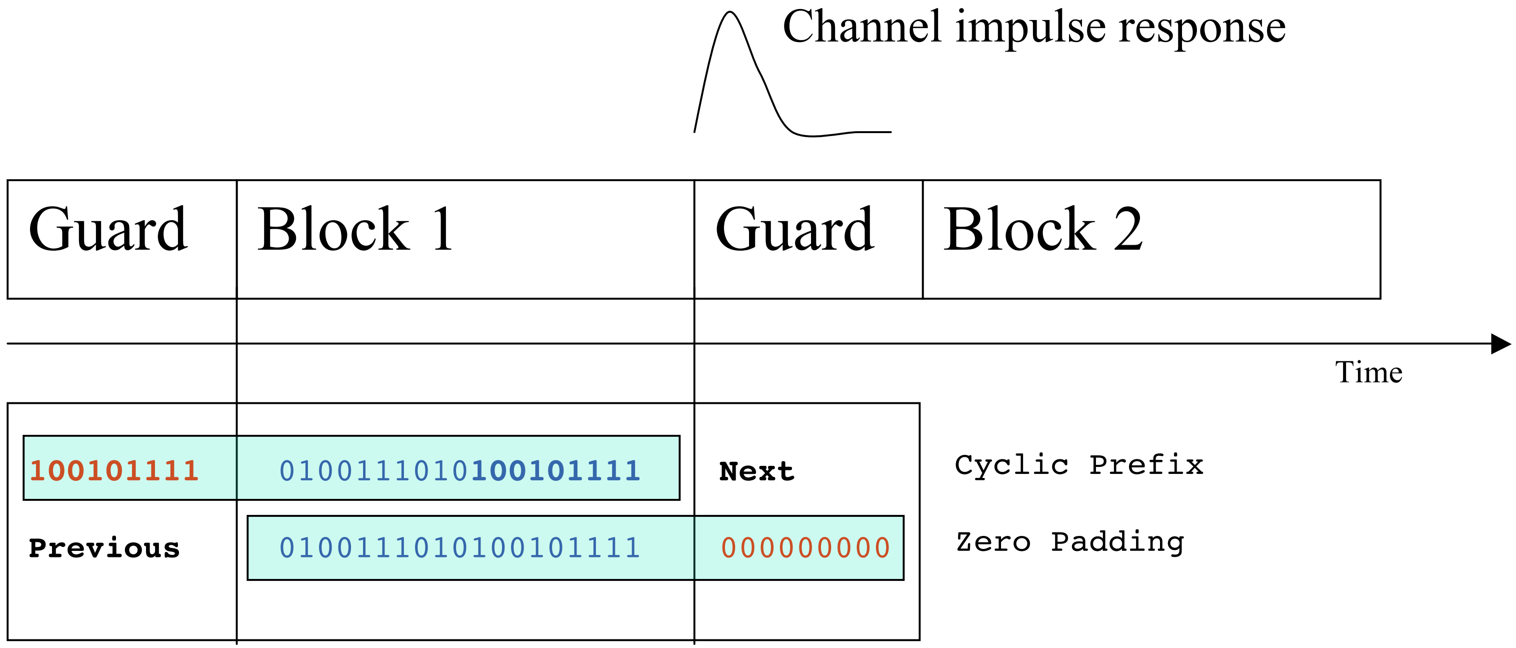
Zero Padding: symbol identifiability *but* high receiver complexity.

Bloc transmission (3)
Cyclic Prefix, Zero Padding

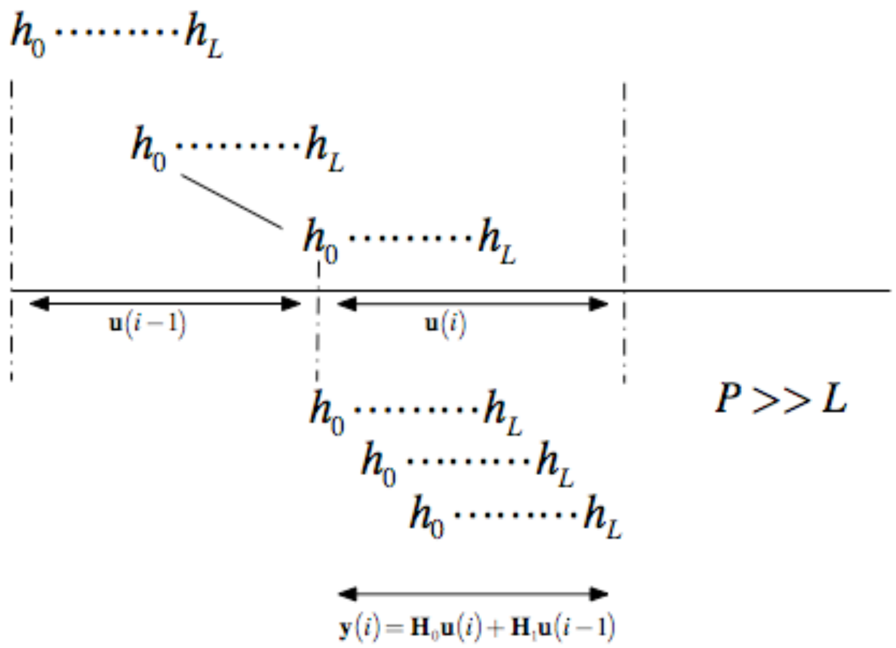
$$\mathbf{x}(i) \rightarrow \mathbf{u}(i) = \mathbf{T}\mathbf{x}(i)$$

thus:

$$\mathbf{y}(i) = \mathbf{H}_0\mathbf{T}\mathbf{x}(i) + \mathbf{H}_1\mathbf{T}\mathbf{x}(i-1) + \text{ noise}$$



Bloc transmission (4)
Cyclic Prefix



Add Cyclic Prefix (CP) to bloc $\mathbf{x}=[x_0 \cdots x_{N-1}]$:

$$\mathbf{x}=[x_0 \cdots x_{N-1}] \rightarrow \mathbf{u} = T_{CP}\mathbf{x} = [x_{N-L+1} \cdots x_{N-1} \mid x_0 \cdots x_{N-L+1} \cdots x_{N-1}]$$

$$\text{with } T_{CP} = \begin{bmatrix} I_{CP} \\ I_N \end{bmatrix}, \quad (N+L) \times N \text{ matrix}$$

$P = N + L$ and I_{CP} last lines of I_L

$$R_{CP} = [0_{N \times L} \, I_N]$$

$$\tilde{H} = R_{CP} H_0 T_{CP} \text{ with } \tilde{H}_{kl} = h(k-l)[N]$$

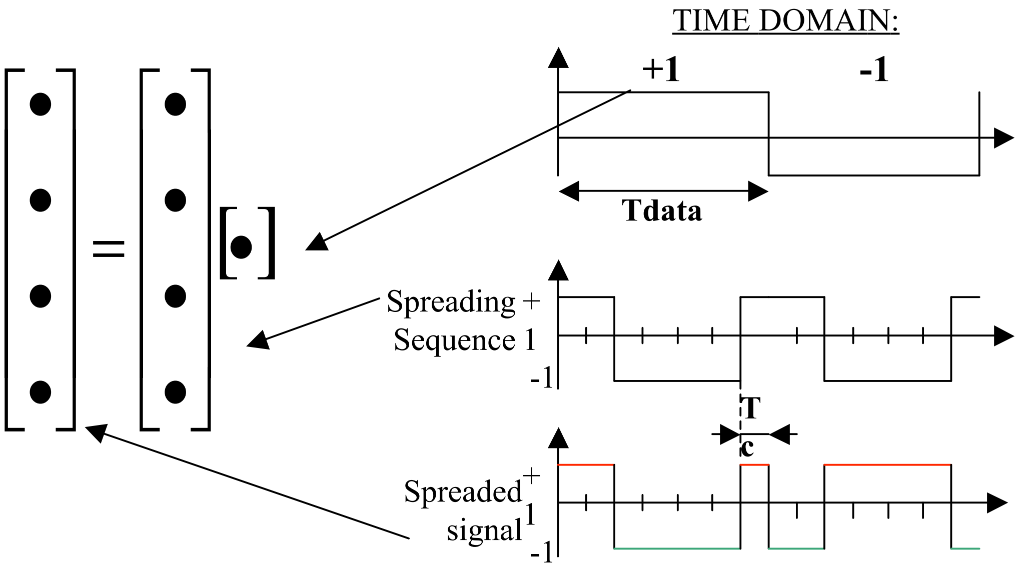
$$F\tilde{H}F^{-1} = D_H = \operatorname{diag}\left[H\left(e^{j0}\right) \cdots H\left(e^{j2\pi\frac{N-1}{N}}\right)\right]$$

$$\text{with } F_{kn} = \frac{1}{\sqrt{N}}\left(\exp\left(-i2\pi\frac{kn}{N}\right)\right)_{0 \le k,n \le N-1}$$

$$\text{and } H\left(\exp(i2\pi f)\right) = \sum_{n=0}^{L} h_n \exp(-i2\pi f n)$$

Usual DS-CDMA

Take one bit and spread it using a code (spreading sequence)

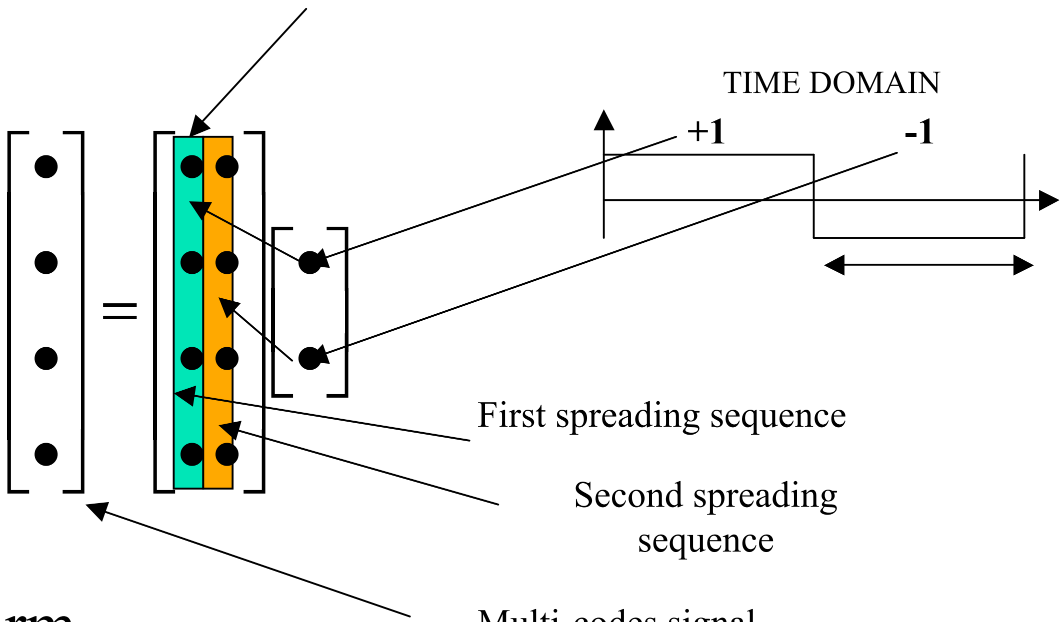


Matrix form

Multi-codes DS-CDMA

Take N bits and spread them using N codes (spreading sequences)

