APPLICATION OF THE TURBO PRINCIPLE

Didier Le Ruyet[†]

[†] Electronique et Communications, CNAM, 292 rue Saint Martin, 75141 Paris Cedex 3, France

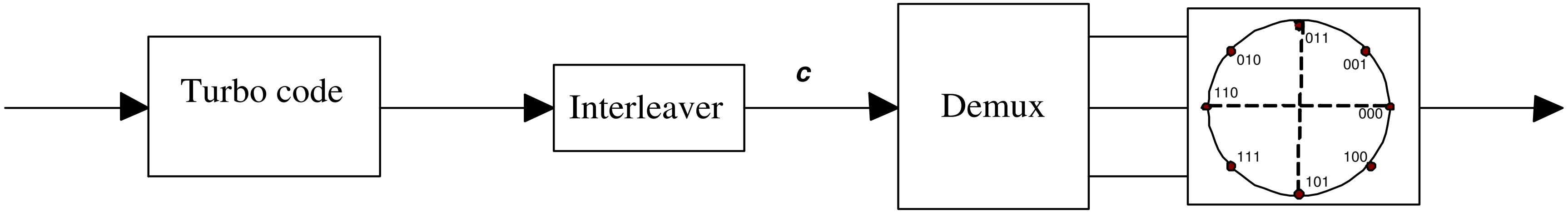
Email: leruyet@cnam.fr

Outline

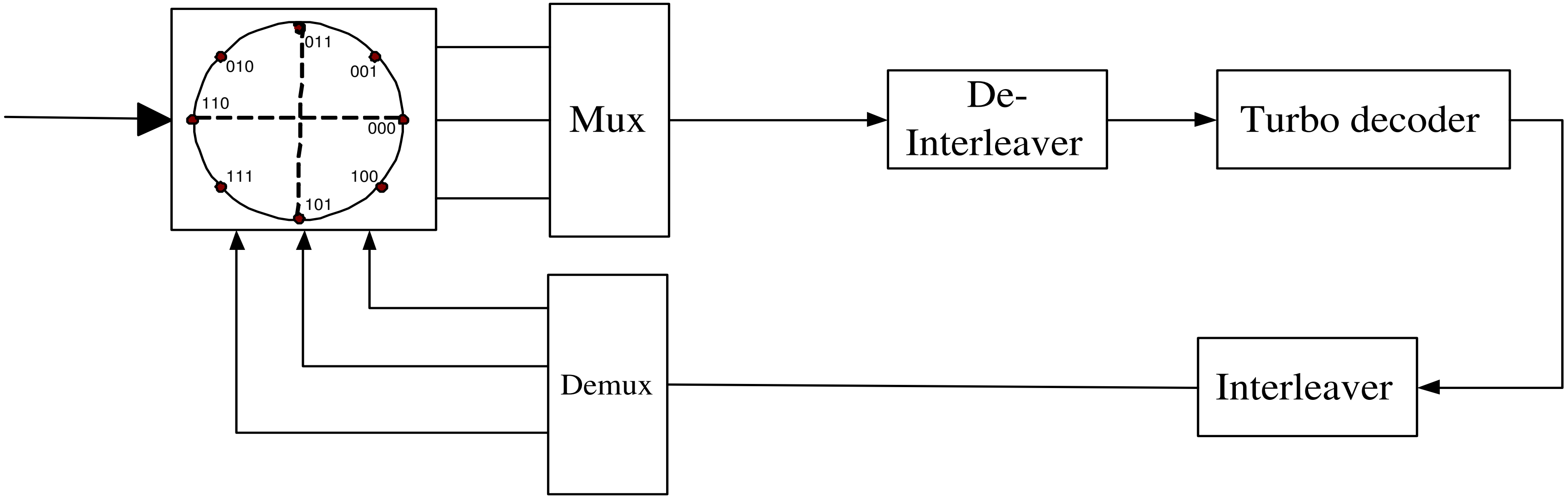
- Bit Interleaved Coded Modulation
- Turbo equalization
- Turbo V-BLAST

Bit Interleaved Coded Modulation

- Encoding



- Iterative Decoding



Bit Interleaved Coded Modulation

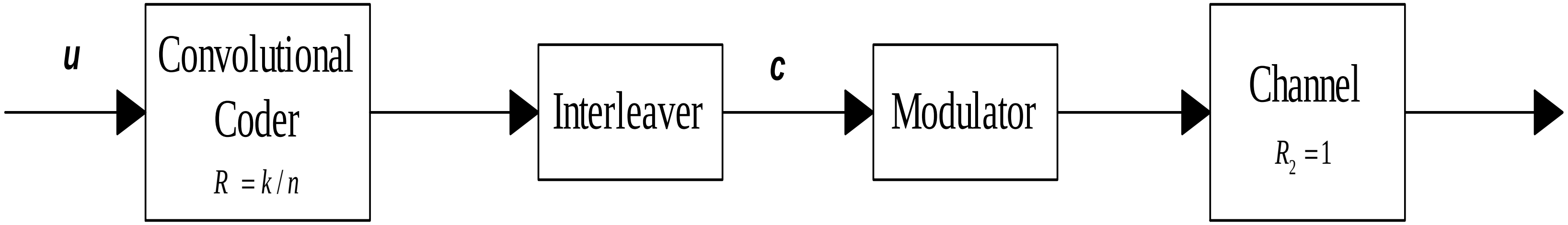
- The demapper generates a posteriori probability or log likelihood ratio about the transmitted bits c_i

$$L(c_i|\mathbf{y}) = \ln \frac{P(c_i = +1)}{P(c_i = -1)} + \ln \frac{\sum_{\mathbf{c}:c_i=0} p(\mathbf{y}|\mathbf{c})P(\mathbf{c}|c_i)}{\sum_{\mathbf{c}:c_i=0} p(\mathbf{y}|\mathbf{c})P(\mathbf{c}|c_i)}$$

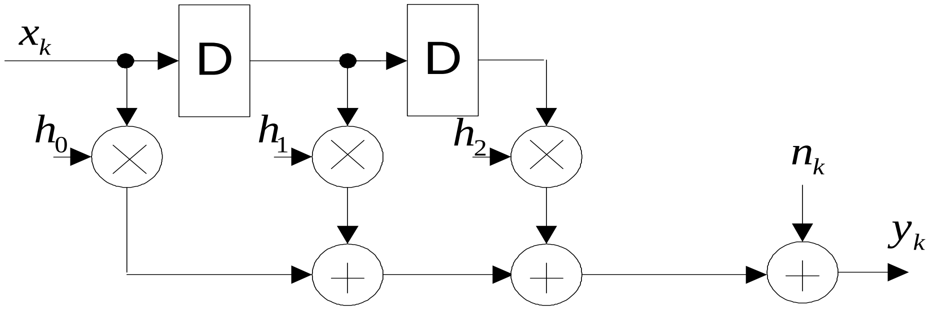
- Assuming the independence of the bits c_i we can write :

$$P(\mathbf{c}|c_i) = \prod_{j \neq i} P(c_j) \qquad (1)$$

Turbo equalization

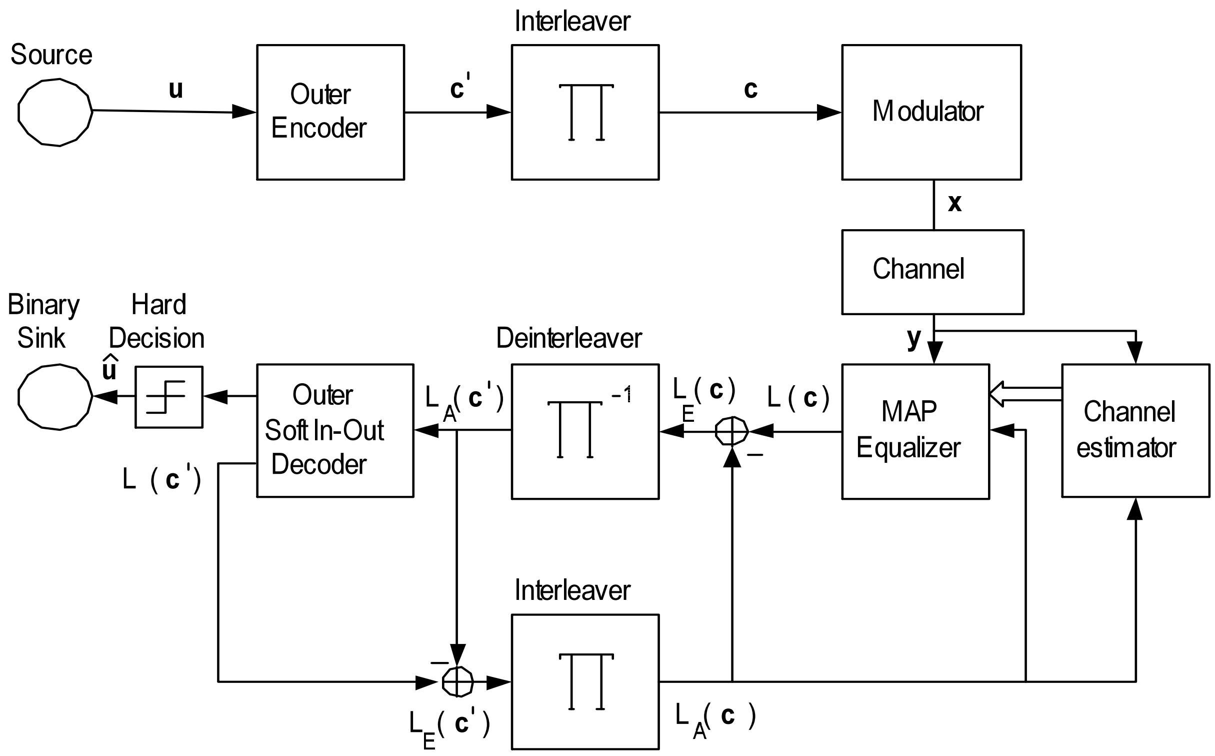


- Channel model



$$y_k = \sum_{l=0}^{L} h_l x_{k-l} + n_k \qquad k = 0, 1, \ldots, N-1 \tag{2}$$

Turbo equalization



- Joint equalization and decoding using the turbo principle
- First proposed by Douillard and als in 1997

MAP equalizer

The a posteriori derivation is given by :

$$APP(x_t^S = a) \propto \sum_{m',m/x_t^S=a} \alpha_t(m')\gamma_t(m',m)\beta_{t+1}(m)$$

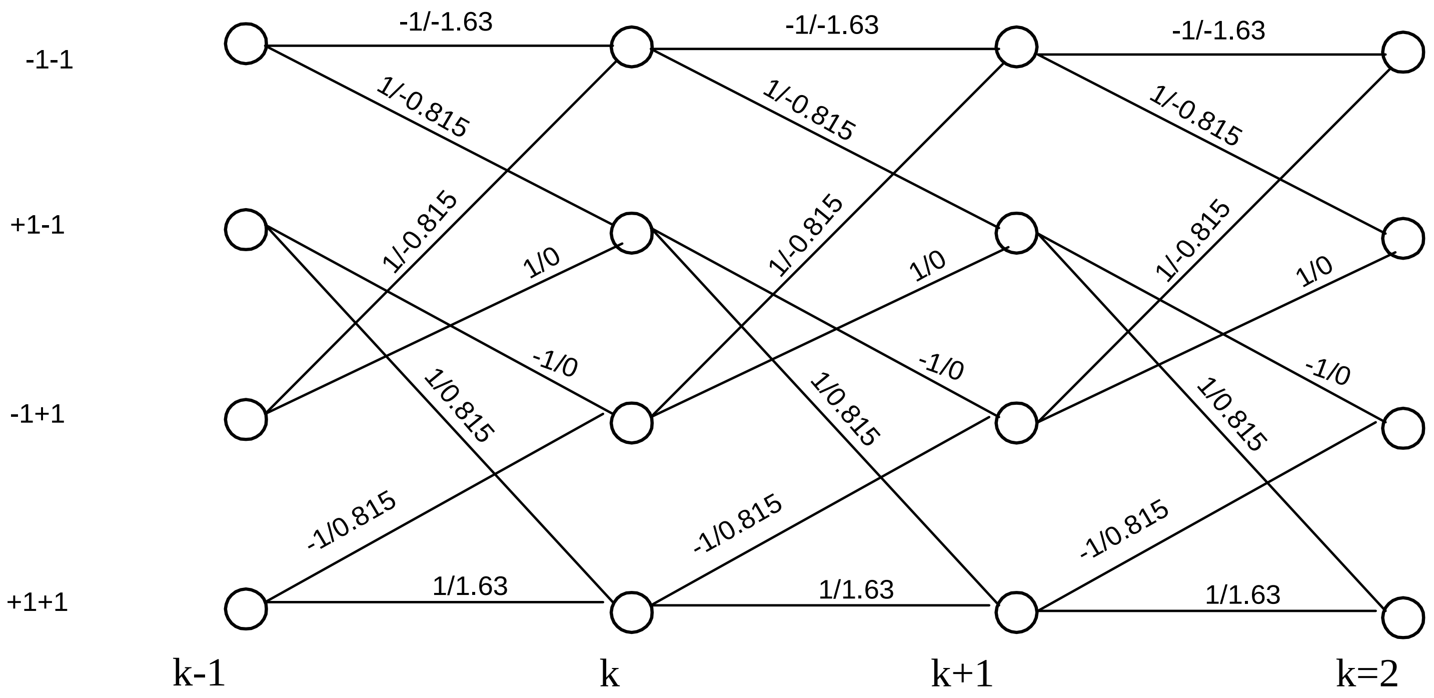
with

$$\begin{aligned}
\gamma_t(m',m) &= Pr\{s_{t+1} = m; \mathbf{y}_t | s_t = m'\} \\
&= Pr\{s_{t+1} = m | s_t = m'\} \times Pr\{\mathbf{y}_t | s_{t+1} = m; s_t = m'\} \\
&= APRI(x_t = x_t(m',m)) \times p(y_t \mid s_{t+1} = m, s_t = m') \\
&= APRI(x_t = x_t(m',m)) \times p(y_t \mid v_t)
\end{aligned}$$

$$\text{with} \quad v_t = \sum_{i=0}^{L} \tilde{x}_{t-i} h_i \qquad \text{and} \qquad p(y_t \mid v_t) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(- \frac{(y_k - v_k)^2}{2\sigma^2} \right) \qquad (3)$$

$\tilde{x}_{t-i}$ is the value of x on the trellis path at the previous time $t - i$.

Trellis representation of the channel



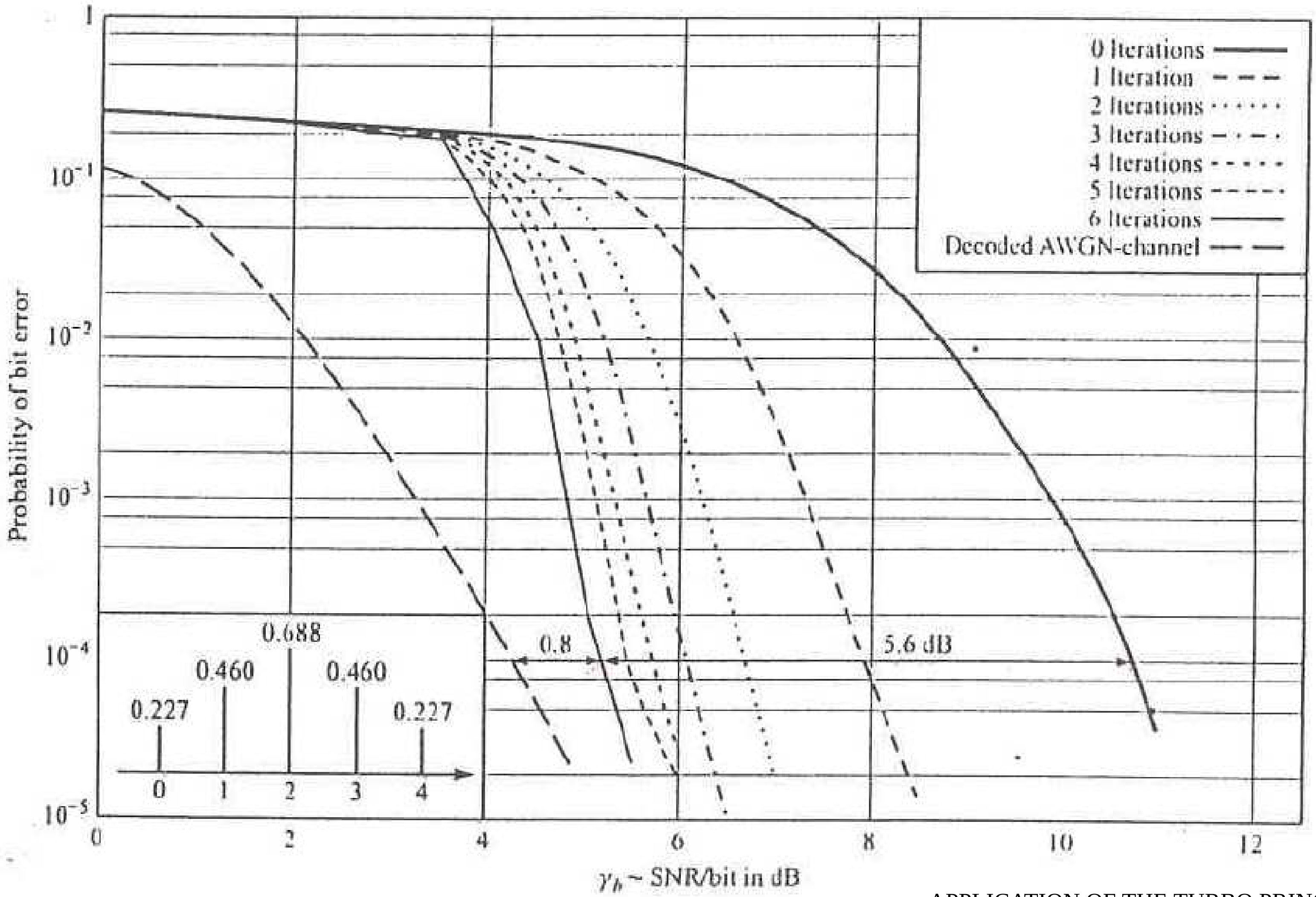
Complexity issue

- The complexity of the MAP decoder is determined by the number of trellis states Q^L where L is the number of delay elements and Q is the size of the symbol constellation.

- Some reduced complexity methods such as M-BCJR, T-BCJR.

Performance

- RSC code ($R = 1/2$, $m = 4$)
- Block length= 4096 bits



MMSE detector

- Linear filter based approach
- The operations are performed sequentially on a subset $\mathbf{y}_k$ of M observed symbols y_k.
- The channel equation can be expressed in a matrix form :

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

For the considered length 3 channel, we have the relation :

$$\begin{pmatrix} y_{k-4} \\ \vdots \\ y_{k+6} \end{pmatrix} = \begin{pmatrix} h_2 & h_1 & h_0 & 0 & 0 & \dots & 0 \\ 0 & h_2 & h_1 & h_0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \\ 0 & 0 & \dots & 0 & h_2 & h_1 & h_0 \end{pmatrix} \begin{pmatrix} x_{k-6} \\ \vdots \\ x_{k+6} \end{pmatrix} + \begin{pmatrix} n_{k-4} \\ \vdots \\ n_{k+6} \end{pmatrix} \quad (4)$$

$\tilde{\mathbf{H}}$ is a $M \times (M+L)$ matrix

MMSE detector

- Computes an affine estimate of x_k at time k:

$$\hat{x_k} = \mathbf{f}_k^H \mathbf{y}_k + \mathbf{b}_k$$

such that the mean square error (MSE) is mimimum : :

$$\hat{x_k} = \arg \min_{\mathbf{f}_k, \mathbf{b}_k} E|x_k - \hat{x_k}|^2$$

This is achieved by :

$$\begin{aligned} \mathbf{f}_k &= cov(\mathbf{y}_k, \mathbf{y}_k)^{-1} cov(\mathbf{y}_k, x_k) \\ b_k &= E(x_k) - \mathbf{f}_k^H \tilde{\mathbf{H}} E(\mathbf{x}_k) \end{aligned} \tag{5}$$

where

$cov(\mathbf{y}_k, x_k) = \tilde{\mathbf{H}} cov(\mathbf{x}_k, x_k)$

$cov(\mathbf{y}_k, \mathbf{y}_k) = \tilde{\mathbf{H}} cov(\mathbf{x}_k, \mathbf{x}_k) \tilde{\mathbf{H}}^H + \sigma^2 \mathbf{I}$

MMSE detector

- We get the following MMSE solution :

$$\hat{x_k} = E(x_k) + \mathbf{f}_k^H(\mathbf{y}_k - \tilde{\mathbf{H}}E(\mathbf{y}_k))$$

$$\hat{x_k} = E(x_k) + cov(x_k, \mathbf{x}_k)\tilde{\mathbf{H}}^H(\tilde{\mathbf{H}}cov(\mathbf{x}_k, \mathbf{x}_k)\tilde{\mathbf{H}}^H + \sigma^2\mathbf{I})^{-1}(\mathbf{y}_k - \tilde{\mathbf{H}}E(\mathbf{x}_k)) \tag{6}$$

- The expectation of x_k is estimated using the available *a priori* information

$$\begin{aligned} E(x_k) &= \sum_{x \in \mathcal{X}_S} x.Pr\{x_k = x\} \\ &= \tanh\left(\frac{1}{2}L_{APRI}(x_k)\right) \qquad \text{in the binary case} \end{aligned}$$

using the relation $\quad \tanh\left(\frac{1}{2}ln\left(\frac{1-a}{a}\right)\right) = 1 - 2a$

MMSE detector

- Since the bits x_k are independent, $cov(x_k, x_l) = 0 \quad \text{for} \quad k \neq l$

$$cov(x_k, \mathbf{x}_k) = (0, \ldots, 0, \mathrm{cov}(x_k, x_k), 0, \ldots, 0) \tag{7}$$

$$\mathrm{cov}(\mathbf{x}_k, \mathbf{x}_k) = \mathrm{diag}(\mathrm{cov}(x_{k-6}, x_{k-6}), \ldots, \mathrm{cov}(x_{k+6}, x_{k+6}))$$

$$cov(x_k, x_k) = \left(\sum_{x \in \mathcal{X}_S} x^2 . Pr\{x_k = x\} \right) - E(x_k)^2$$

MMSE detector

- In order to extract the extrinsic information we have to modify the previous equation in order to guarantee that $\hat{x}_k$ is not calculated using $L_{APRI}(x_k)$

$$\hat{x_k} = \sigma_x^2 \tilde{\mathbf{H}}_k^H (\tilde{\mathbf{H}} cov_k(\mathbf{x}_k, \mathbf{x}_k) \tilde{\mathbf{H}}^H + \sigma^2 \mathbf{I})^{-1} (\mathbf{y}_k - \tilde{\mathbf{H}} E_k(\mathbf{x}_k)) \tag{8}$$

where $\tilde{\mathbf{H}}_k$ is the kth column of $\tilde{\mathbf{H}}$,

$$\mathrm{E}_k(\mathbf{x}_k) = (\mathrm{E}(x_{k-6}), \ldots, \mathrm{E}(x_{k-1}), 0, \mathrm{E}(x_{k+1}), \ldots, \mathrm{E}(x_{k+6}))^H$$

$$\mathrm{cov}_k(\mathbf{x}_k, \mathbf{x}_k) = \mathrm{diag}(\mathrm{cov}(x_{k-6}, x_{k-6}), \ldots, \mathrm{cov}(x_{k-1}, x_{k-1}), \sigma_x^2, \mathrm{cov}(x_{k+1}, x_{k+1}), \ldots$$
$$\ldots, \mathrm{cov}(x_{k+6}, x_{k+6}))$$

Soft input Soft output MMSE detector

Finally the calculation of the extrinsic information is done assuming that the estimates $\hat{x}_q$ are the outputs of equivalent gaussian channels :

$$\hat{x}_k = \mu_k x_k + \eta_k$$

where

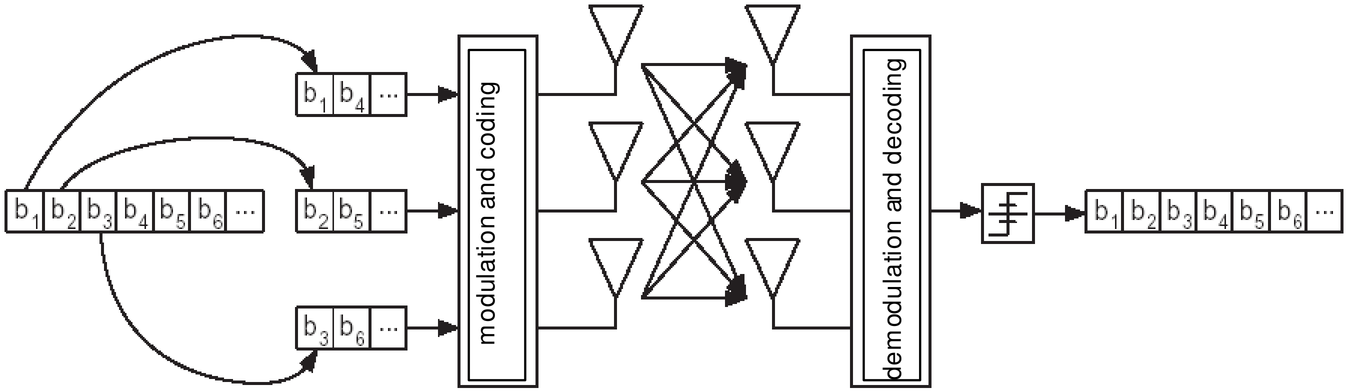
$$\mu_k = \sigma_x^2 \tilde{\mathbf{H}}_k^H (\tilde{\mathbf{H}} cov_k(\mathbf{x}_k, \mathbf{x}_k) \tilde{\mathbf{H}}^H + \sigma^2 \mathbf{I})^{-1} \tilde{\mathbf{H}}_k$$

$$\eta_k \quad \text{gaussian noise with variance} \quad \nu_k^2 = \sigma_x^2 (\mu_k - (\mu_k)^2)$$

Then :

$$L_{EXTR}(x_k) = \frac{2\hat{x}_k \mu_k}{\nu_k^2}$$

Spatial multiplexing V-BLAST Foschini99

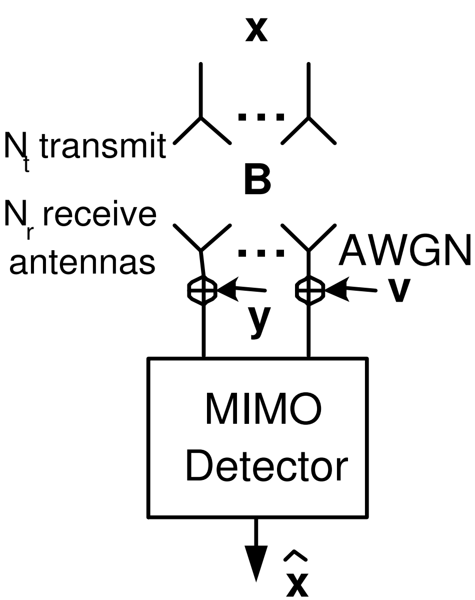


Example :$N_t = N_r = 2, Q = 2, T = 1$:

- The received signal is given by :

$$\begin{bmatrix} y_{11} \\ y_{21} \end{bmatrix} = \mathbf{H} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} n_{11} \\ n_{21} \end{bmatrix}$$

Turbo V-BLAST



- We consider a V-BLAST scheme with $N_t = N_r = N$
  - We assume iid MIMO channels
  - Then we have the classical real-valued equation:

$$\mathbf{y} = \mathbf{B}\mathbf{x} + \mathbf{v}$$

  - Assuming that $\mathbf{B}$ is known at the receiver

    the optimal decoder is the ML decoder:

$$\hat{\mathbf{x}} = \arg\min_x ||\mathbf{y} - \mathbf{B}\mathbf{x}||^2$$

  - Equivalent to the search of the closest point in a lattice

Sphere Decoder

$$
\begin{aligned}
\mathcal{M}(\mathbf{x}(\mathbf{c})) &= \|\mathbf{y} - \mathbf{B}\mathbf{x}(\mathbf{c})\|^2 \\
&= (\mathbf{x} - \tilde{\mathbf{x}})^T \mathbf{B}^T \mathbf{B} (\mathbf{x} - \tilde{\mathbf{x}}) + \mathbf{y}^T (\mathbf{I} - \mathbf{B}(\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T)\mathbf{y} \\
&= (\mathbf{x} - \tilde{\mathbf{x}})^T \mathbf{R}^T \mathbf{R} (\mathbf{x} - \tilde{\mathbf{x}}) + \mathbf{y}^T (\mathbf{I} - \mathbf{B}(\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T)\mathbf{y}
\end{aligned}
$$

where $\tilde{\mathbf{x}} = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{y}$ is the ZF estimate
$\mathbf{R} = \{r_{ij}\}_{2N \times 2N}$ is the upper triangular matrix with $\mathbf{B}^T\mathbf{B} = \mathbf{R}^T\mathbf{R}$ obtained using the Cholesky factorization.

- Instead of searching among the 2^{2N} points (QPSK modulation), we limit the search to the points situated inside an hypersphere of square radius C_1 and with center $\mathbf{y}$.

Metric calculation

$$\mathcal{M}(\mathbf{x}(\mathbf{c})) = \sum_{i=1}^{2N} w(\mathbf{x}_i^{2N}) + \mathcal{M}'$$

$$\text{where} \qquad w(\mathbf{x}_i^{2N}) = \left(q_{ii} \left(z_i + \sum_{j=i+1}^{2N} q_{ij} z_j \right) \right)^2$$

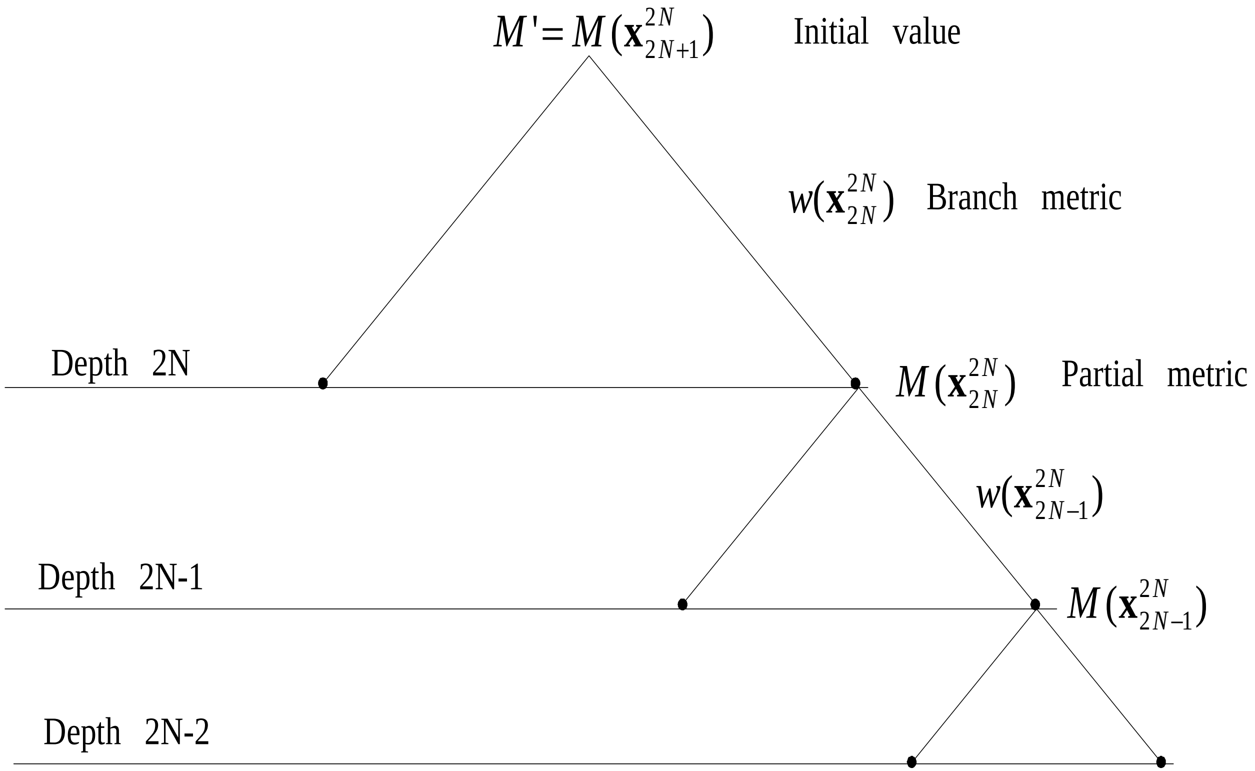
$$z_i = x_i - \tilde{x}_i, q_{ii} = r_{ii}^2, q_{ij} = \frac{r_{ij}}{r_{ii}} \quad \text{for} \quad j > i$$

- The metric can be calculated sequentially over the support tree :

$$\begin{aligned}
\mathcal{M}(\mathbf{x}_i^{2N}) &= \sum_{j=i}^{2N} w(\mathbf{x}_j^{2N}) + \mathcal{M}(\mathbf{x}_{2N+1}^{2N}) \\
&= \mathcal{M}(\mathbf{x}_{i+1}^{2N}) + w(\mathbf{x}_i^{2N})
\end{aligned}$$

with $\mathcal{M}(\mathbf{x}_{2N+1}^{2N}) = \mathcal{M}'$.

Support tree



Example of sphere decoding Chan and Lee

- SISO system : $N = 1$ => 2 real dimension lattice
- Signal constellation :64-QAM $x_i \in \{-7, -5, -3, -1, +1, +3, +5, +7\}$

$$\mathbf{x} = [1, -3]^T$$

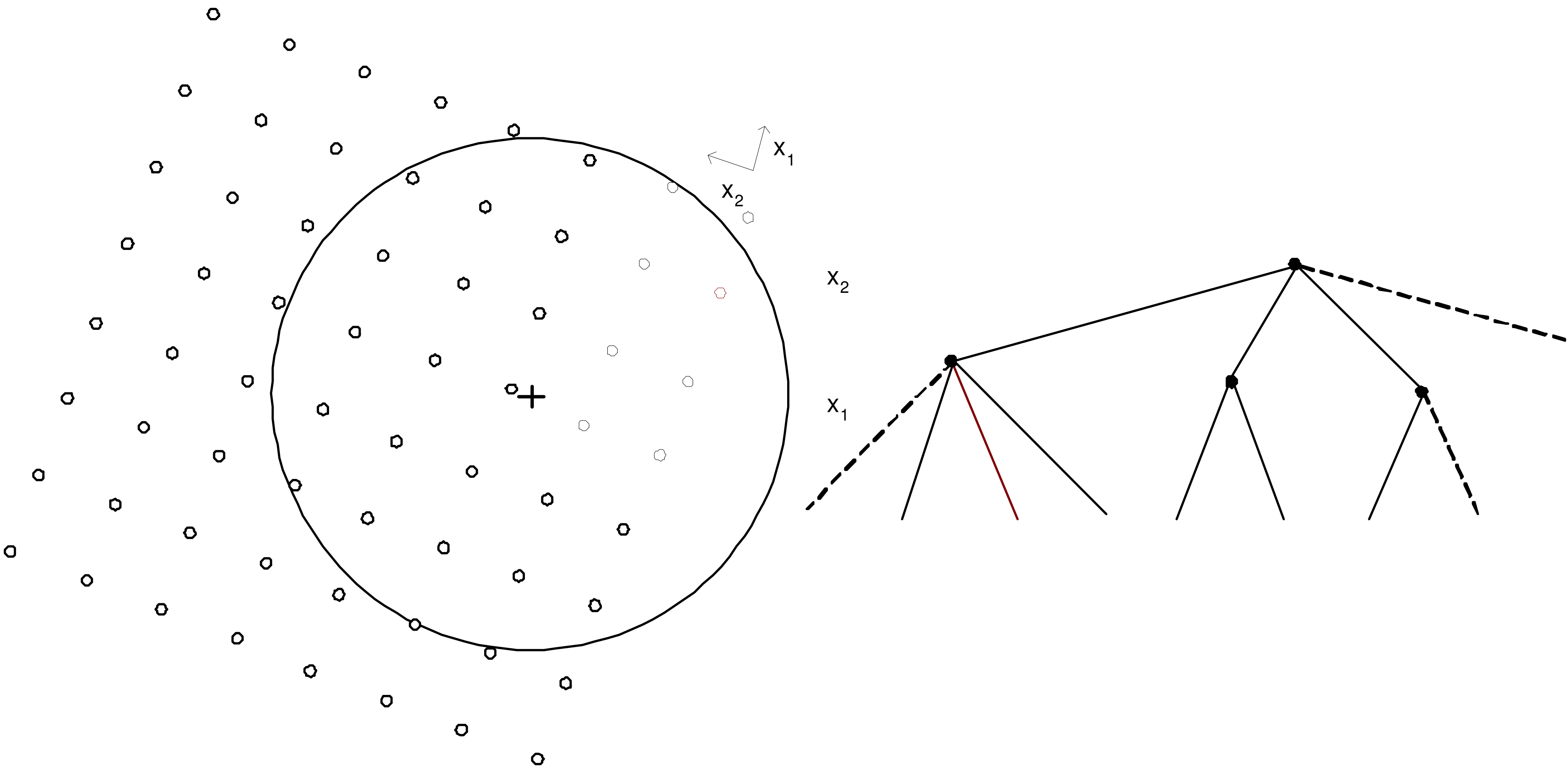
$$\mathbf{B} = \begin{pmatrix} 0.5 & -1 \\ 1 & 0.5 \end{pmatrix} \qquad \mathbf{v} = [0.58, -0.31]^T$$

$$\mathbf{y} = [4.08, -0.81]^T$$

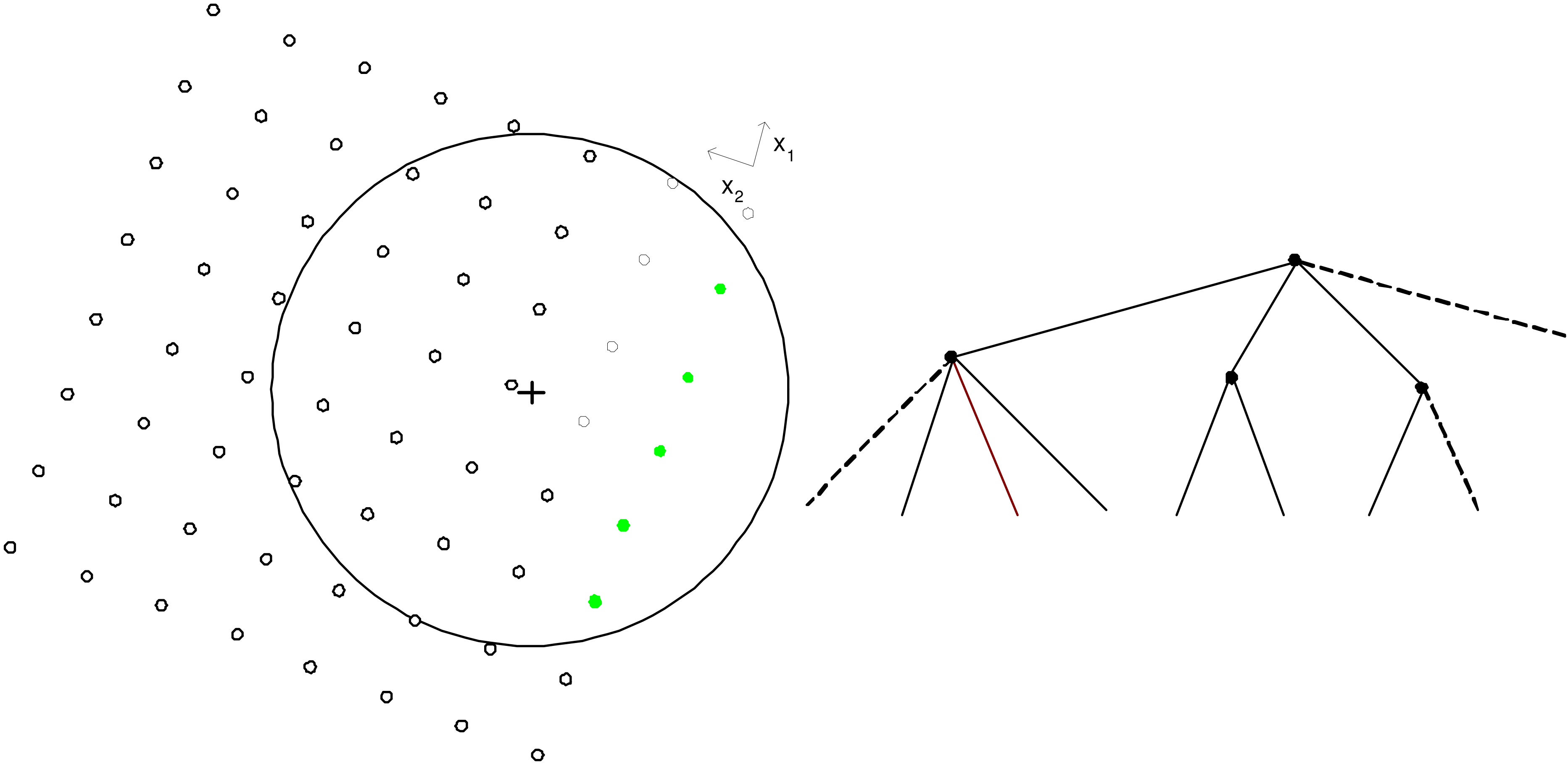
$$\tilde{\mathbf{x}} = [0.984, -3.588]^T$$