Each column is a code

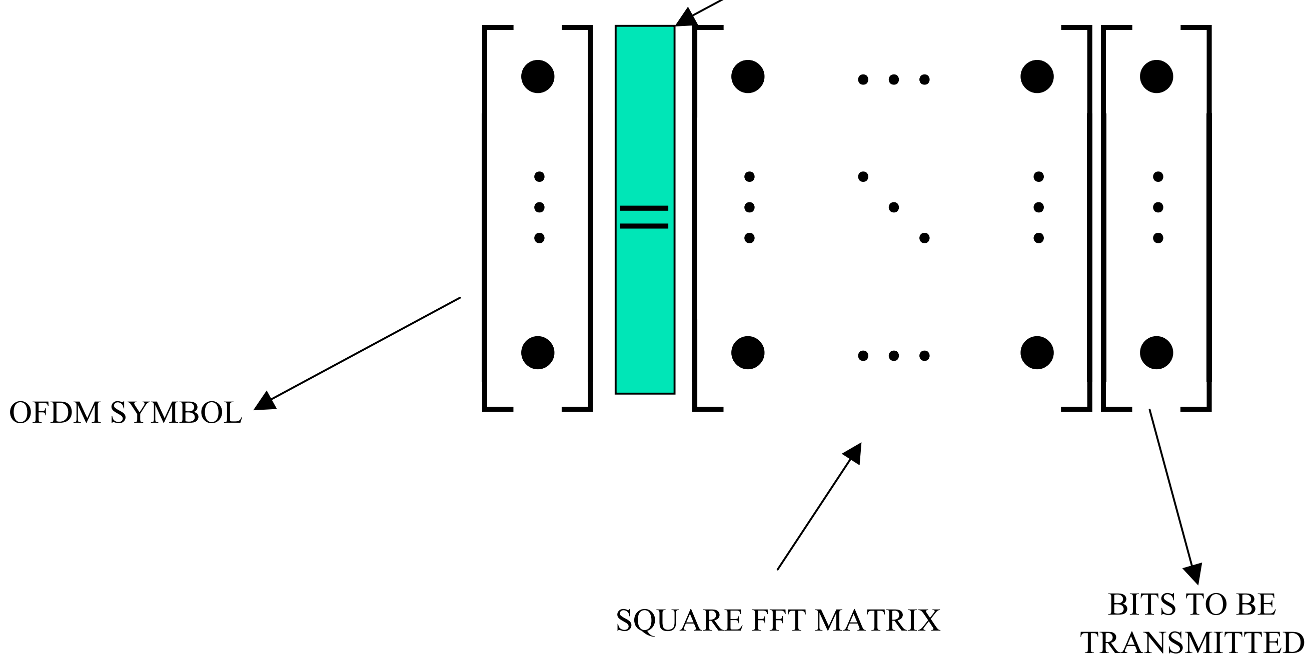


Matrix form

OFDM

A kind of multi-codes …

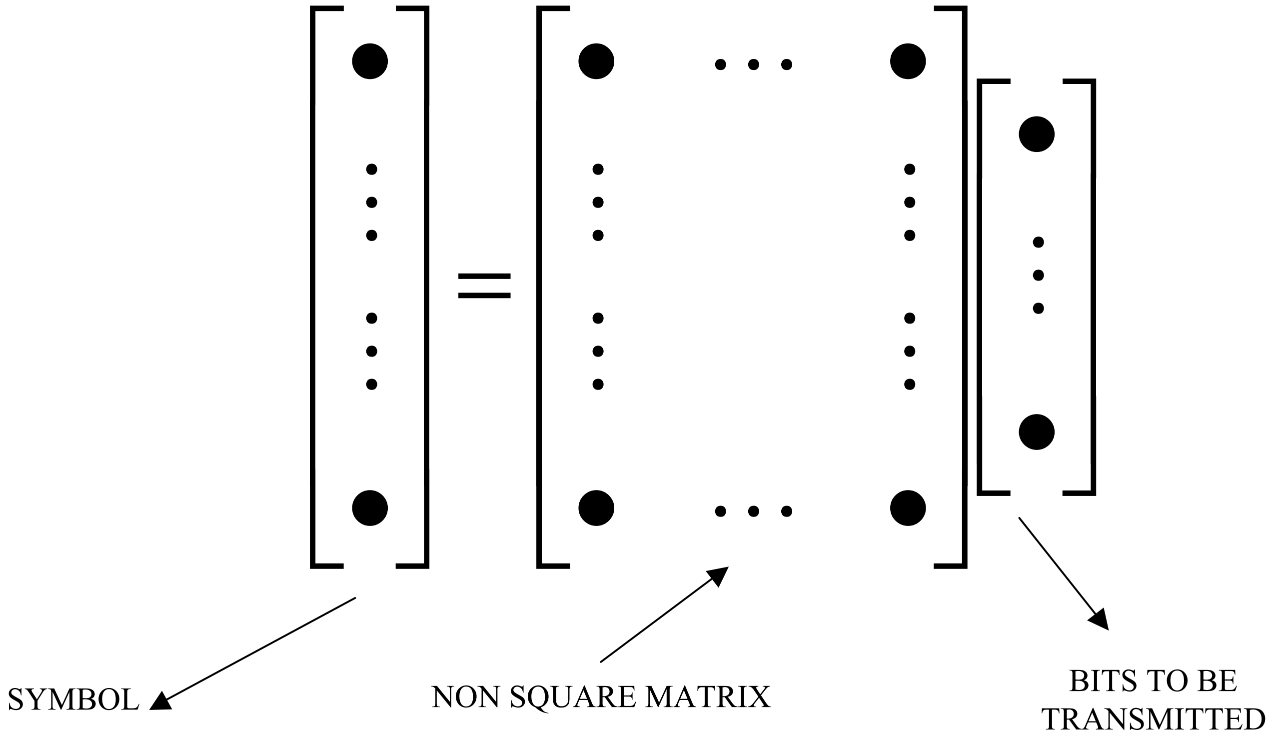
Each column is a sub-carrier, it is also a « code »



Matrix form

Block Spreading

A generalization of both OFDM and CDMA



Matrix form

Some Factorizations … MC-CDMA

$$\underline{\underline{\Sigma}} = \underline{\underline{F}}^H \underline{\underline{C}}$$

A GENERALIZED MC-CDMA

FFT matrix

$$\underline{\underline{C}} = \left[\underline{c}_1 \right]$$

LEADS TO THE USUAL MC-CDMA

Column of Σ are non-binary valued codes.

Some Factorizations …
Generalized MultiCarrier

$$\underline{\underline{\Sigma}} = \underline{\underline{F}}^H \, \underline{\underline{\Phi}}\,\underline{\underline{\Theta}}$$

GENERALIZED
MC

IFFT

MAPPING ON A SUBSET OF CARRIERS

FULL COLUMN RANK
SPREADING MATRIX

Factorize Σ in order to fight against the three major issues:

- Noise: error correcting code
- ISI: *e.g.* CP-DMT
- MUI: *e.g.* Vandermonde

What about the receiver ?

- Zero Forcing: pseudo-inverse

$$\underline{\underline{\Sigma}}^{\#}$$

Σ NEED TO BE FULL COLUMN RANK

- Matched Filter: Hermitian

$$\underline{\underline{\Sigma}}^{H}$$

$$\underline{\underline{\Sigma}} = \underline{\underline{F}}^{H}\,\underline{\underline{\Phi}}\,\underline{\underline{\Theta}}$$

OFDM – CDMA combinations

In a wireless context

	OFDM	CDMA
Strength	▪ MUI resilience ▪ Equalization	▪ Fading resilience
Weakness	▪ Selective Fading ▪ Zeros near the FFT grid.	▪ MUI sensitivity: MUD needed

COMPARISON serial vs block

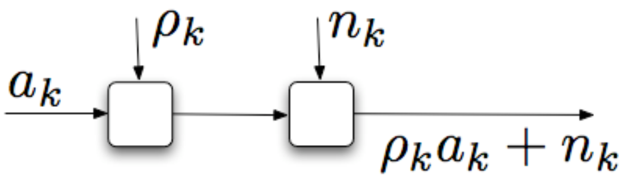
- Serial:
  - Minimum phase channel?
  - Zeros near the unit circle?
  - IIR response for a sample rate equalizer
- Block:
  - Symbol detectability (ZP)
  - Equalization (CP)

Trends

Multidimensional processing

- SISO - Single Input Single Output
  - Channel capacity (deterministic, random)
  - Random channels and diversity
- SIMO - Single Input Multiple Outputs
  - Beam forming
  - Diversity
- MIMO - Multiple Inputs Multiple Outputs

SISO channels (1)
Single Input – Single Output



- a_k belongs to a constellation (complex plan): BPSK, PSK, QAM.
- n_k white gaussian circular noise
- flat BUT TIME VARYING frequency response within the Nyquist band

- Flat frequency response: frequency non selective channels
- Several kinds of time selective channels:
  - $\rho_k = \rho$ is a time independent complex number.
  - ρ_k is random and changes from one channel use to another one.
  - ρ_k is random but remains constant over a large block:
    - the channel remains constant during a codeword duration.

SISO channels (2)
Limit performance

- Capacity $C_k = \log_2\left(1 + \frac{\rho_k^2}{\sigma^2}\right)$ bits
  - A well defined number for deterministic AWGN channels.
  - A random variable for non-ergodic random channel:
    - The capacity is limited by the lowest SNR values. If the lowest value is zero, e.g. Rayleigh channel, the capacity is zero too!
    - We must accept “channel outage”. What capacity can be achieved for almost all transmissions (typically 99%)?
    - This is the 1%-outage capacity.

SISO channels (3)
Bit Error Rate

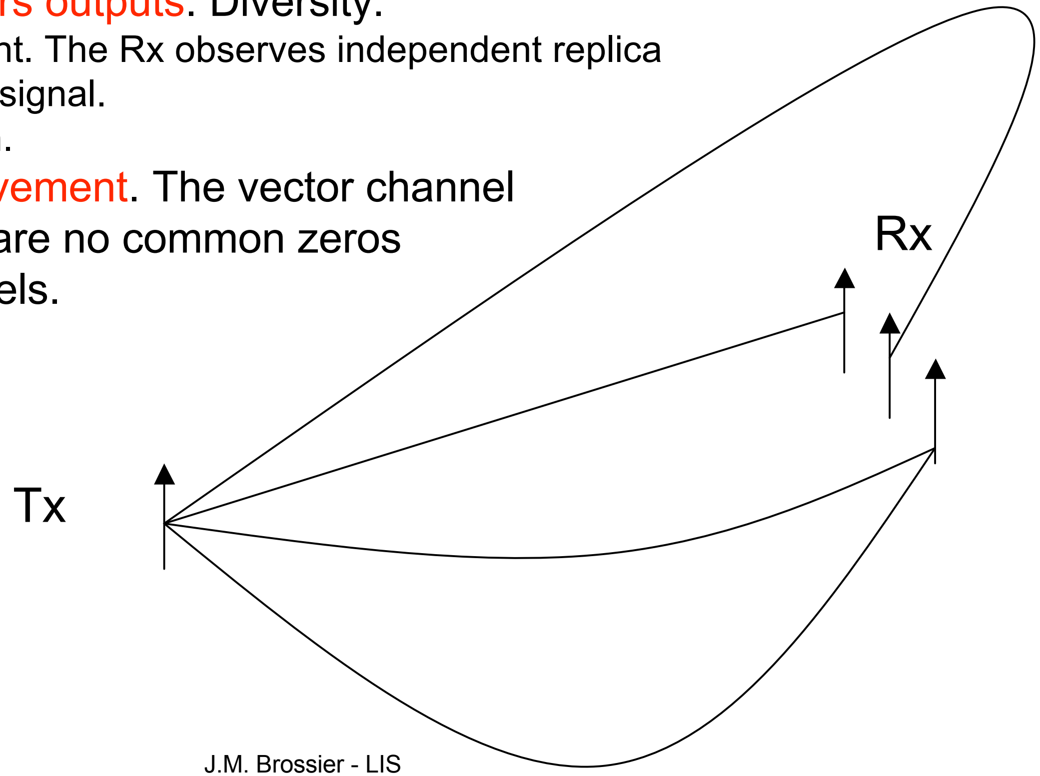
- For an AWGN channel, the error probability decreases exponentially as SNR increases.

- For a Rayleigh channel, it is only inversely proportional to the SNR: the error probability is dominated by worst cases.
  - Some “diversity” is needed.
    - Type I: no coding at the Tx.
      - Antenna array with Maximum Ratio Combining (MRC) exploits the available sensors to maximize the SNR.
      - Path diversity and rake receivers.
    - Type II: the transmitter repeats the symbols in some way (frequency, time, space, …), the receiver combines several sources of information.

SIMO channels (1)

Single Input – Multiple Outputs

- Strongly correlated sensors outputs. Beamforming, spatial filtering.
  - Fourier based methods.
  - High resolution angle estimation.
  - Adaptive antennas.
- Independent sensors outputs. Diversity.
  - SNR improvement. The Rx observes independent replica of the transmitted signal.
  - Fading mitigation.
- Identifiability improvement. The vector channel is invertible if there are no common zeros between sub-channels.