- The sphere square radius C_1 is fixed to 49 (chosen according to the noise variance)

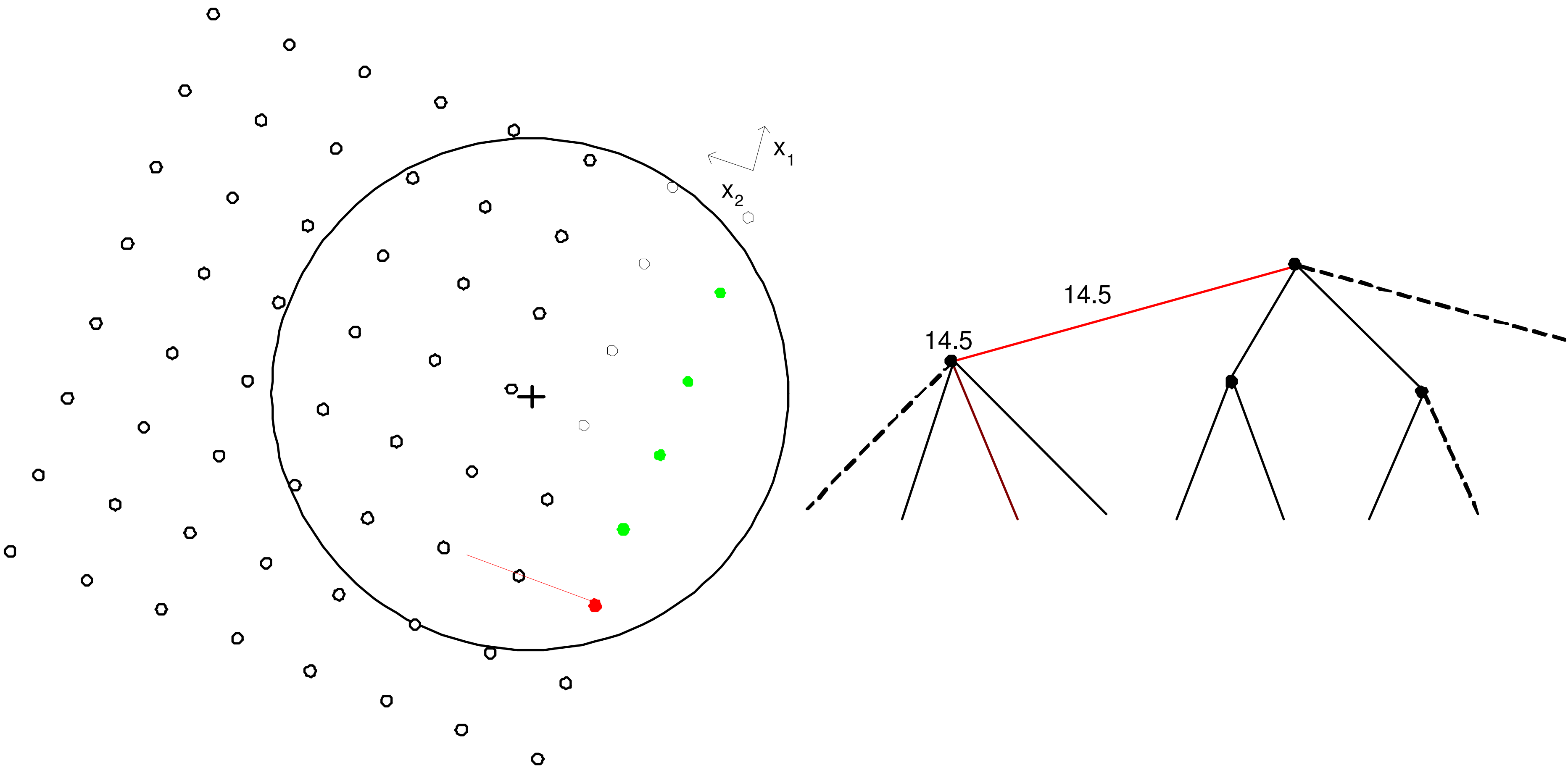
Example



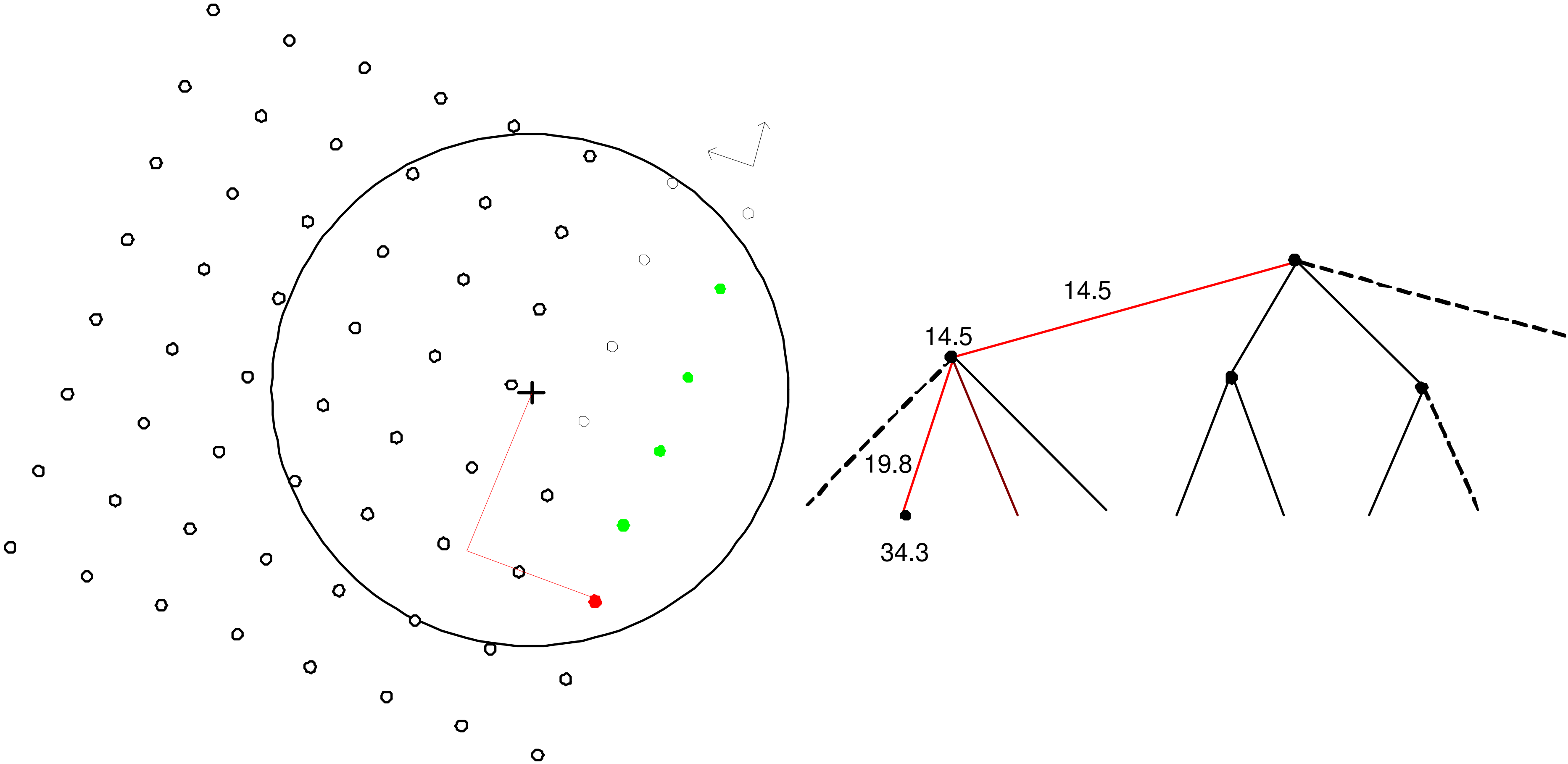
Example



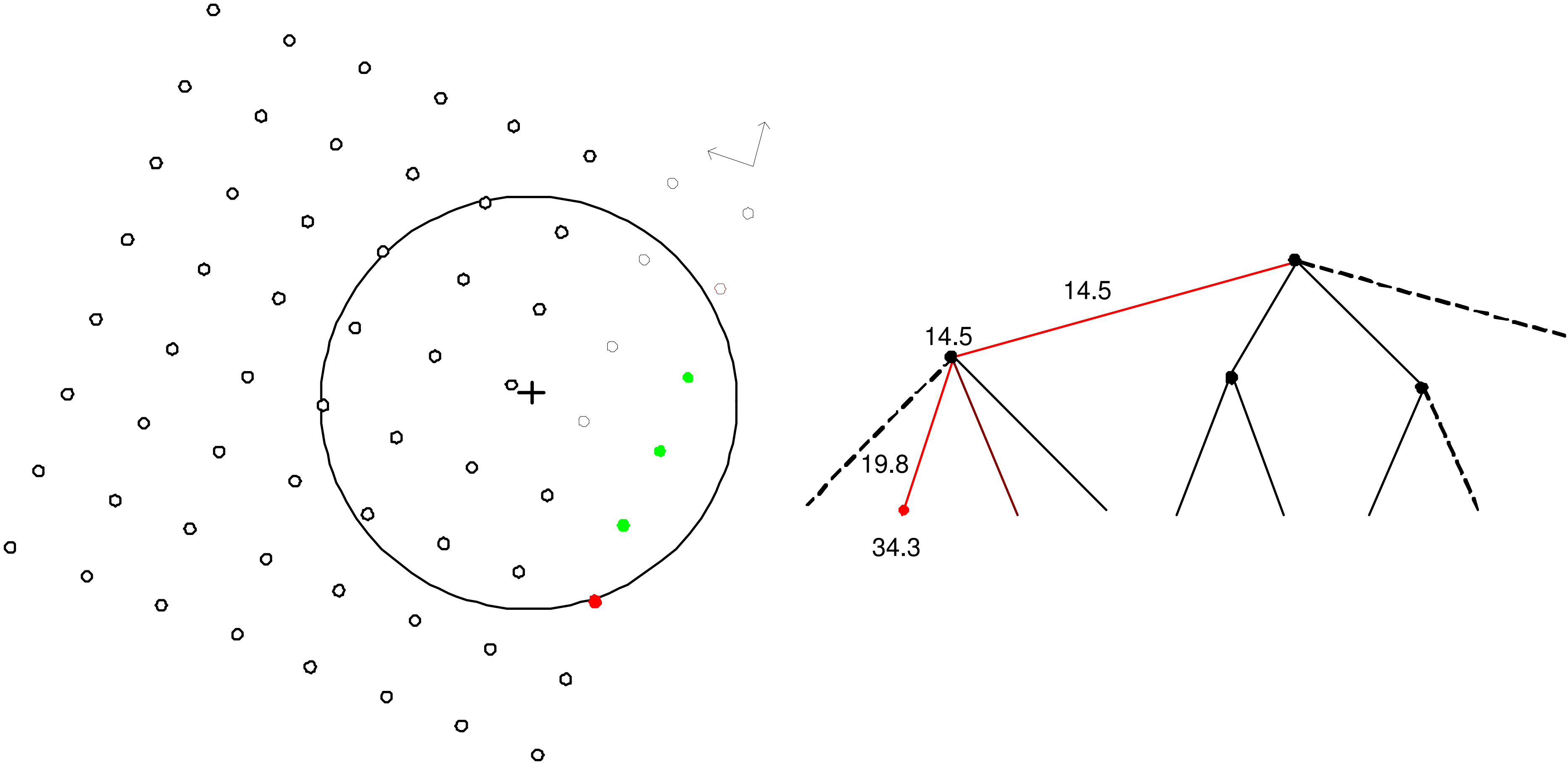
Example



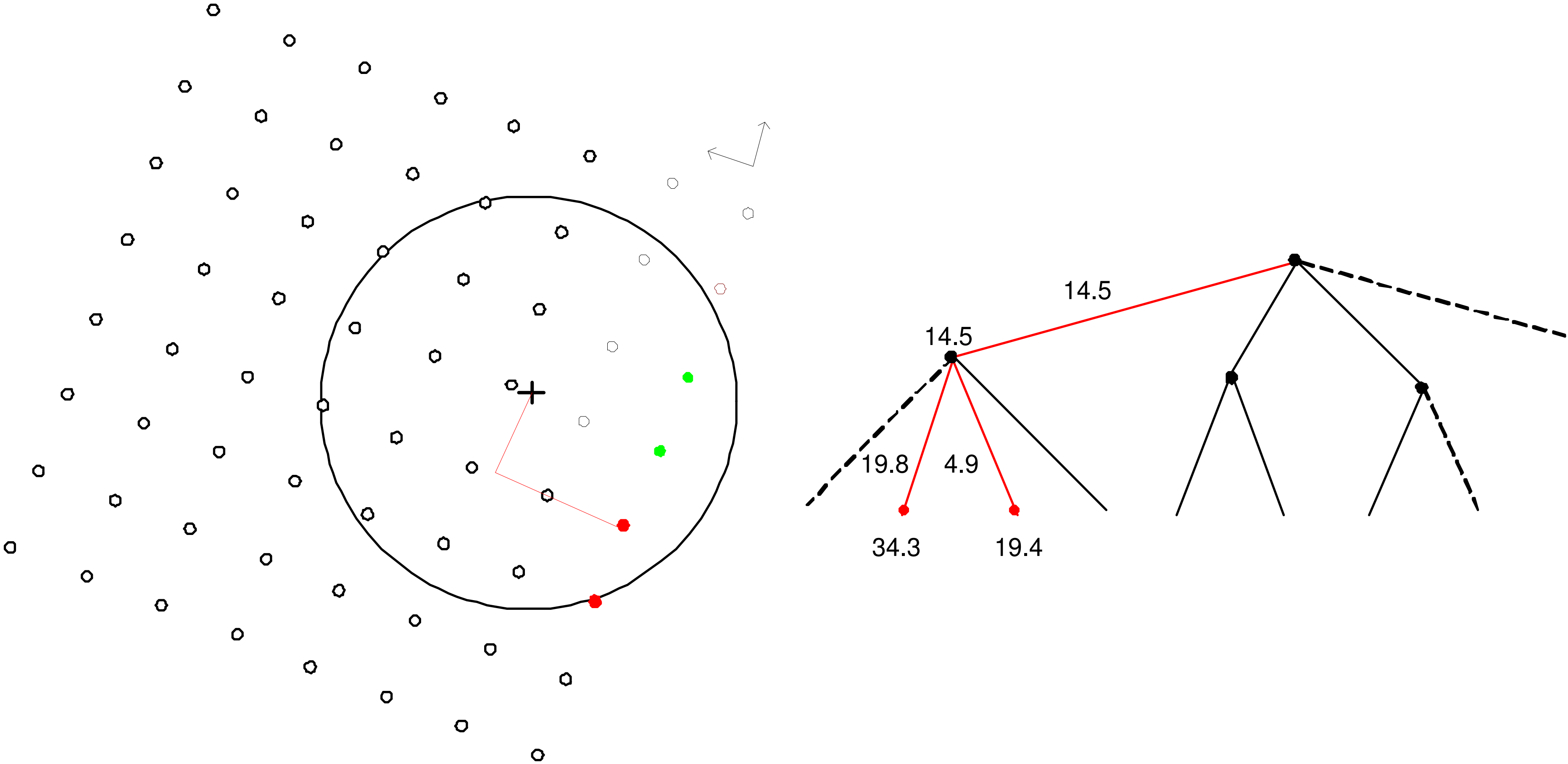
Example



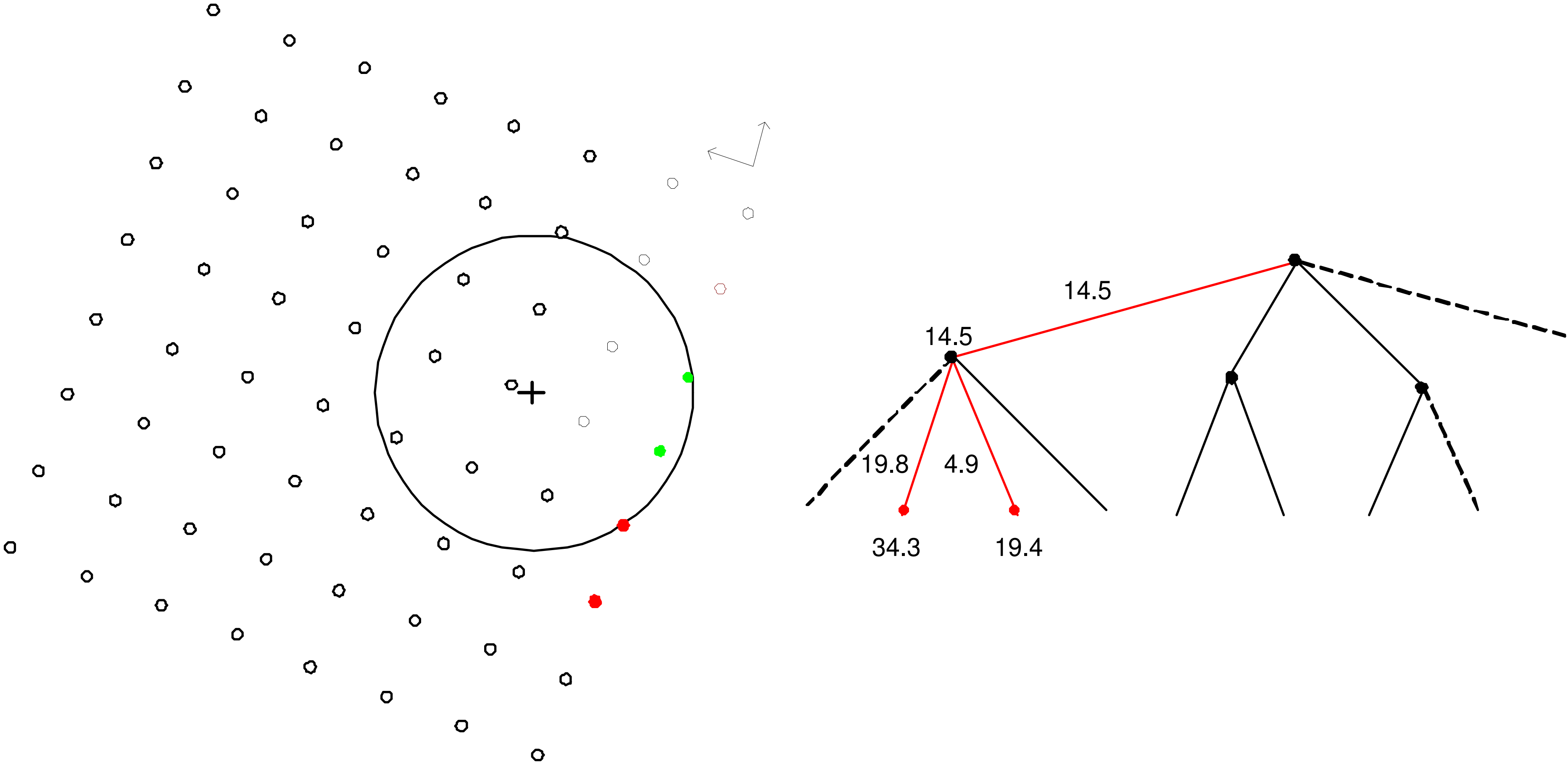