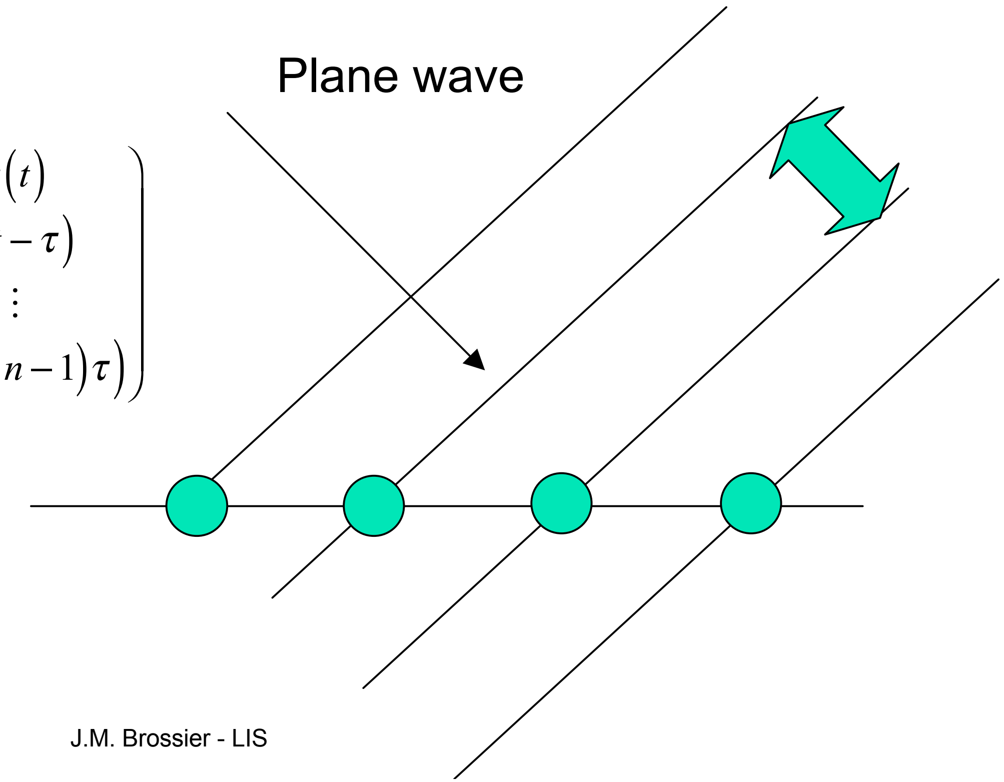


SIMO channels (2)
Beamforming

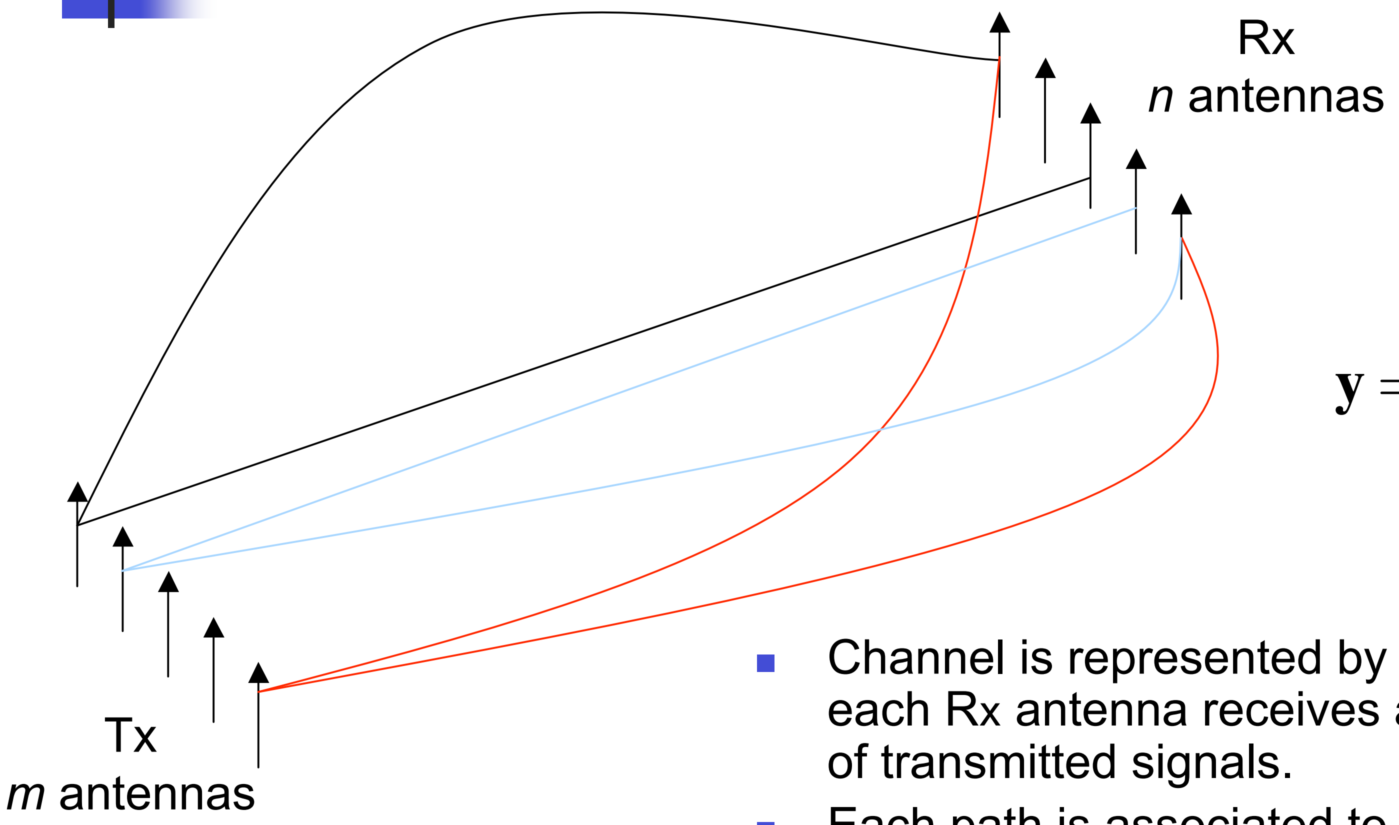
Narrowband approximation: a usual Fourier transform.

$$\mathbf{r}_v = \begin{pmatrix} 1 \\ e^{i\varphi} \\ \vdots \\ e^{i(n-1)\varphi} \end{pmatrix} s$$

$$\mathbf{r}(t) = a \begin{pmatrix} s(t) \\ s(t-\tau) \\ \vdots \\ s(t-(n-1)\tau) \end{pmatrix}$$



MIMO channels



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

- Channel is represented by an *n* x *m* matrix **A**: each Rx antenna receives a linear combination of transmitted signals.
- Each path is associated to a frequency non selective – flat - channel (perhaps subcarriers)

MIMO channels
Problems

- Spatial interference - non diagonal terms - are to be suppressed:
  - Multi User Detection like techniques
  - Equalization
  - Linear and non-linear techniques.

MIMO channels
Usual assumptions

- H_{ij} are independent random variables:
  - Coherence time much greater than symbol duration: H_{ij} are r.v. (no random process)
  - Antennas are spaced sufficiently apart: H_{ij} are independent.
- Fading conditions
  - Rayleigh fading valid for rich scattering environment – no line-of-sight (LOS). H_{ij} are independent Gaussian circular random variables.
  - Rice fading (LOS between Tx and Rx)

Parallel additive gaussian channels

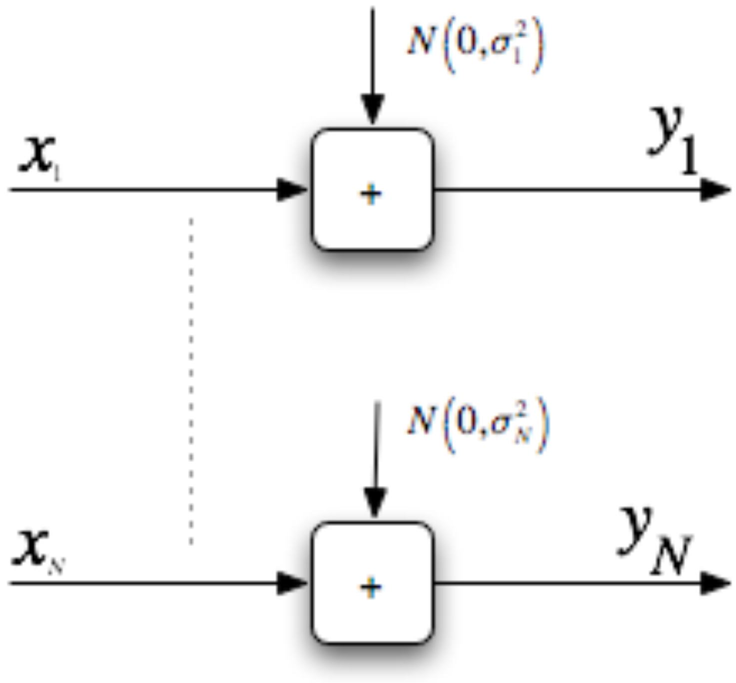
- Energy constraint:

$$E = \sum_{n=1}^{N} E_n$$

- Waterfilling solution:

$$C = \sum_{n=1}^{N} \log_2\left(1 + \frac{E_n}{\sigma_n^2}\right)$$

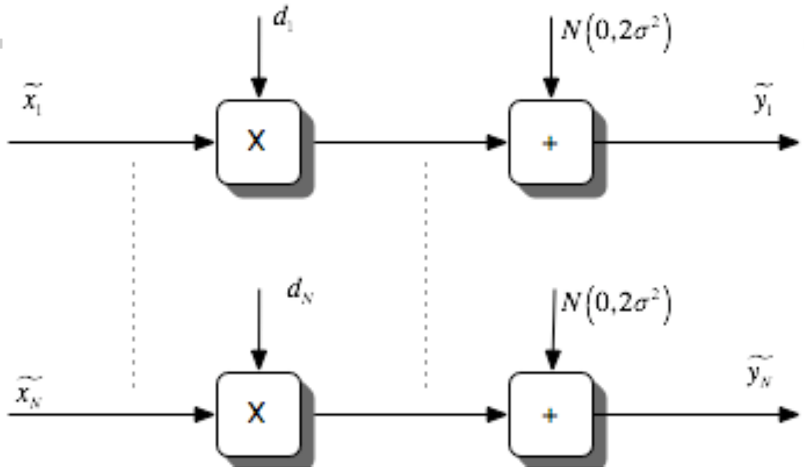
$$\begin{cases} \sigma_n^2 + E_n = \mu \text{ for } \sigma_n^2 < \mu \\ E_n = 0 \text{ for } \sigma_n^2 > \mu \end{cases}$$



Singular Values Decomposition of a MIMO channel

- Channel model:

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}$$
$$= \mathbf{U}\mathbf{D}\mathbf{V}^{+}\mathbf{x} + \mathbf{n}$$



- SVD:

$$\tilde{\mathbf{y}} = U^{+}\mathbf{y} = \mathbf{D}\tilde{\mathbf{x}} + U^{+}\mathbf{n}$$
$$\text{with } \tilde{\mathbf{x}} = \mathbf{V}^{+}\mathbf{x} \text{ and } \tilde{\mathbf{n}} = U^{+}\mathbf{n}$$

$$\tilde{y}_n = d_n \tilde{x}_n + \tilde{n}_n \Rightarrow C = \sum_{n=1}^{N} \log_2\left(1 + \frac{d_n^2 E_n}{2\sigma^2}\right)$$