Example



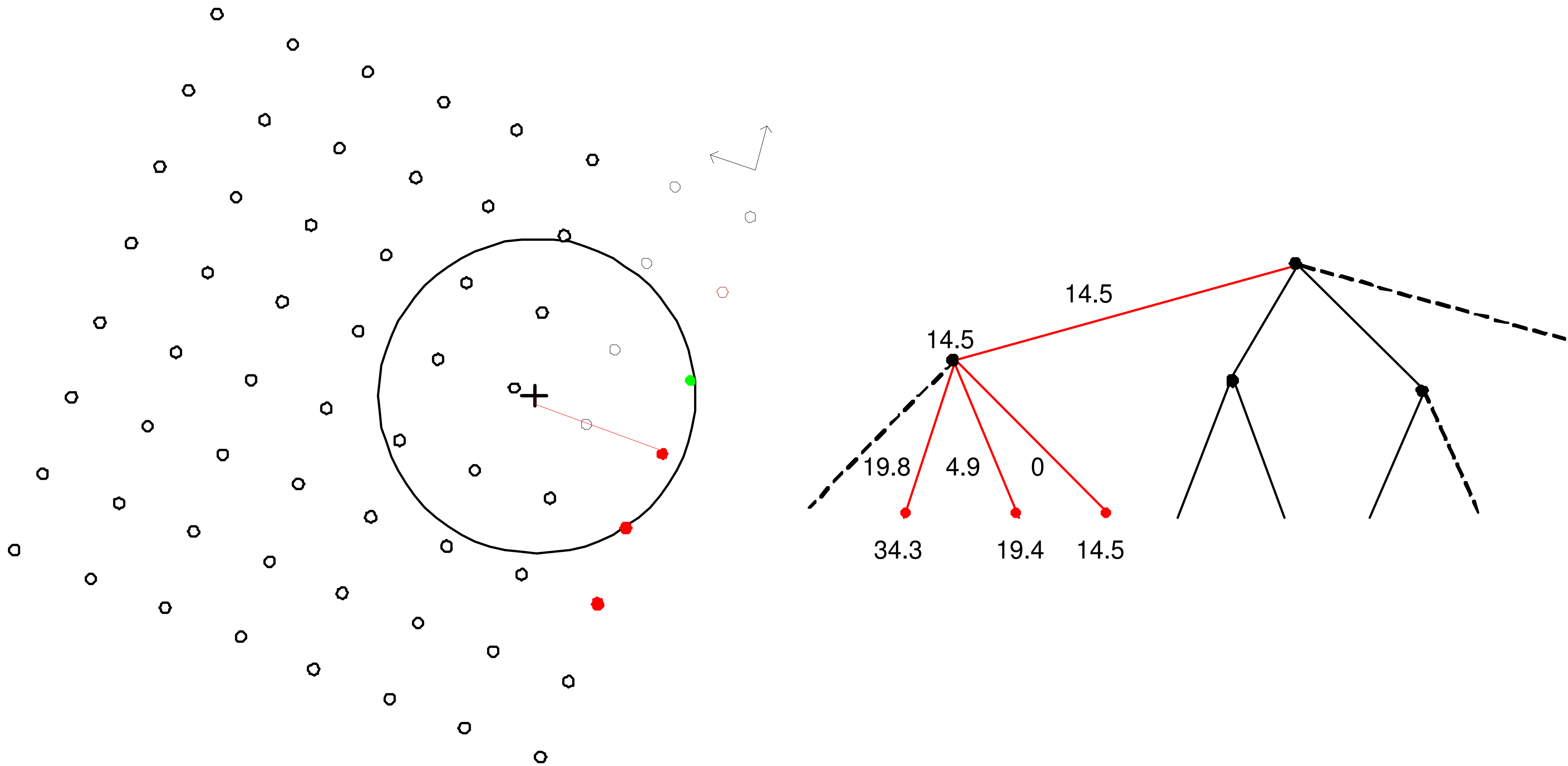
Example



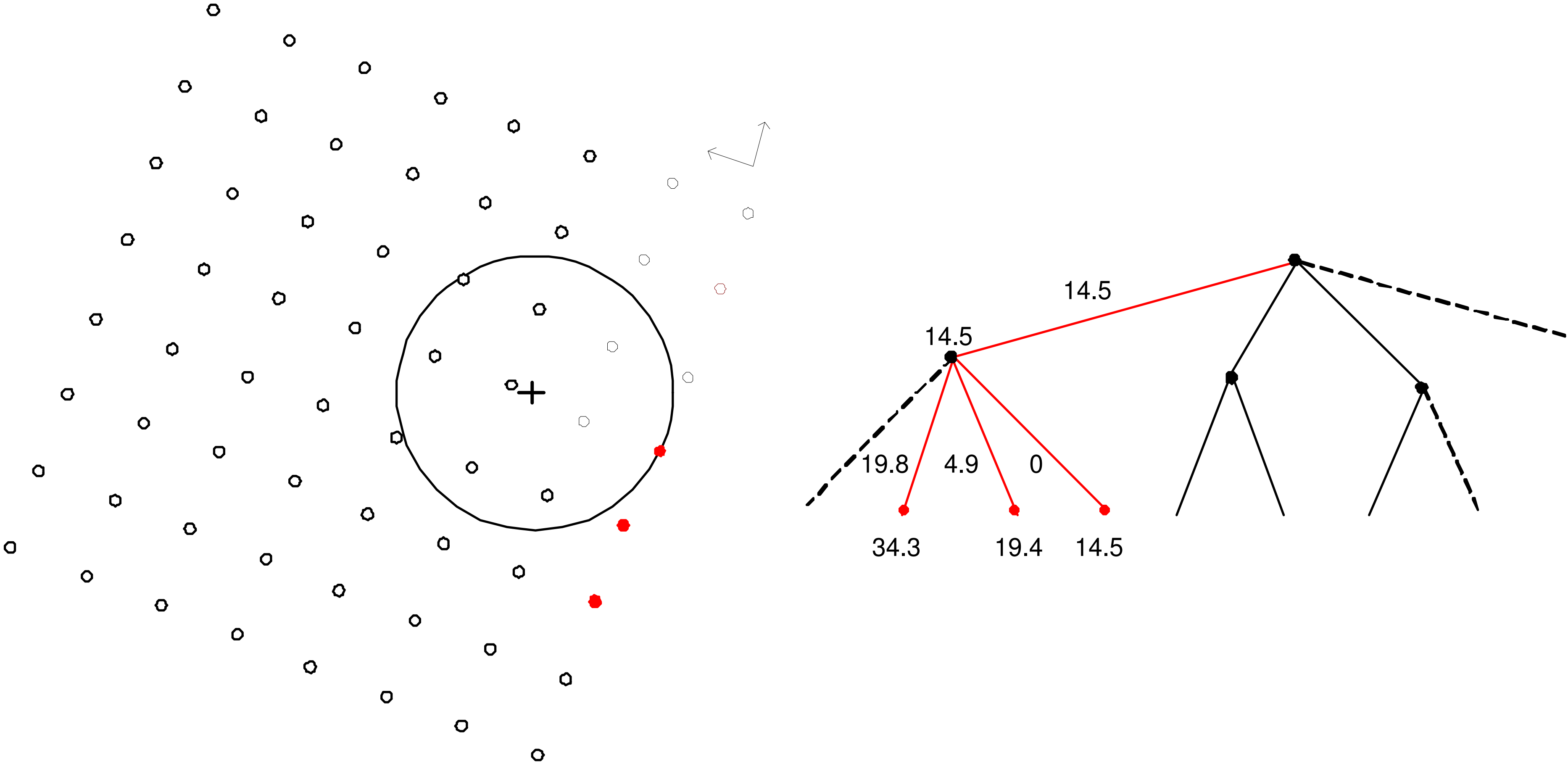
Example



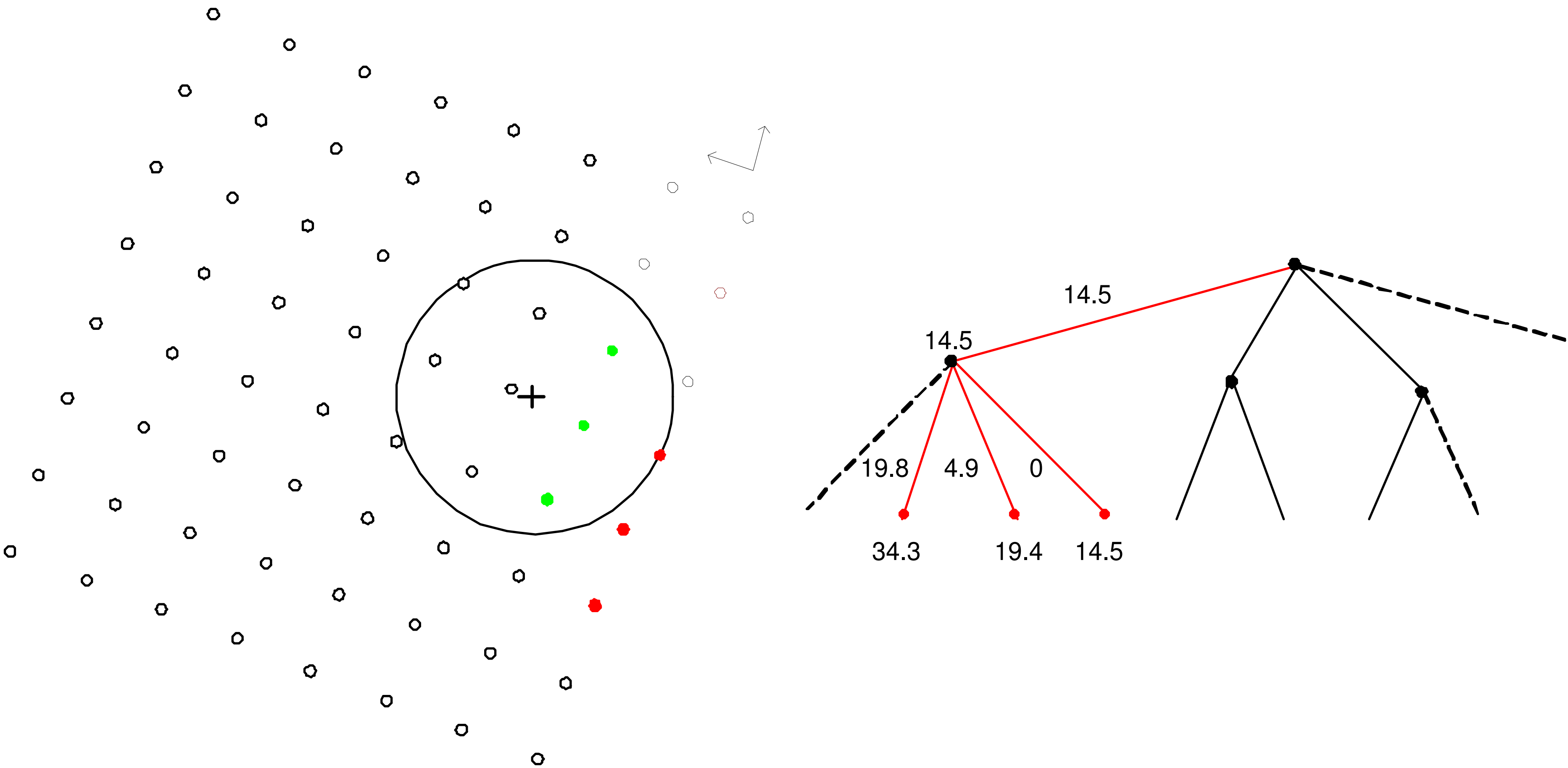
Example



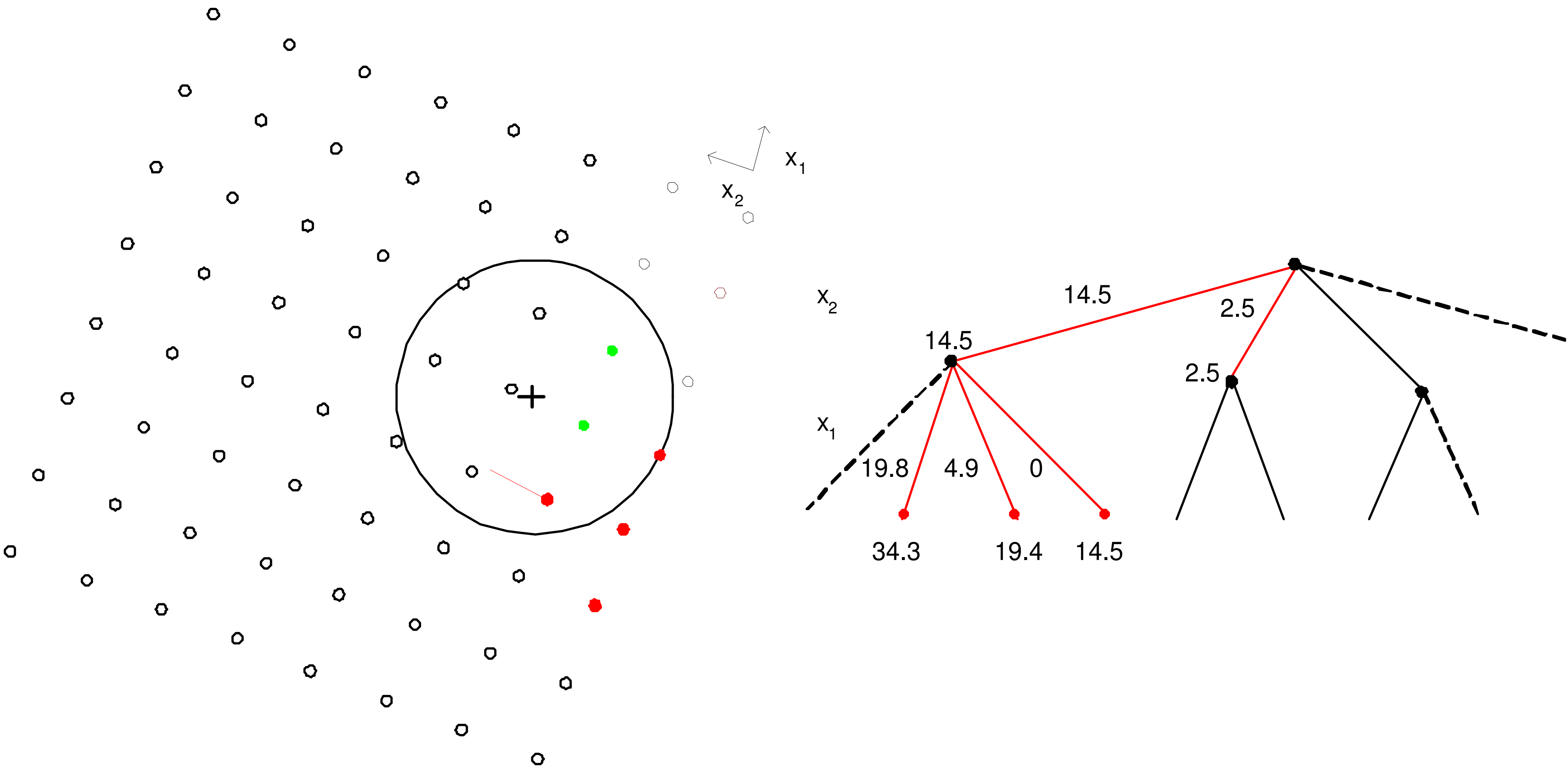
Example



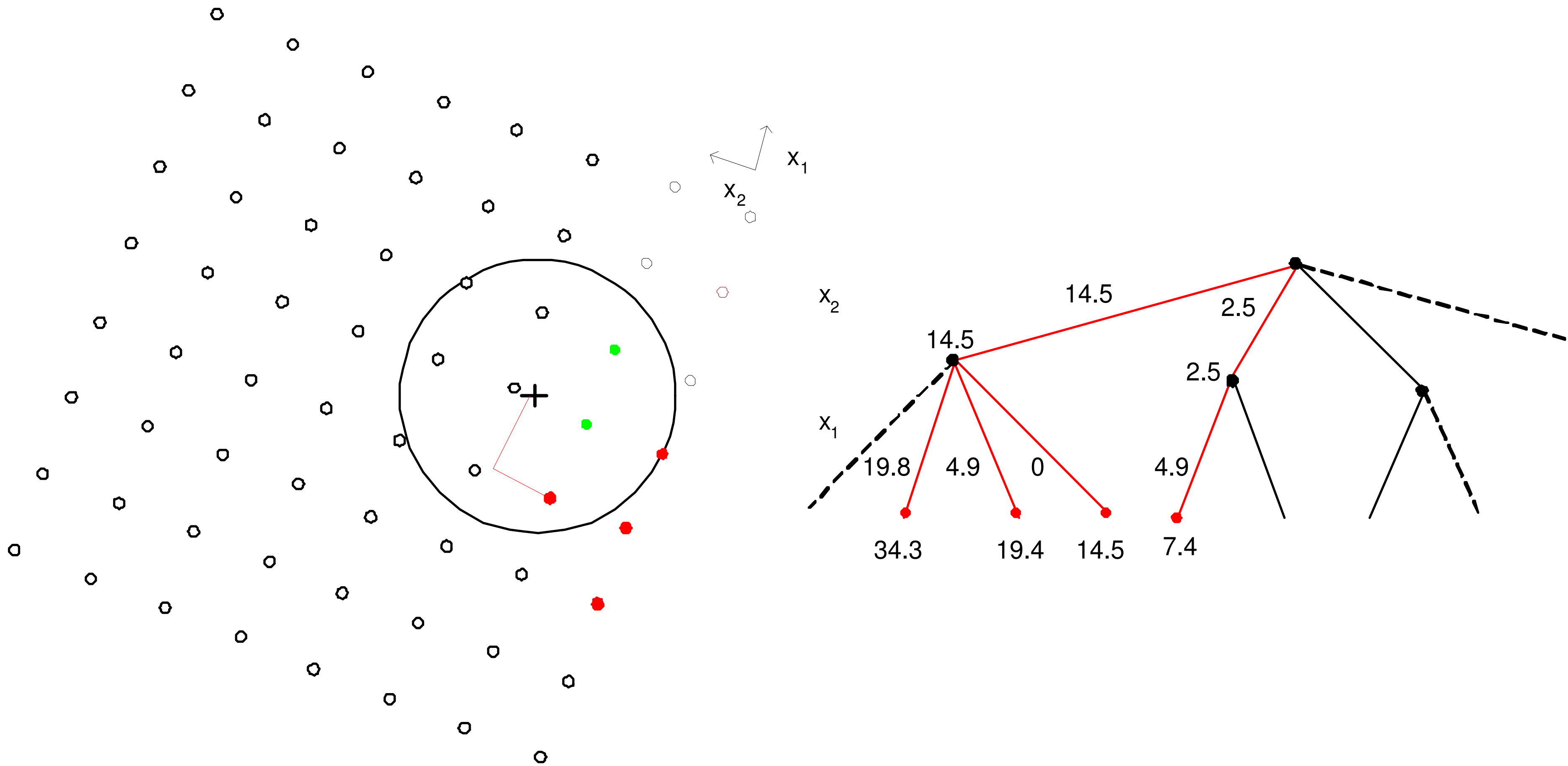
Example



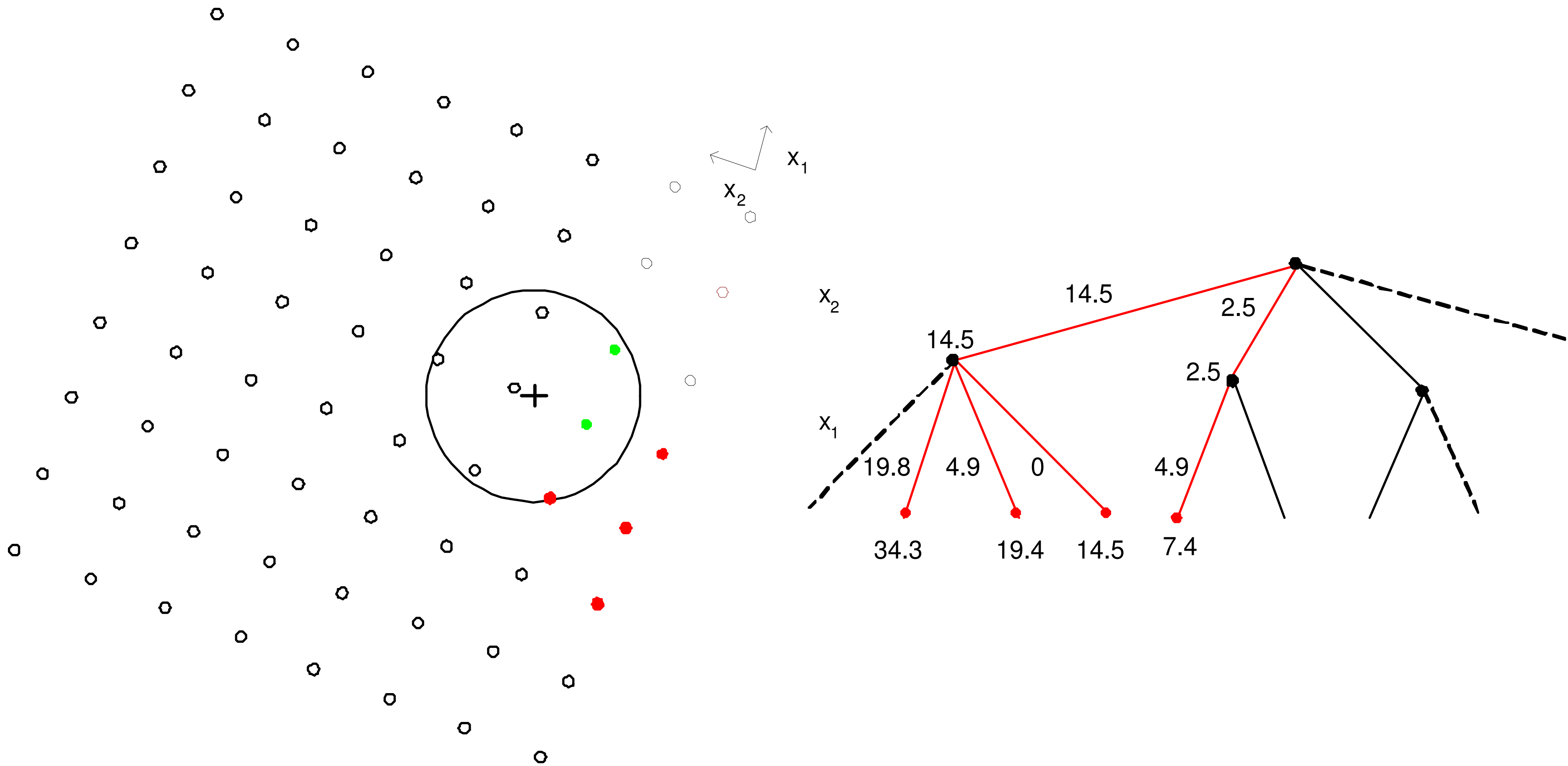
Example