Optimum transmission system

- Tx: Matrix processing using V
- Rx: Matrix processing using $\mathbf{U}^+$

Unknown CSI

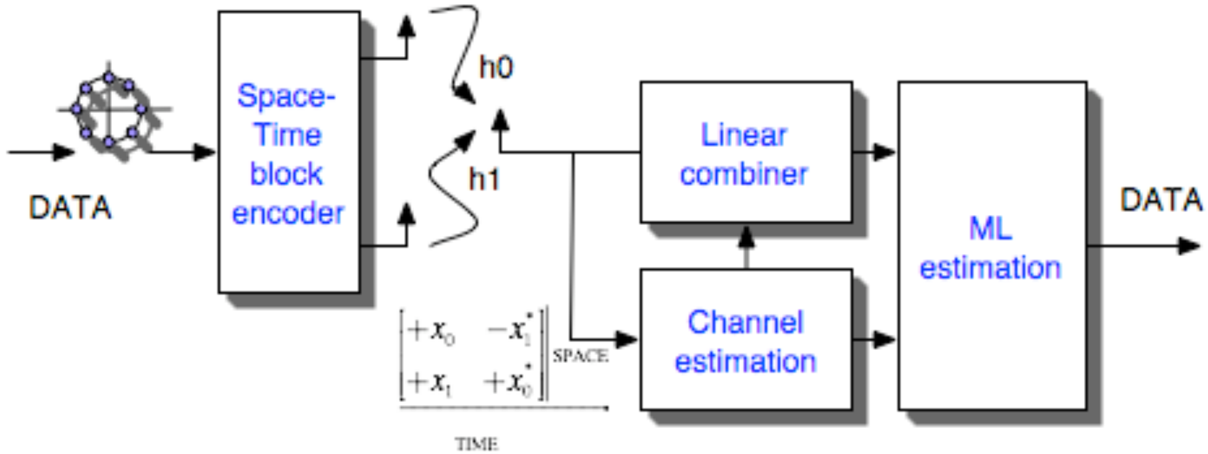
- Uniform distribution of the energy:

$$C = \sum_{n=1}^{N} \log_2 \left(1 + \frac{d_n^2 E}{2m\sigma^2} \right)$$

$$= \log_2 \det \left(I_n + \frac{E}{2m\sigma^2} \mathbf{H}^{+} \mathbf{H} \right)$$

Space-Time Processing

Alamouti scheme, Tx diversity



- Alamouti processing (1998):

$$[y_0, y_1] = [h_0 x_0 + h_1 x_1, -h_0 x_1^* + h_1 x_0^*] + [n_0, n_1]$$

$$\begin{bmatrix} h_0^* & -h_1 \\ h_1^* & h_0 \end{bmatrix} \begin{bmatrix} y_0 \\ -y_1^* \end{bmatrix} = \left[\left(|h_0|^2 + |h_1|^2\right) x_0 + \left(h_0^* n_0 + h_1 n_1^*\right), \left(|h_0|^2 + |h_1|^2\right) x_1 + \left(h_0^* n_1^* + h_1^* n_0\right)\right]$$

Space-Time Processing

Complex case: Alamouti scheme

- Full data rate, delay optimal code for N=2 only.
- Alamouti scheme

	Time 1	Time 2
Antenna 1	x_0	$-x_1^*$
Antenna 2	x_1	x_0^*

Capacity of a MIMO channel

Known CSI

$$\mathbf{r} = \rho \mathbf{A}\mathbf{x} + \mathbf{n}$$

- When m sensors are available at the transmitter and n at the receiver, the channel capacity writes:

$$\mathbf{C}_{mn} = \log\left(\det\left[\mathbf{I}_n + \frac{\rho_T}{m}\mathbf{A}\mathbf{A}^*\right]\right) \text{ with } \rho_T = \frac{\rho^2 P_T}{\sigma^2}$$

- The channel can be reduced to K (rank of **A**) parallel sub-channels:

$$C_{mn} = \sum_{i=1}^{K} \log\left[1 + \frac{\rho^2}{\sigma^2}\lambda_{X,i}\left|\lambda_{H,i}\right|^2\right]$$

$$\lambda_{H,i} \text{ singular values of } \mathbf{A} \text{ and } \sum_{i=1}^{m}\lambda_{X,i} = P_T$$

Capacity of a MIMO channel
Unknown CSI

- Known CSI (waterfilling)

$$C_{mn} = \sum_{i=1}^{K} \log\left[1 + \frac{\rho^2}{\sigma^2}\lambda_{X,i}\left|\lambda_{H,i}\right|^2\right]$$

$$\begin{cases} \lambda_{X,i} = \left(\psi - \dfrac{\sigma^2}{a^2 \lambda_{H,i}{}^2}\right)^+ \\ \psi \text{ is chosen to satisfy the constraint } \sum_{i=1}^{m} \lambda_{X,i} = P_T \end{cases}$$

- Unknown CSI

$$C_{mn} = \sum_{i=1}^{K} \log\left[1 + \frac{a^2}{\sigma^2}\lambda_{X,i}\lambda_{H,i}{}^2\right]$$

$$\lambda_{X,i} = \frac{P_T}{m}$$

Capacity of a MIMO channel

- For high SNR, capacity increases almost linearly with K.
- Examples:
  - 1 Rx, little to be gain for more than 4 Tx.
  - 2 Rx, little to be gain for more than 6 Tx.

Space only processing (1)

- Transmission of a ***x***-scaled version of symbol *s* through each antenna. Each symbol is transmitted only once.

- Received vector: $\mathbf{r} = \mathbf{A}\mathbf{x}s + \mathbf{n}$

- *s*: transmitted symbol.
- ***x***: scaling vector.
- ***A***: m x n channel matrix.
- ***n***: additive white Gaussian noise vector.

Space only processing (2)

- With a linear combiner **z**, decisions are based on:

$$\mathbf{D} = \mathbf{z}^* \left(\mathbf{A}\mathbf{x}s + \mathbf{n} \right)$$

- **Channel State Information (CSI) needed at the transmitter side.**
- **Optimum solution:**
  - **x:** principal right singular vector of A.
  - **z:** principal left singular vector of A.
  - Optimum SNR is given by: $SNR_{max} = \frac{\lambda_{max}(A^* A)}{\sigma^2} E_s$

Space-Time Processing (1)

- *N* channel use to transmit symbol s (time diversity)
- At time *k* (from 1 to N), symbol *s* is scaled using a scaling vector $\mathbf{x}_k$

$$r_k = \mathbf{A}\mathbf{x}_k s + \mathbf{n}_k$$

- Decision variable:

$$D = \frac{1}{N}\sum_{k=1}^{N} \mathbf{z}_k^* \left(\mathbf{A}\mathbf{x}_k s + \mathbf{n}_k \right)$$

- SNR at the linear combiner output:

$$SNR = \frac{N}{\sigma^2} \frac{\left| tr\left(\mathbf{A}\mathbf{R}_{xz} \right) \right|^2}{tr\left(\mathbf{R}_{zz} \right)} E_s$$

$$\mathbf{R}_{ab} = \frac{1}{N}\sum_{k=1}^{N} a_k b_k^*$$

Space-Time Processing (2)

- Optimum solution: $\mathbf{z}_k \propto \mathbf{A}\mathbf{x}_k$
- Optimum SNR: $SNR = \frac{N}{\sigma^2}\left|tr\left(\mathbf{A}\mathbf{R}_{\mathbf{xx}}\mathbf{A}^*\right)\right|E_s$
- The channel must be known at the receiver end.
- Two usual contraint:
  - Trace constraint: CSI available at the Tx, same solution than for space only processing.
  - Max Eigenvalue constraint: no CSI at the Tx.

Space-Time Processing (3)

- Tx is allowed to transmit in all possible directions.
- Maximum power in each direction lower than a given threshold: constraint on the maximum eigenvalue of $\mathbf{R}_{\mathbf{xx}} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{x}_k \mathbf{x}_k^*$
- Optimum solution:

$$\mathbf{R}_{\mathbf{xx}} = \mathbf{I}_m$$

$$SNR_{max} = N \frac{\mathrm{tr}(\mathbf{A}^* \mathbf{A})}{\sigma^2} E_s$$

Space-Time Processing (4)

- One possible choice to get $\mathbf{R}_{\mathbf{xx}} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{x}_k \mathbf{x}_k^* = \mathbf{I}$

  $\mathbf{x}_k = \sqrt{m}\mathbf{e}_k$ with $\mathbf{e}_k$ is the m x 1 vector having 1 in position k and 0 elsewhere.

- N must be at least m: date rate decreases by a factor N: cycling is not efficient.

- Solution: transmission of N symbols by channel use and being able to separate symbols at the Rx side.

Space-Time Processing (5)
Real case

- Coding matrix X_i : Column *k* of matrix X_i is the scaling vector for the transmission of symbol *i* at time *k*. Noting **N** the noise matrix, the received matrix is:

$$\mathbf{R} = \mathbf{A}\sum_{i=1}^{N} \mathbf{X}_i s_i + \mathbf{N}$$

$$D_i = \Re\left[\mathbf{A}^*\mathbf{A}\mathbf{X}_i\mathbf{X}_i^*\right]s_i + \sum_{j\neq i}\Re\left\{tr\left[\mathbf{A}^*\mathbf{A}\mathbf{X}_j\mathbf{X}_i^*\right]\right\}s_j$$

$$\mathbf{X}_i\mathbf{X}_i^* = \mathbf{I} \text{ and } \mathbf{X}_i\mathbf{X}_j^* = -\mathbf{X}_j\mathbf{X}_i^* \text{ for } i \neq j$$

- Full data rate and delay optimal solutions - using Hurwitz-Radon family - exist for *N*=2,4,8.

(complex numbers, quaternion, octonion)

- Full data rate solution exist for other values of *N* but they are no longer delay optimal.

Space-Time Processing (5)
Complex case: rate 3/4

- Complex coder can outperform the real coder. Transmission of *m*=3 complex symbols with 4 channel use.

- Higher rates with $N \geq m$?

Space-Time Coding (5)

Received matrix

Channel

Transmitted matrix

Noise matrix

Space

$$\mathbf{R} = \mathbf{A}\mathbf{X} + \mathbf{N}$$

Time

Optimum ML estimation of the transmitted signal:

$$\mathbf{X}_* = \arg\min_{\mathbf{X}} \left\| \mathbf{R} - \mathbf{A}\mathbf{X} \right\|^2$$

Space-Time Coding (5)
Code design criterion (Tarok)

Pair-wise error probability:

$$P(x \to x') \leq \underbrace{\left(\prod_{i=1}^{\rho} \lambda_i \right)^{-n}}_{\text{Coding gain}} \underbrace{\left(\frac{E_S}{4N_0} \right)^{-\rho n}}_{\text{Diversity gain}}$$

$\rho \leq m$: Rank of matrix **D** of differences $x - \hat{x}$

- **Tarok like design:**
  - **D** full rank for all differences. The achieved diversity is *n* times the min rank over the set of codewords pair.
  - Maximize the product of nonzero eigenvalues.
- Problems:
  - This bound is valid for very high SNR only.
  - the waterfall region becomes more and more extended as we increase the number of antennas.

Space-Time Coding (5)
Asymptotic analysis (Biglieri)

- m finite, n large: asymptotically, the PEP (pair-wise error probabilities) depends only on the Euclidian distance between code words and not on the determinant or the rank. The fading channel tends to a Gaussian channel with no spatial interference.
- 2 Tx antennas and 4 Rx antennas is often enough.

Space-Time Coding (5)
Asymptotic analysis

- Low complexity receivers:
  - Zero Forcing (pseudo-inverse) and MMSE
    - For m finite, n large: quasi ML performance.
    - For m,n large with a fixed ratio m/n:
      - ML receiver performance remains the same.
      - Bad performance of linear receivers when n is close to m: trade-off between ML complexity and the complexity due to a large number of antennas.
  - DFE like structures: D-BLAST
  - V-BLAST

Graphes factoriels
et
communications numériques

Modèles graphiques

- Historique :
  - Représentation graphique d'un code :
    - Gallager (1963), Tanner (1981)
    - Wiberg (1995) Ajout de variables cachées, interprétation Bayésienne.
  - Graphes factoriels (graphes de Tanner généralisés) Application aux fonctions
- Aji & McEliece 1997, 2000 : beaucoup d'algorithmes calculent des marginales d'un produit de fonctions
- Graphes factoriels
  - Applicable lorsque la solution exacte est d'une complexité rédhibitoire.
  - Liens avec d'autres approches : MRF (Markov Random Fields), réseaux Bayésiens (BP est un cas particulier de SP)

Marginalisation d'un produit de fonctions

- Fonction de *n* variables :

$$g(x_1,\cdots,x_n) : S = A_1 \times \cdots \times A_n \to R$$

$$n \text{ marginales } g_i(x_i) = \sum_{\sim x_i} g(x_1,\cdots,x_n)$$

- Algorithme de calcul de marginales efficace :
  - Exploiter les factorisations

$$g(x_1,\cdots,x_n) = \prod_{j\in J} f_j(X_j)$$

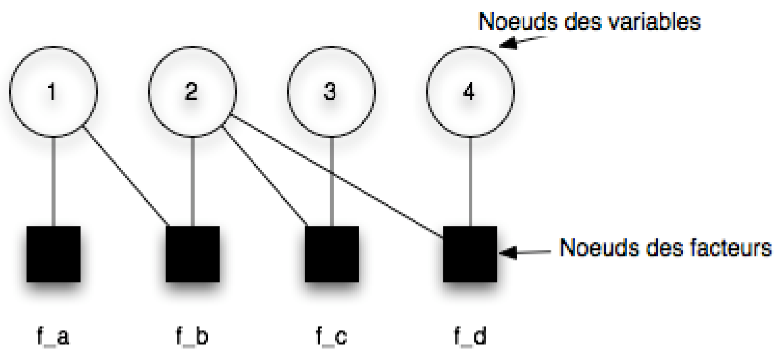
  - Réutiliser les sommes partielles

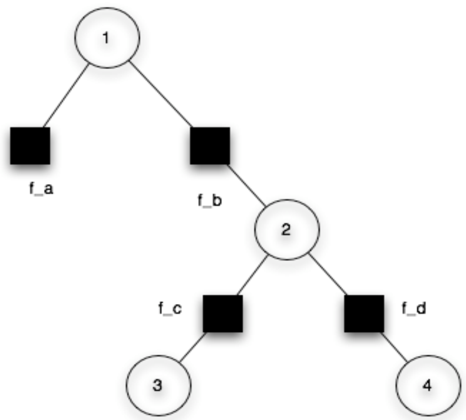
Modèle graphique
Graphe factoriel

- Graphe biparti

$$g(x_1,x_2,x_3,x_4)=f_a(x_1)\,f_b(x_1,x_2)\,f_c(x_2,x_3)\,f_d(x_2,x_4)$$

- Arbre enraciné en 1





Graphes et modélisation

- Approche comportementale
  - Configurations valides des variables
- Approche probabiliste
  - Représentation d’une probabilité conjointe
- Approche mixte
  - Exemple : codage canal et communications numériques.

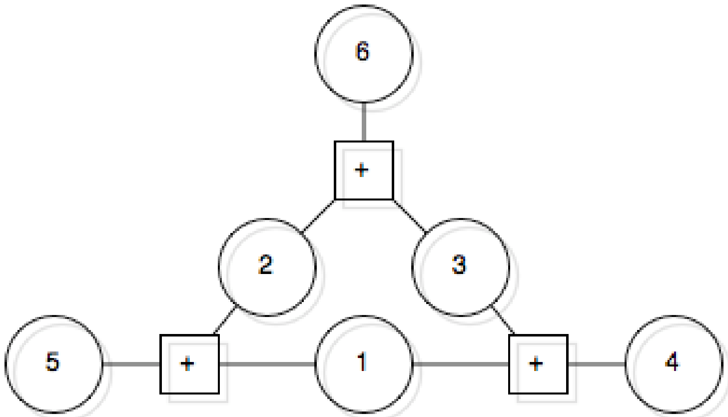
Approche comportementale

Exemple d’un code bloc linéaire

- Fonction indicatrice d’un ensemble B

$$1_B(x_1,\cdots,x_n)=\left[(x_1,\cdots,x_n)\in B\right]$$

- Calcul par une série de tests successifs



$$H=\begin{array}{c}\underbrace{\begin{bmatrix}110010\\011001\\101100\end{bmatrix}}_{n}\end{array} n-k$$

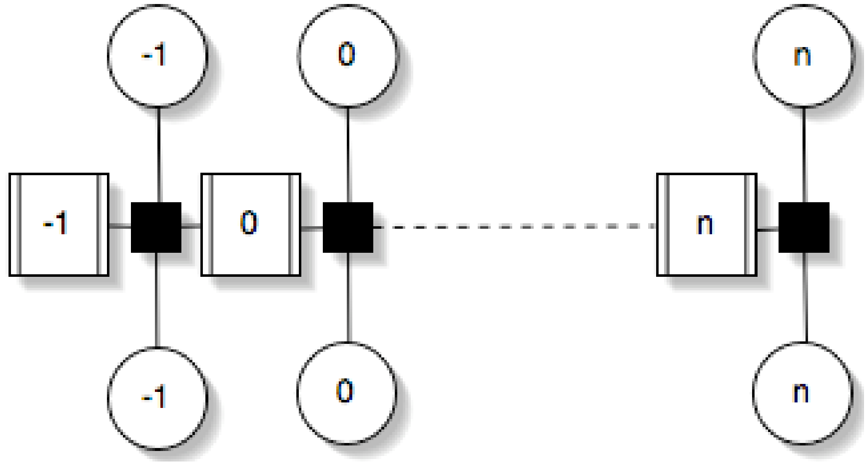
$$[P_1\wedge P_2\wedge\cdots\wedge P_n]=[P_1][P_2]\cdots[P_n]$$

$$1_{code}(x_1,\cdots,x_n)=[x_1\oplus x_2\oplus x_5=0][x_2\oplus x_3\oplus x_6=0][x_1\oplus x_3\oplus x_4=0]$$

Approche comportementale

Exemple des modèles de Markov cachés

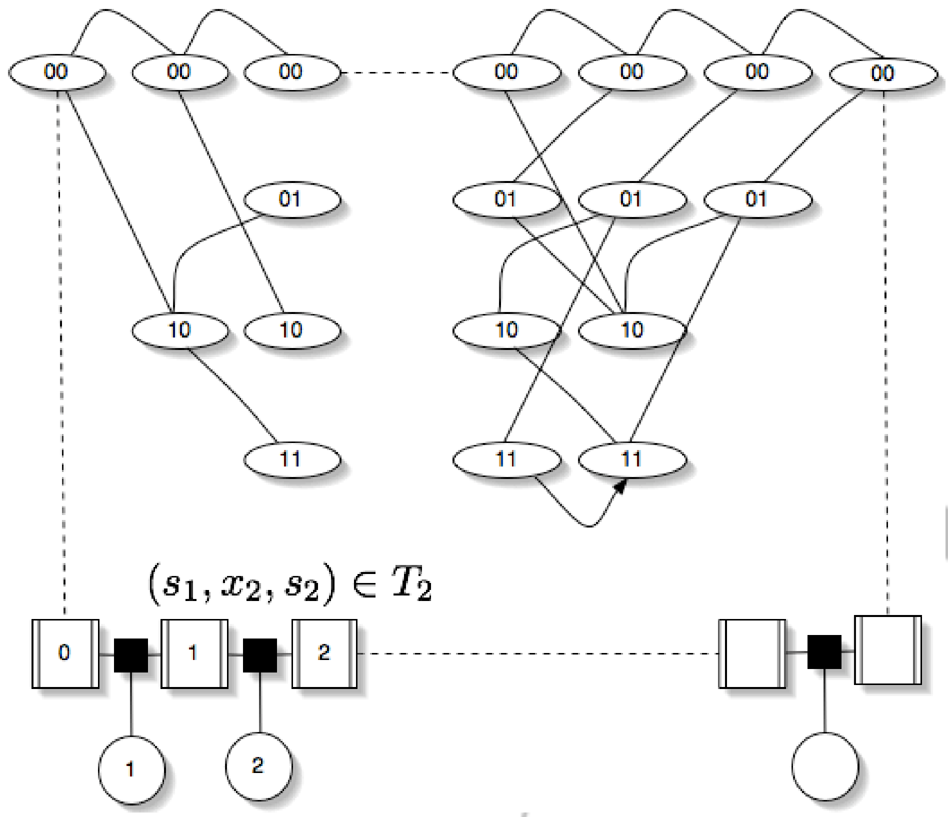
- Modèle d'état
  - Représentation des transitions autorisées



Approche comportementale

Définition du comportement par un treillis

- Section i :
  - Arêtes (i) vers (i-1) : variable visible
  - Comportement local : (s_{i-1}, x_i, s_i)
  - Les contrôles sont les indicatrices des comportements locaux.
- Treillis : comportement dans l'espace *s,x*



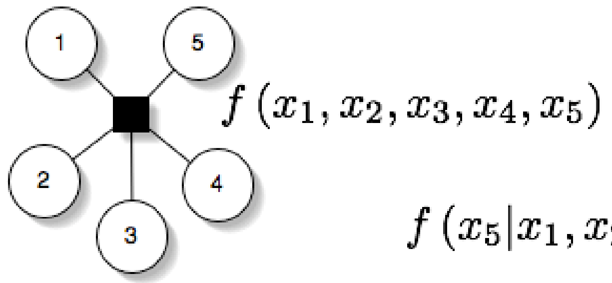
Approche probabiliste

- Représentation de distributions de probabilités :
  - Expression des indépendances et des indépendances conditionnelles.
  - Exemples :
    - Calcul des probabilités *a posteriori* en codage.
    - Chaînes de Markov cachées.

Approche probabiliste
Quelques factorisations courantes

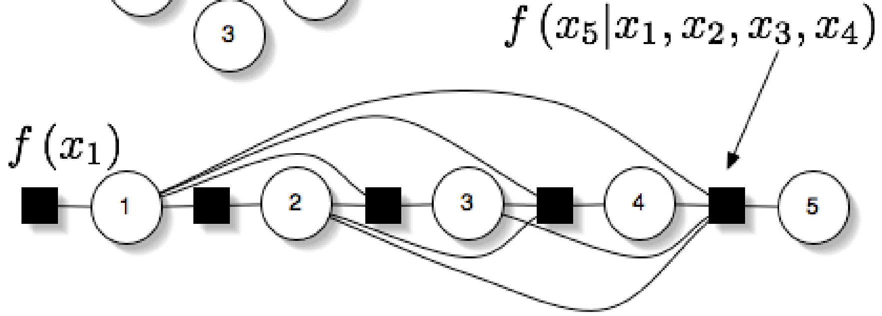
- Forme directe

$$f\left(x_1, x_2, \cdots, x_n\right)$$



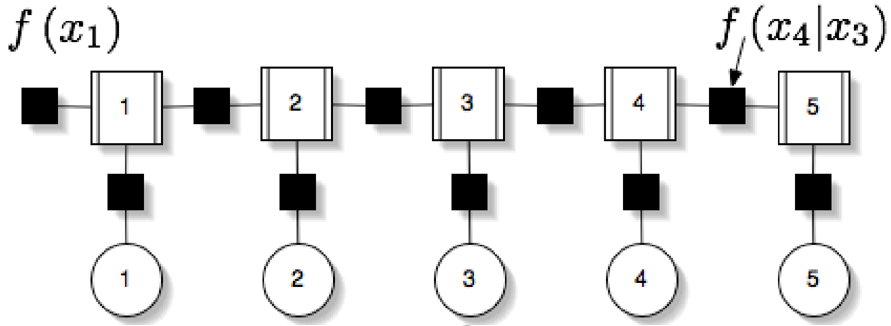
- Bayes

$$f\left(x_1, x_2, \cdots, x_n\right) = \prod_{j=1}^{n} f\left(x_j \mid x_1, \cdots, x_{j-1}\right)$$

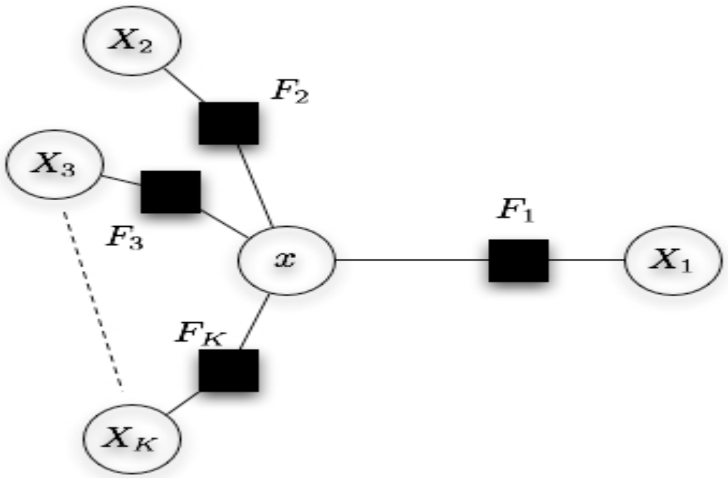


- Markov

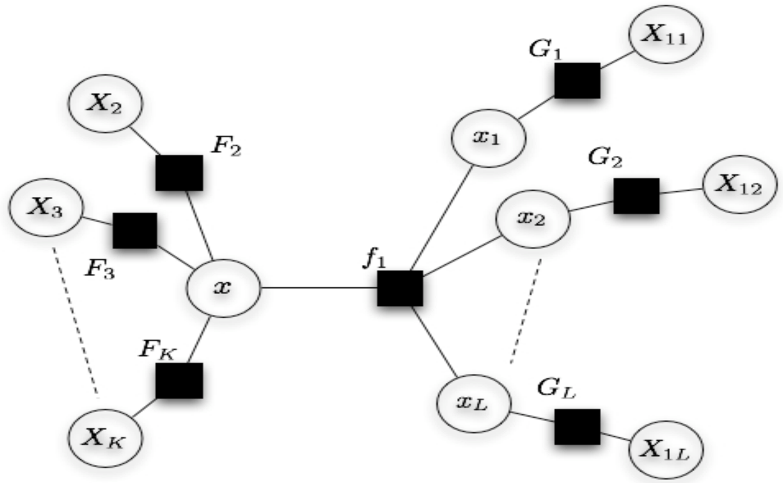
$$f\left(x_1, x_2, \cdots, x_n\right) = \prod_{j=1}^{n} f\left(x_j \mid x_{j-1}\right)$$



Marginalisation récursive de



$X_1,\cdots,X_K$ est une partition de $x_1,\cdots,x_{N-1}$



$x_1,\cdots,x_L,X_{11},\cdots,X_{1L}$ partition de X_1

$$\sum_{\sim x} g(x,x_1,\cdots,x_{N-1}) = \left(\sum_{X_1} F_1(x,X_1)\right)\cdots\left(\sum_{X_K} F_K(x,X_K)\right)$$

$$= \prod_{i=1}^{K} \sum_{\sim x} F_i(x,X_i)$$

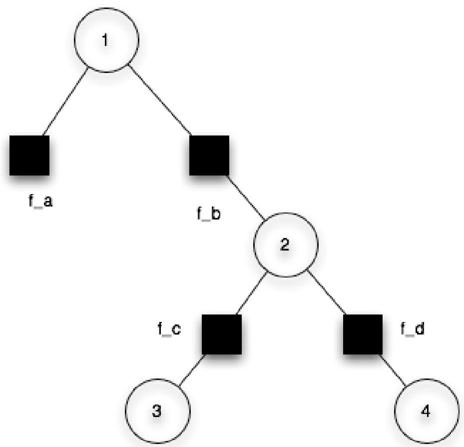
$$\sum_{\sim x} F_1(x,X_1) = \sum_{\sim x} f_1(x,x_1,\cdots,x_L)\, G_1(x_1,X_{11})\cdots G_L(x_L,X_{1L})$$

$$= \sum_{x_1,\cdots,x_L} f_1(x,x_1,\cdots,x_L)\left(\sum_{X_{11}} G_1(x_1,X_{11})\right)\cdots\left(\sum_{X_{1L}} G_L(x_L,X_{1L})\right)$$

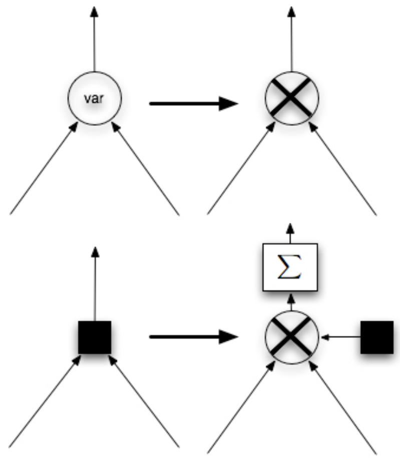
$$= \sum_{\sim x}\left(f_1(x,x_1,\cdots,x_L)\prod_{i=1}^{L}\sum_{\sim x_i} G_i(x_i,X_{1i})\right)$$

Arbre factoriel : arbre de calcul

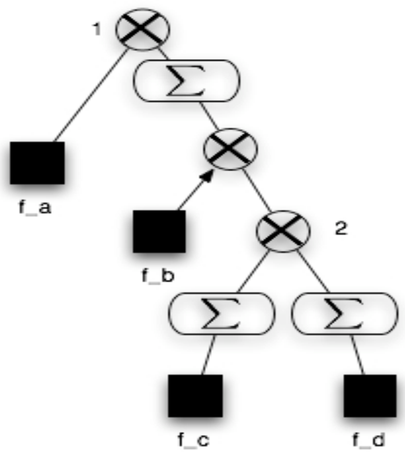
Arbre factoriel



Transformations élémentaires



Arbre de calcul



Le message d'un nœud *v* vers une arête *e* est le produit de la fonction locale en *v* (*Id* pour nœud de type variable) par le résumé en *e* des messages reçus par les autres arêtes.

Cas particuliers

Les nœuds de type variable de degré deux ne font rien.

Le résumé est superflu pour les fonctions de une seule variable.

Arbre factoriel : arbre de calcul

- Pour un arbre, le graphe encode :
  - la factorisation
  - L'algorithme de calcul des marginales :
    - Récursif du haut vers le bas.
    - Calcul du bas vers le haut.

- Vision imagée du fonctionnement :
    - Les nœuds sont des processeurs qui communiquent entre eux par des canaux (arêtes)
    - Les messages décrivent des marginales.

Marginalisation

Algorithme « somme-produit » pour un noeud

- Le nœud *i* est pris comme racine
  - Chaque variable feuille envoie la fonction identité à son père.
  - Chaque fonction feuille envoie la description de f à son père.

- Les nœuds attendent les messages de tous leurs enfants pour calculer le message à destination du père :
  - Un nœud de type variable envoie le produit des messages provenant de ses enfants.
  - Un nœud de type fonction envoie vers son père *x* le résumé en *x* du produit par f des messages provenant de ses enfants.

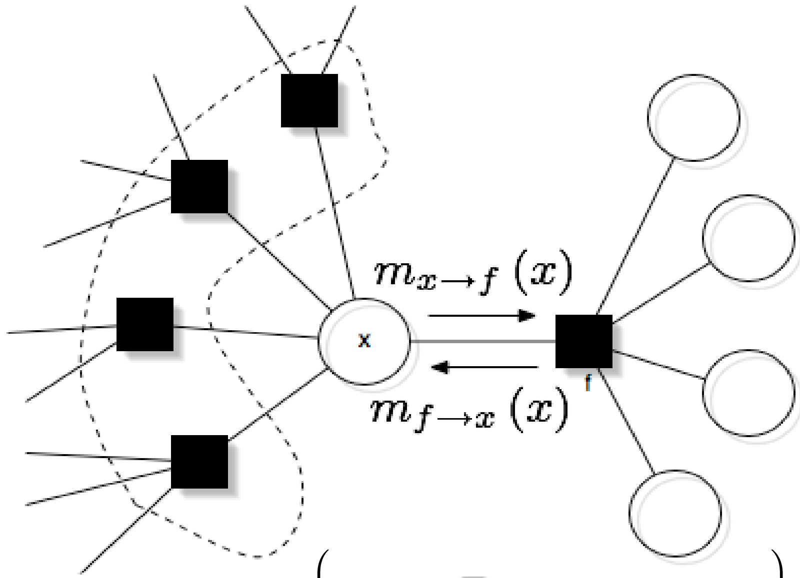
Marginalisations multiples

Algorithme « somme-produit »

•L'application de l'algorithme à l'ensemble des variables est (très) redondante :

$$m_{x\to f}(x) = \prod_{w\in n(x)\setminus\{f\}} m_{h\to x}(x)$$

- •Il suffit de calculer deux messages par arête.
- •Une seule règle : le message d'un nœud *v* vers une arête *e* est le produit de la fonction locale en *v* (*Id* pour nœud de type variable) par le résumé en *e* des messages reçus par les autres arêtes.



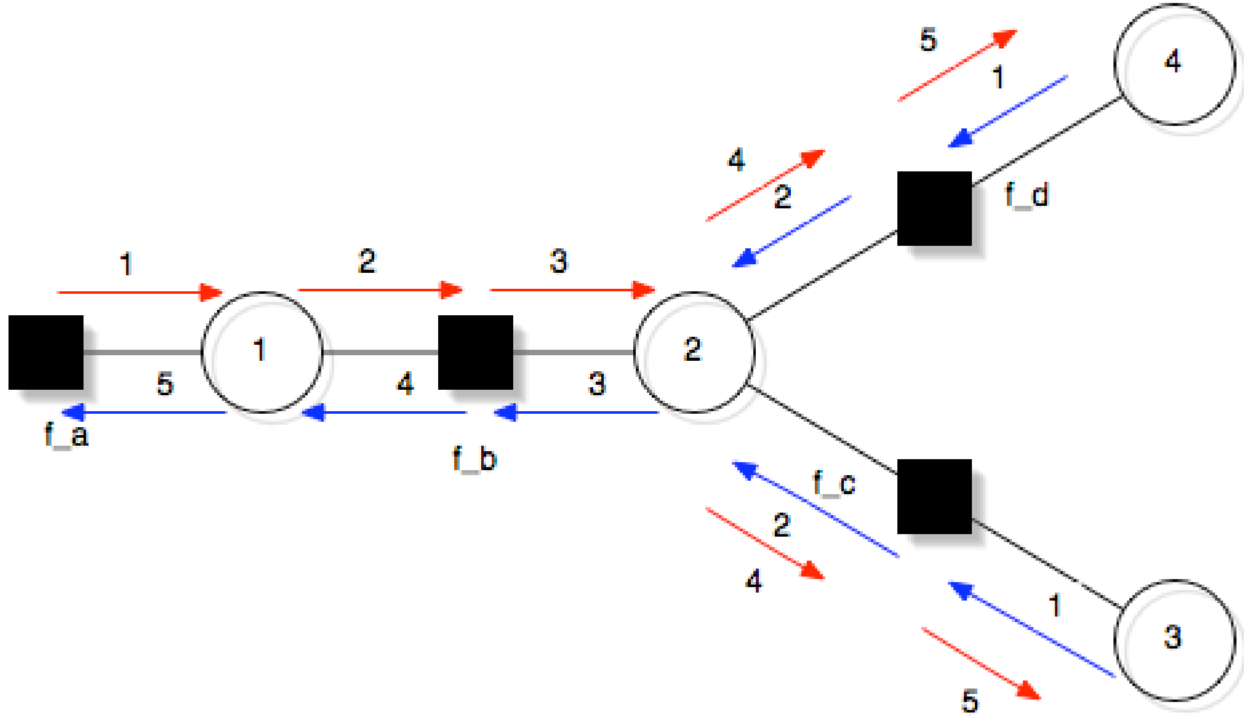
$$m_{f\to x}(x) = \sum_{\sim x}\left(f(X) \prod_{w\in n(f)\setminus\{x\}} m_{w\to f}(w)\right)$$

$$X = n(f)$$

Algorithme « somme-produit ».

Exemple.

- On commence aux feuilles



Quelques cas particuliers célèbres

Algorithme SP en signal, IA, communication

- Sur des chaînes :
  - Algorithme « forward/backward » (BCJR, MAP)
  - Algorithme de Viterbi bidirectionnel
  - Lisseur de Kalman
- Sur des arbres :
  - Algorithme « Belief Propagation » de Pearl.
  - Décodeurs itératifs (turbo-codes, LDPC). Instance de l'algorithme BP sur un graphe à cycles longs.
  - Certains algorithmes de FFT

Algorithme MAP [BCJR, forward/backward …]

Modèle mixte comportemental/probabiliste

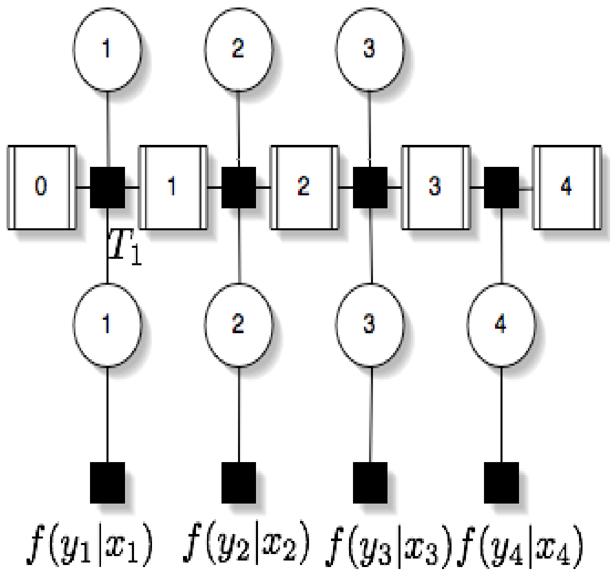
Entrées *u*

Etats *s liés par contrôles locaux*

$T_i\left(s_{i-1}, x_i, u_i, s_i\right)$

Sorties *x*

Observations



- Loi conjointe sachant l'observation :

$$g_y\left(u, s, x\right) = \prod_{i=1}^{n} T_i\left(s_{i-1}, x_i, u_i, s_i\right) \prod_{i=1}^{n} f\left(y_i \mid x_i\right)$$

- Probabilités proportionnelles aux fonctions marginales :

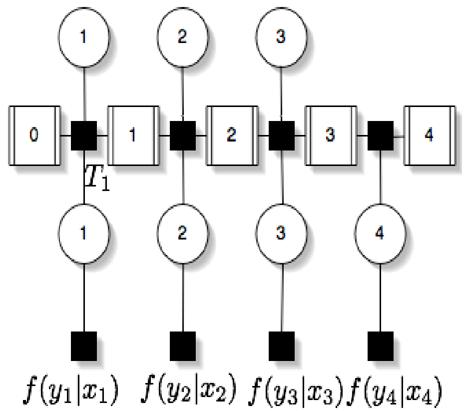
$$p\left(u_i \mid y\right) \propto \sum_{\sim u_i} g_y\left(u, s, x\right)$$

- Application possible de l'algorithme « somme-produit »

Algorithme MAP en tant que cas particulier de « somme-produit »

Algorithme générique | **Algorithme MAP**

$$
\begin{aligned}
m_{T_i \to u_i}(u_i) &\longleftarrow\cdots\longrightarrow \delta(u_i)\\
m_{s_i \to T_{i+1}}(s_i) &\longleftarrow\cdots\longrightarrow \alpha(s_i),\ \text{forward}\\
m_{s_i \to T_i}(s_i) &\longleftarrow\cdots\longrightarrow \beta(s_i),\ \text{backward}\\
m_{x_i \to T_i}(x_i) &\longleftarrow\cdots\longrightarrow \gamma(x_i)
\end{aligned}
$$



$$m_{f\to x}(x) = \sum_{\sim x}\left(f(X) \prod_{w\in n(f)\setminus\{x\}} m_{w\to f}(w) \right)$$

$$X = n(f)$$

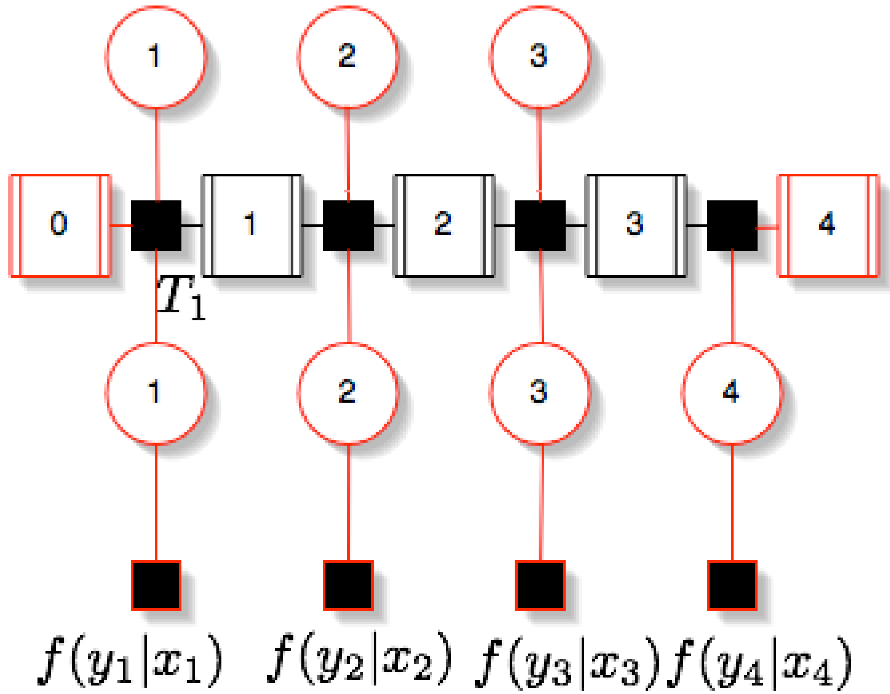
$$\alpha(s_i) = \sum_{\sim s_i} T_i(s_{i-1}, x_i, u_i, s_i)\,\alpha(s_{i-1})\gamma(x_i)$$

$$\beta(s_{i-1}) = \sum_{\sim s_{i-1}} T_i(s_{i-1}, x_i, u_i, s_i)\,\beta(s_i)\gamma(x_i)$$

$$\delta(u_i) = \sum_{\sim u_i} T_i(s_{i-1}, x_i, u_i, s_i)\,\alpha(s_{i-1})\gamma(x_i)$$

Algorithme MAP, BCJR, forward/backward …

- Etape 1 : initialisation
  - Initialisation des feuilles.
  - Les nœuds de type variable d'ordre 2 ne font que transmettre.
  - Deux récurrences parallèles :
    - directe et rétrograde.

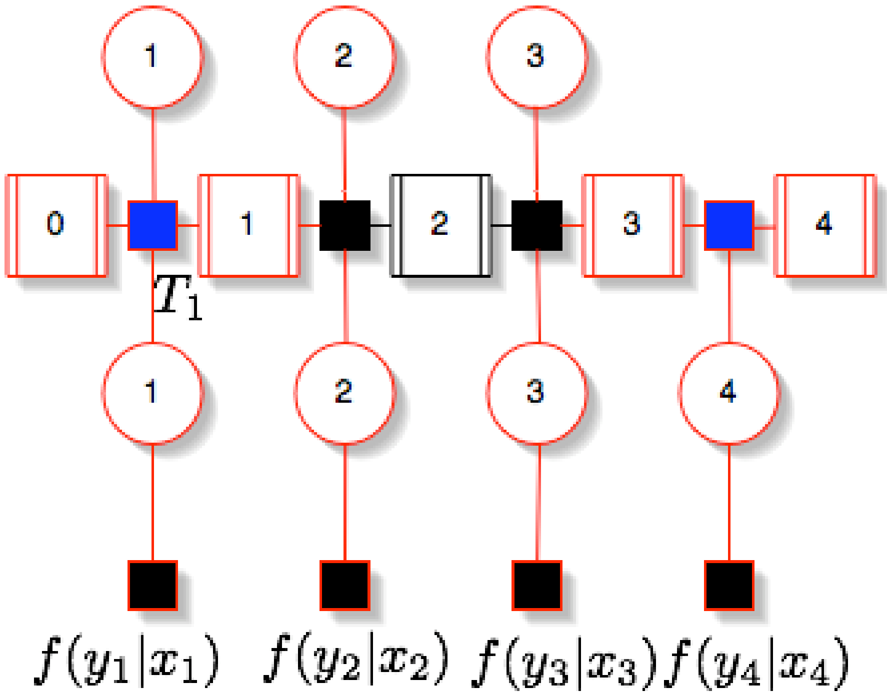


Algorithme MAP, BCJR, forward/backward …

- Etape 2 :
  - Propagation des messages directs et rétrogrades

$$\alpha(s_i) = m_{s_i \to T_{i+1}}(s_i) = \sum_{e \in E_i(s_i)} \alpha(e)\gamma(e)$$

$$\beta(s_{i-1}) = m_{s_i \to T_i}(s_i) = \sum_{e \in E_i(s_{i-1})} \beta(e)\gamma(e)$$



Turbo-synchronisation de phase

- Observation d'un mot du code :

Mot code

Phase

Bruit

$$y_k = x_k e^{i\theta_k} + b_k,\ k = 0 \cdots N-1$$

Observation

But : sachant les observations, retrouvez les symboles :

$$p(\mathbf{a},\theta \mid \mathbf{y}) \propto p(\mathbf{a})\, p(\theta)\, p(\mathbf{y} \mid \mathbf{a},\theta)$$
$$= [\mathbf{x} \in C]\, p(\theta)\, p(\mathbf{y} \mid \mathbf{x},\theta)$$

A priori

Vraisemblance

Turbo-synchronisation de phase

- Observation

$$y_k = x_k e^{i\theta_k} + n_k,\ \ k = 0 \cdots N-1$$

x_k version codée des symboles a_k

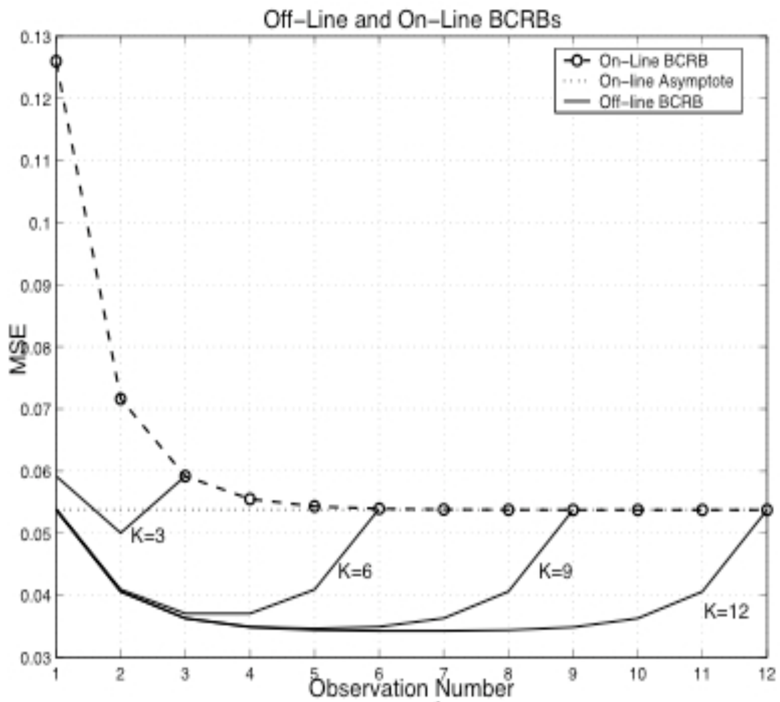
- *A priori*
  - *Phase (Markov caché)*

$$\theta_k = \theta_{k-1} + w_k,\ \mathrm{w_k} \sim N\left(0, \sigma_w^2\right) \text{ i.e. } p(\theta)$$

  - *Code*

$$1_C\left(x_1, \cdots, x_n\right) = \left[\left(x_1, \cdots, x_n\right) \in C\right]$$

Bornes Cramer Rao Bayésiennes en ligne et hors ligne

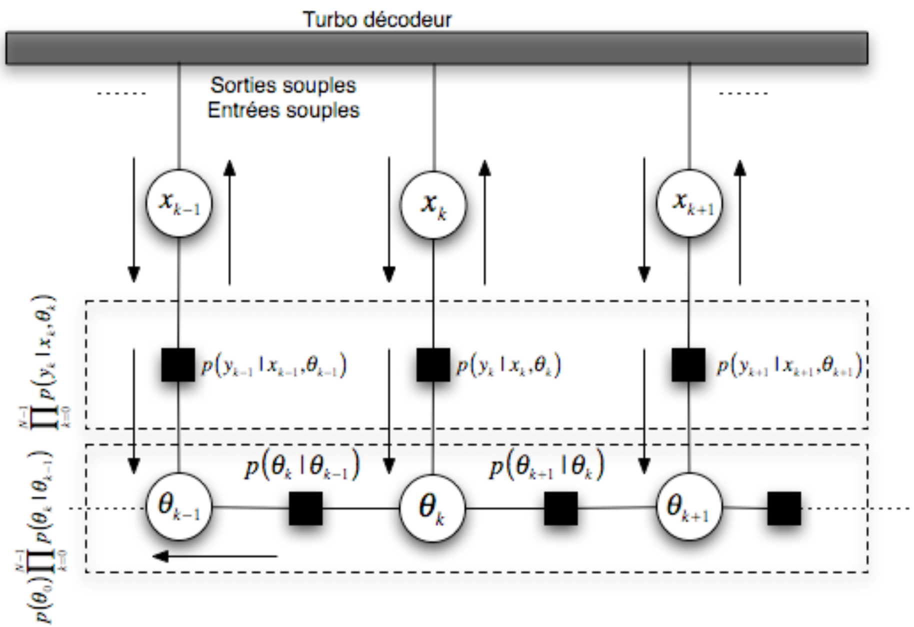


Turbo-synchronisation de phase
Modèle graphique du problème

- Complexité rédhibitoire de la solution optimale

$$p(\mathbf{a},\theta \mid \mathbf{y}) \propto [\mathbf{x}\in C]\, \underbrace{p(\theta_0)\prod_{k=0}^{N-1} p(\theta_k \mid \theta_{k-1})}_{\text{Factorisation de la trajectoire de phase.}} \underbrace{\prod_{k=0}^{N-1} p(y_k \mid x_k,\theta_k)}_{\text{Factorisation de la vraisemblance.}}$$

- Le graphe présente des cycles.
- La résolution par propagation de croyance itérative (BP) fournit une bonne approximation.



Turbo-estimation de phase
Mise en œuvre pratique

- Voies vers une implantation de l'algorithme
  - Discrétiser la phase
  - Paramétrer les densités de probabilités.
  - Méthodes particulaires.

Bloc ou séquentiel ?

- Traitements « par bloc » :
  - Complexes et performants
  - Naturels dans les normes récentes
  - Hypothèses rigides (paramètres constants, …)

- Traitements séquentiels type algorithmes adaptatifs :
  - Simples et peu performants
  - Naturels en « flot continu »
  - Hypothèses peu contraignantes : robustesse

Trends

- Iterative algorithms
  - Turbo-codes
  - Turbo everything

Coding

- **Emitted/Received word:**

$$X=(X_1,\cdots,X_n)\rightarrow\begin{cases}Y=(Y_1,\cdots,Y_n) \text{ received word}\\ \alpha=(\alpha_1,\cdots,\alpha_n) \text{ reliability}\end{cases}$$

Error vector: $Z^m=Y\oplus X^m$

Weight of the error vector: $W(Z^m)=\sum_{i=1}^{n} Z_i^m$

- **Decoder**

Incomplete decoder: codeword $X^m=(X_1^m,\cdots,X_n^m)$ close (Hamming) to $Y=(Y_1,\cdots,Y_n)$ / $W(Z^m)\leq\left\lfloor\frac{d_{\min}-1}{2}\right\rfloor$

- one word if $W(Z^m)\leq\left\lfloor\frac{d_{\min}-1}{2}\right\rfloor$ else no word

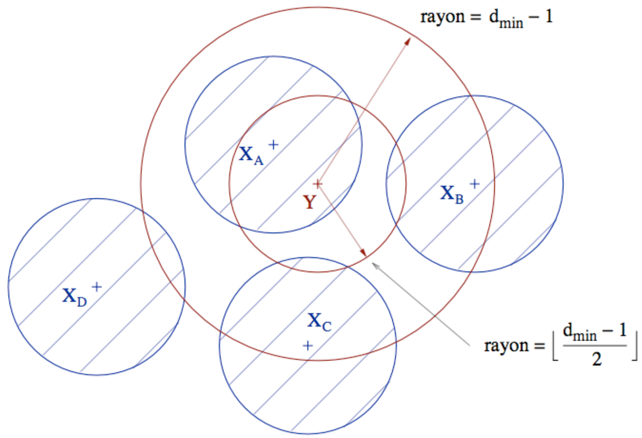
- Right decision if the number of errors is less than $\left\lfloor\frac{d_{\min}-1}{2}\right\rfloor$

Complete decoder: $\min_m W(Y\oplus X^m)$ can provide a codeword even if the number of errors is greater than $\left\lfloor\frac{d_{\min}-1}{2}\right\rfloor$

Complete soft decoder: $\min_m W_\alpha(Y\oplus X^m)$ with analog weight $W_\alpha(Z^m)=\sum_{i=1}^{n}\alpha_i Z_i^m$

Chasing

- Hard decoding: X_A
- Use the hard decoder to find a small number of words and select the minimum analog distance
  - Modify Y using test sequences: Y'
  - Select minimum Euclidian distance



$$\left|\mathbf{R} - \mathbf{C}^i\right|^2 \leq \left|\mathbf{R} - \mathbf{C}^l\right|^2 \quad \forall l \in [1, 2^k]$$

Reliability

- For a bit *j*

$$\Lambda(d_j) = \log\left(\frac{P(a_j = +1|R)}{P(a_j = -1|R)}\right)$$

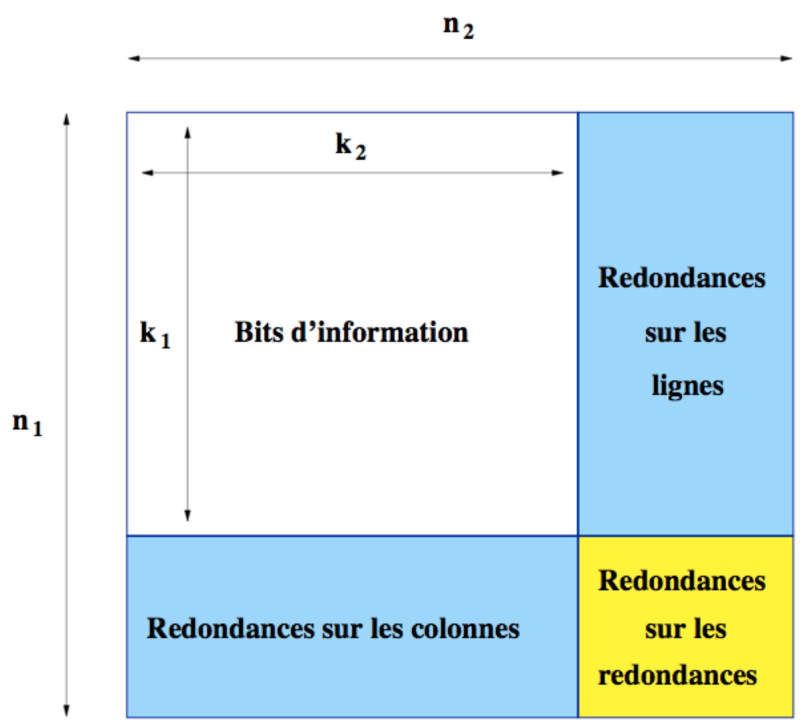
$$P(a_j = \pm 1|R) = \sum_{C^i \in S_j^{\pm 1}} P(E = C^i|R) \text{ with } S_j^{\pm 1} \text{ set of word / } c_j^i = \pm 1$$

$$\Lambda(d_j) = \log\left(\frac{\sum_{C^i \in S_j^{+1}} P(R|E = C^i)}{\sum_{C^i \in S_j^{-1}} P(R|E = C^i)}\right) \text{ with } P(R|E = C^i) = \left(\frac{1}{\sqrt{2\pi\sigma}}\right)^n \exp\left(-\frac{|R - C^i|^2}{2\sigma^2}\right)$$

$$\Lambda(d_j) \approx \frac{1}{2\sigma^2}\left(\left|R - C^{-1(j)}\right|^2 - \left|R - C^{+1(j)}\right|^2\right)$$

$C^{\pm 1(j)}$ codewords in $S_j^{\pm 1}$ at minimum Euclidian distance / R

Product code



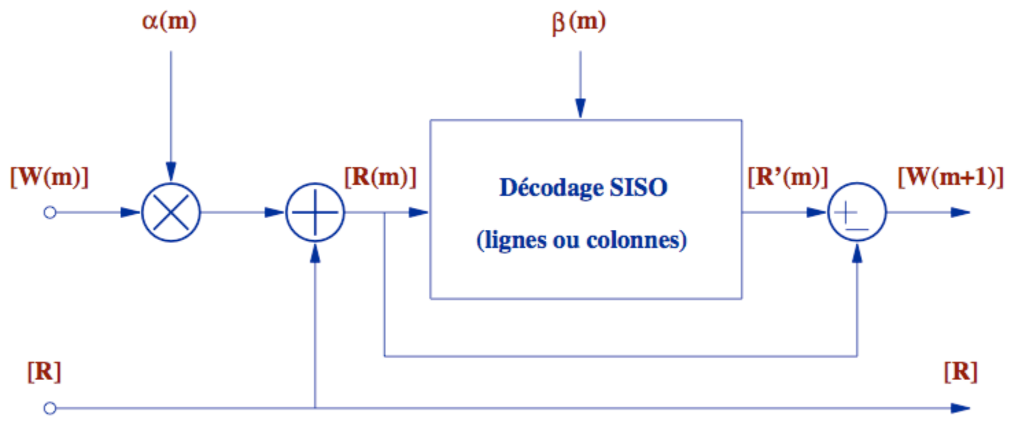
Iterative decoding

$$R = (r_1, \cdots, r_n) = E + G$$

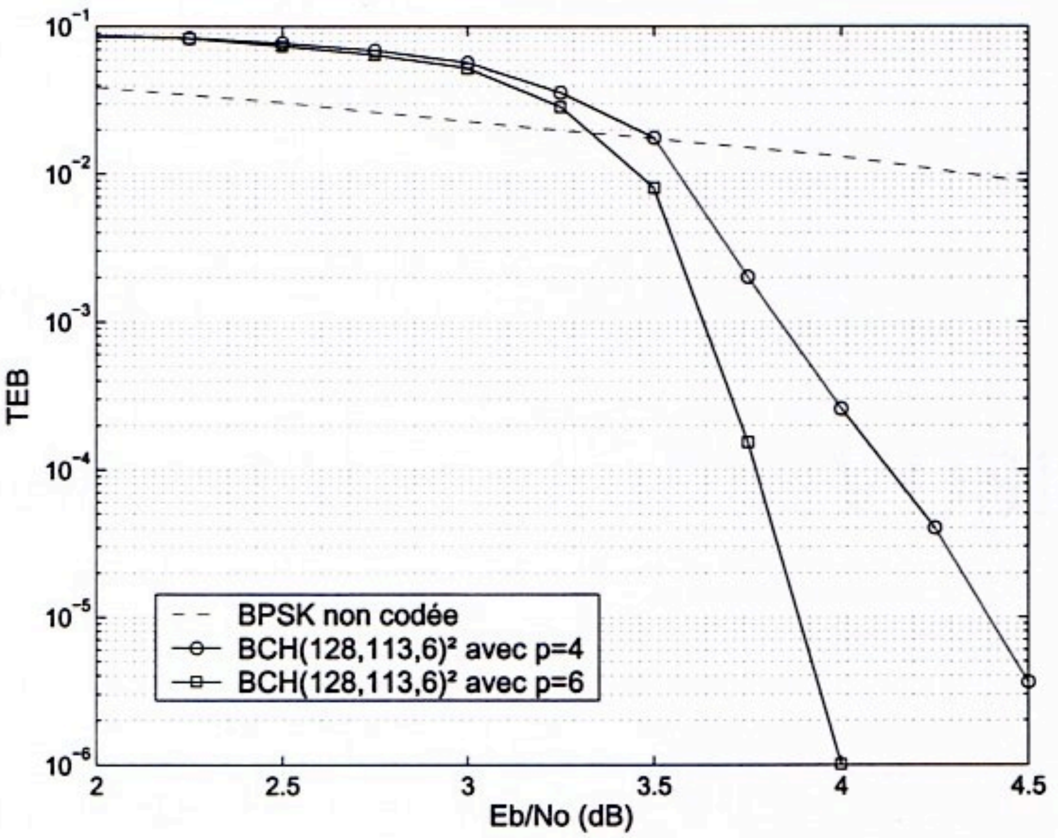
E : transmitted codeword

G : gaussian noise

- Decision: $|\mathbf{R} - \mathbf{C}^i|^2 \leq |\mathbf{R} - \mathbf{C}^l|^2 \quad \forall l \in [1, 2^k]$



Gain de codage



Turbo

- Concatenation of SISO (soft input soft output) algorithms can be extended to all blocks (equalization, multi-user detection, synchronization, …)