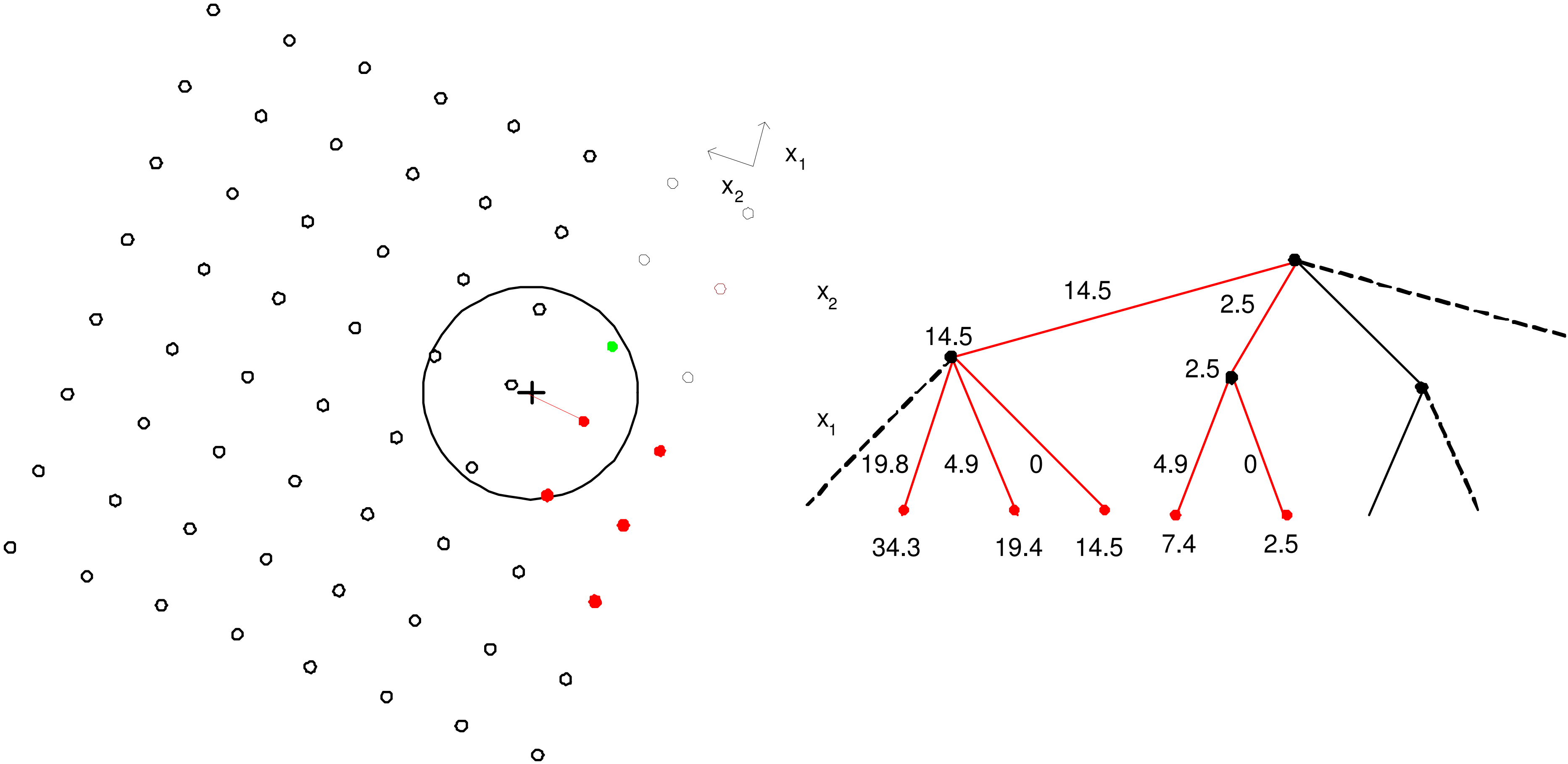
Example



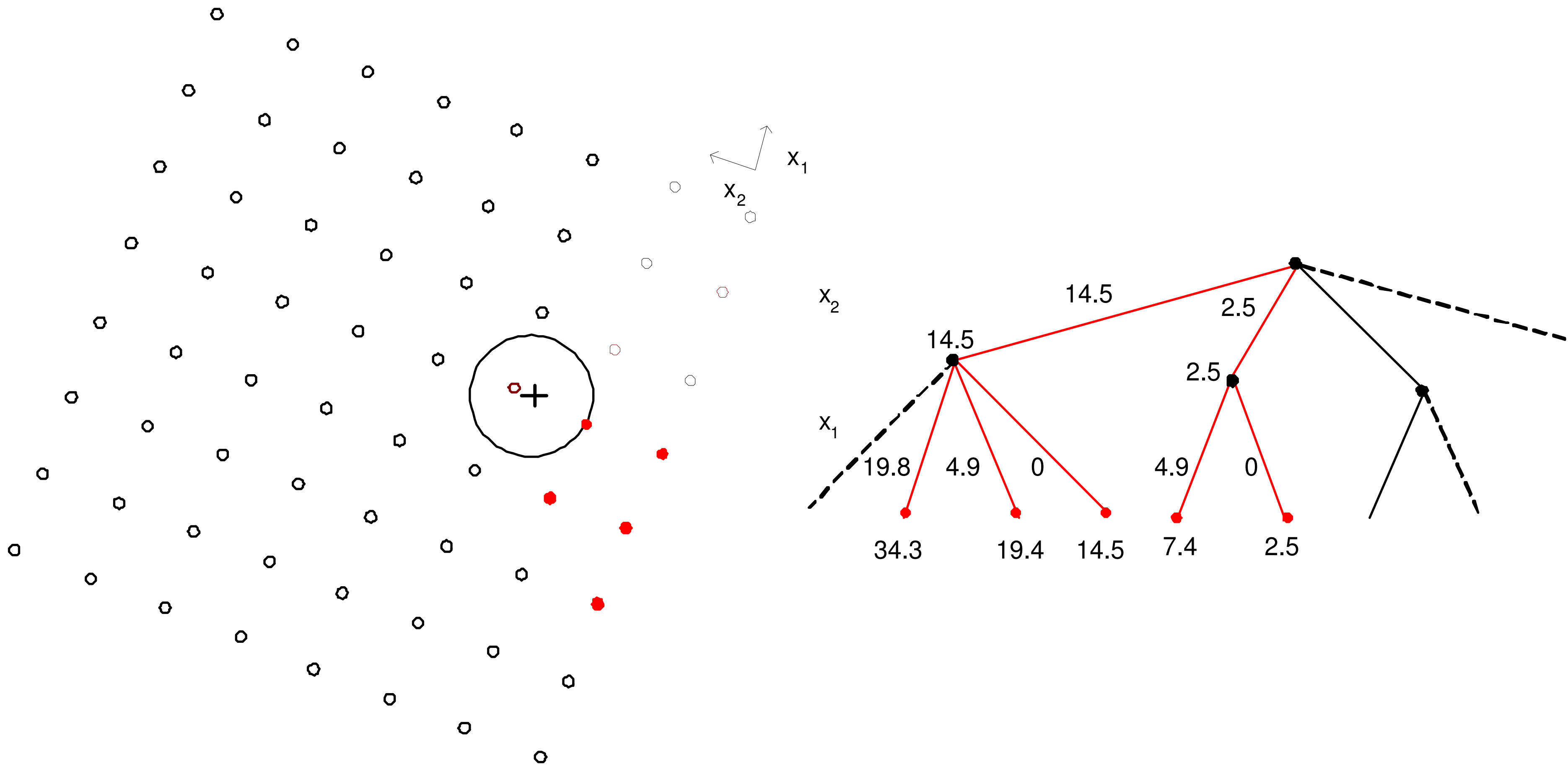
Example



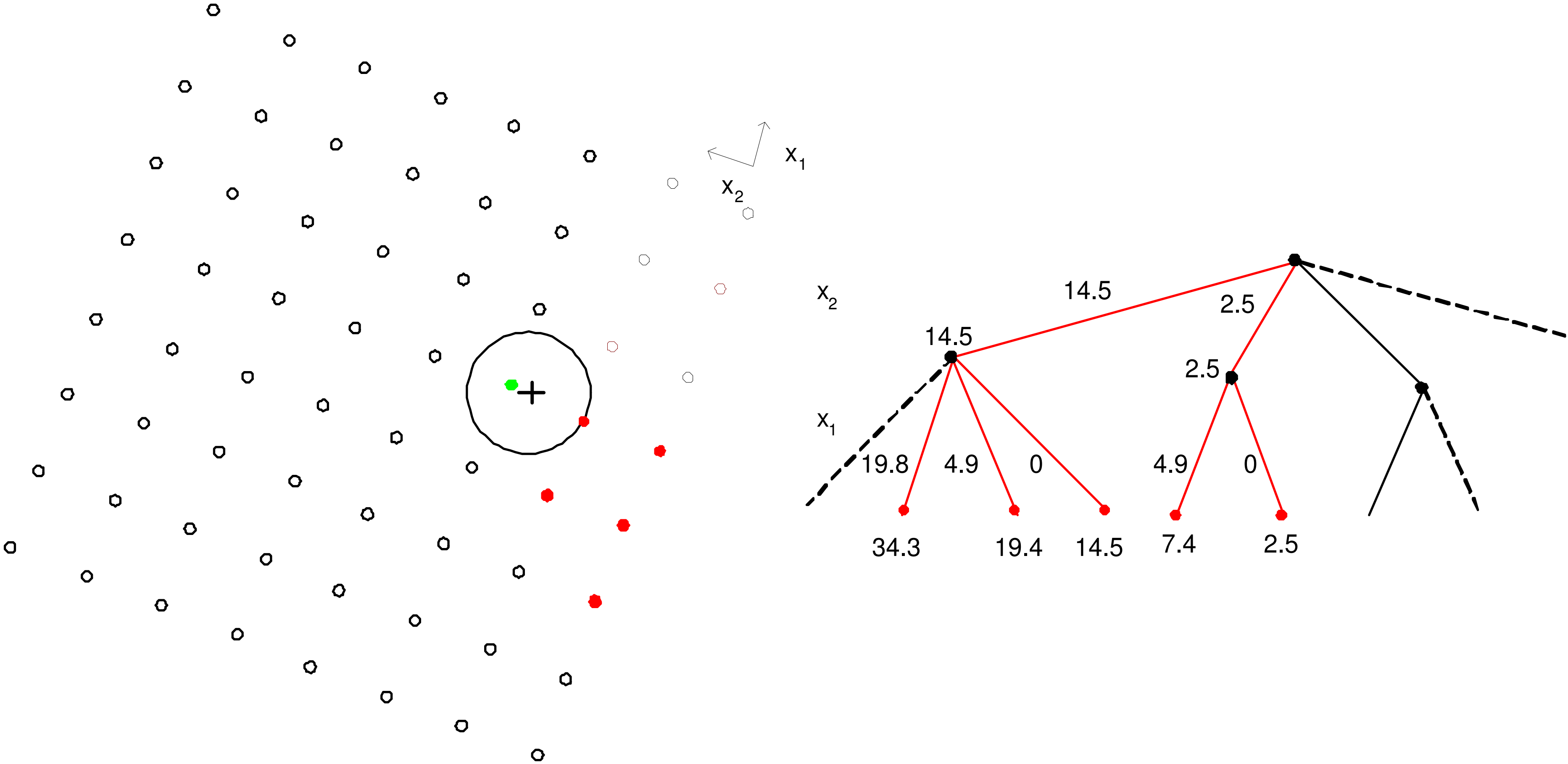
Example



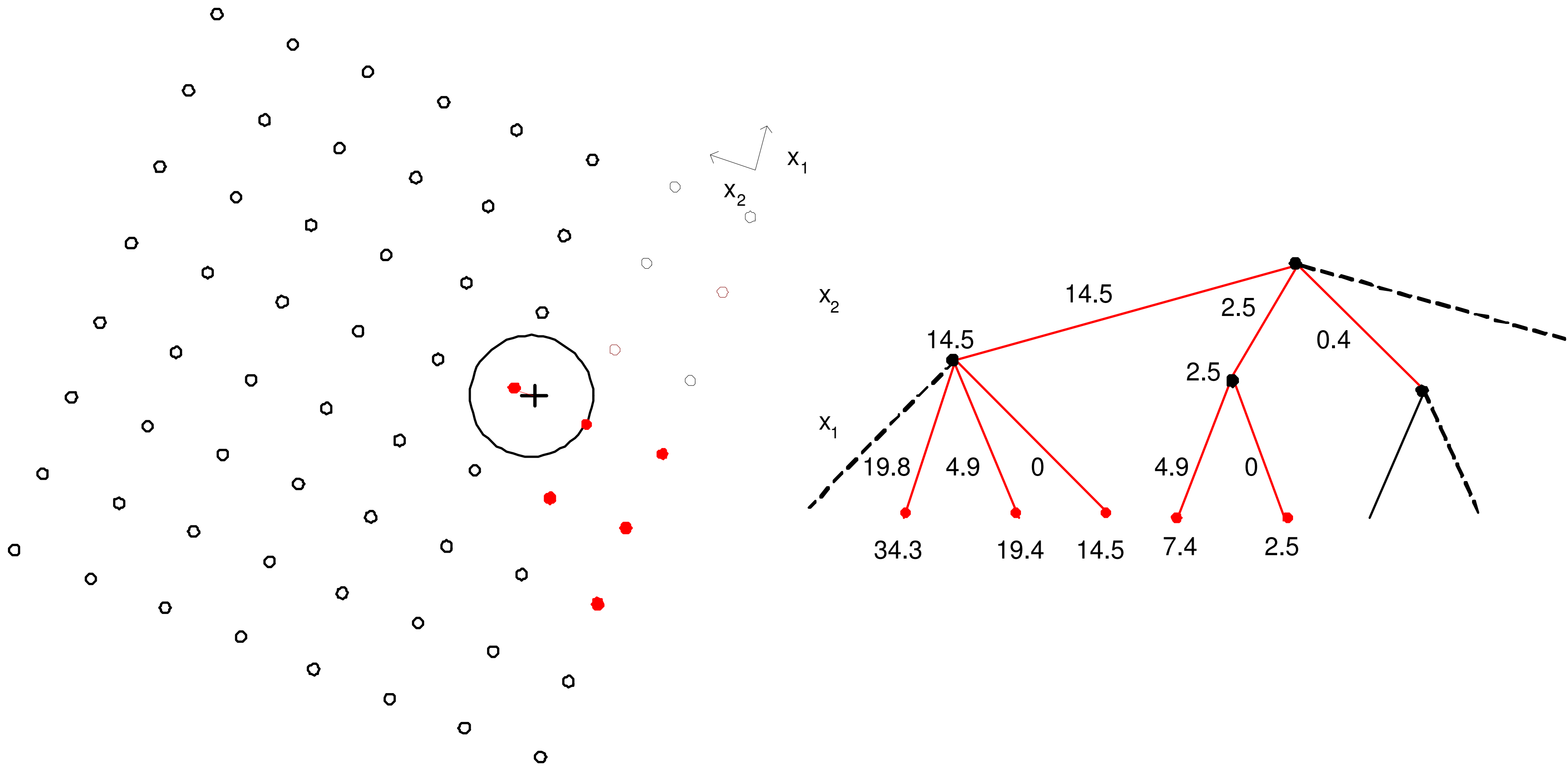
Example



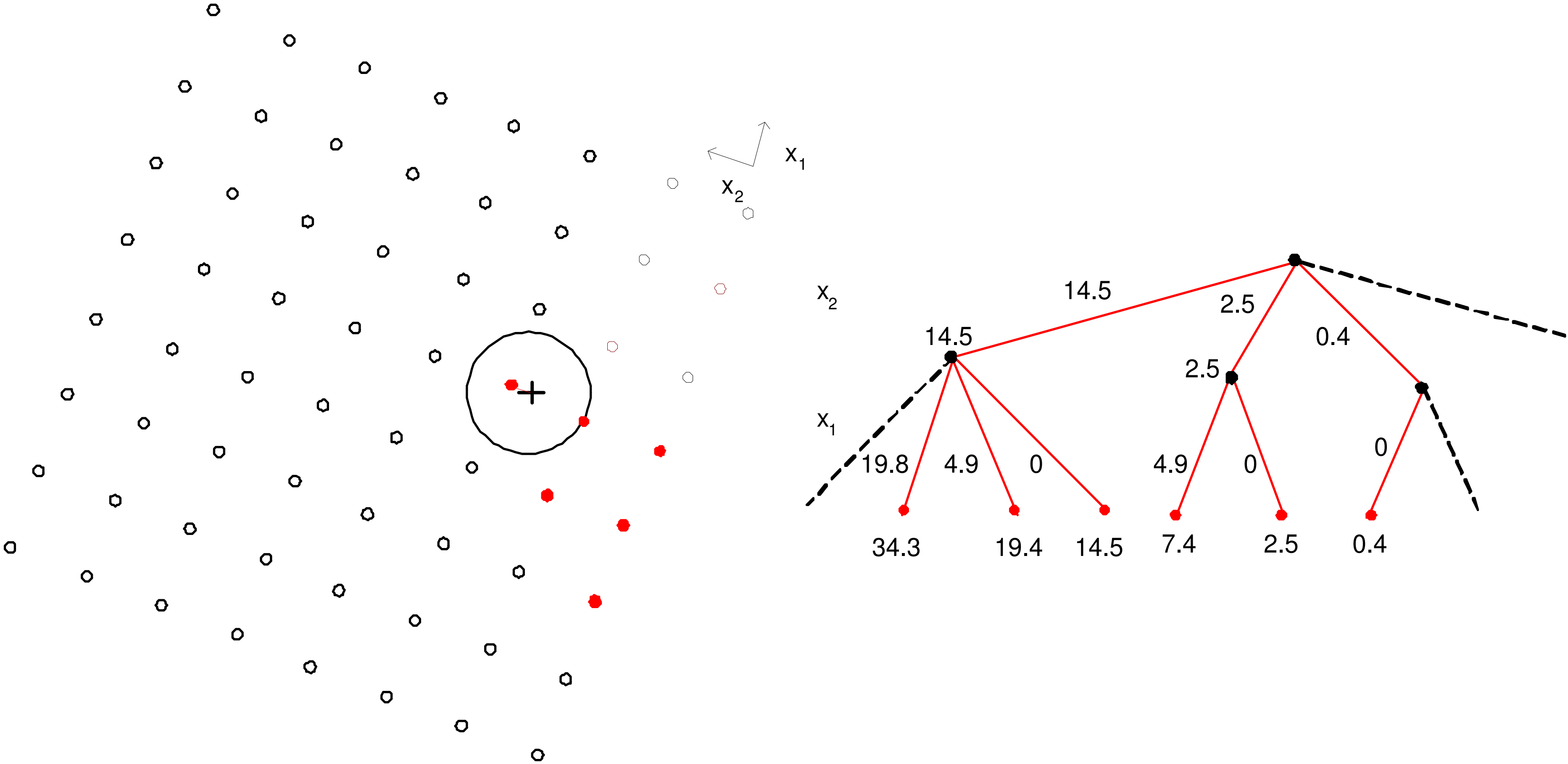
Example



Example

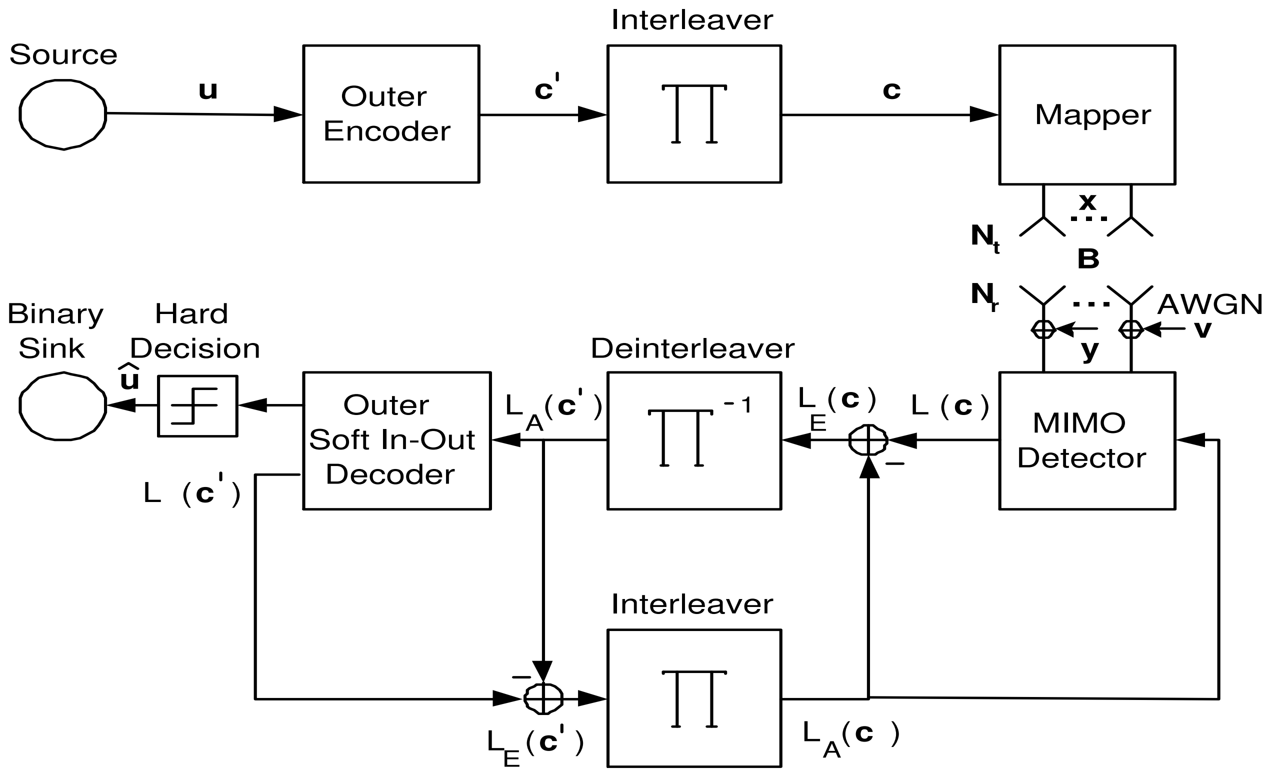


Example



$$\hat{\mathbf{x}} = [1, -3]^T$$

Turbo V-BLAST



- Scheme capable of supporting rate close to the capacity

  Hochwald and Ten Brink 03

- The outer code can be a convolutional, turbo or LDPC code

APP detector for MIMO channels

- APP detector generates a posteriori probability or log likelihood ratio about the transmitted bits c_{ij}

$$L(c_{ij}|\mathbf{y}) = \ln \frac{P(c_{ij}=+1)}{P(c_{ij}=-1)} + \ln \frac{\sum_{\mathbf{c}\in\mathbb{C}_{ij,+1}} p(\mathbf{y}|\mathbf{c})P(\mathbf{c}|c_{ij})}{\sum_{\mathbf{c}\in\mathbb{C}_{ij,-1}} p(\mathbf{y}|\mathbf{c})P(\mathbf{c}|c_{ij})}$$

- Using the Max-log approximation and assuming the independence of the bits c_{ij} we obtain :

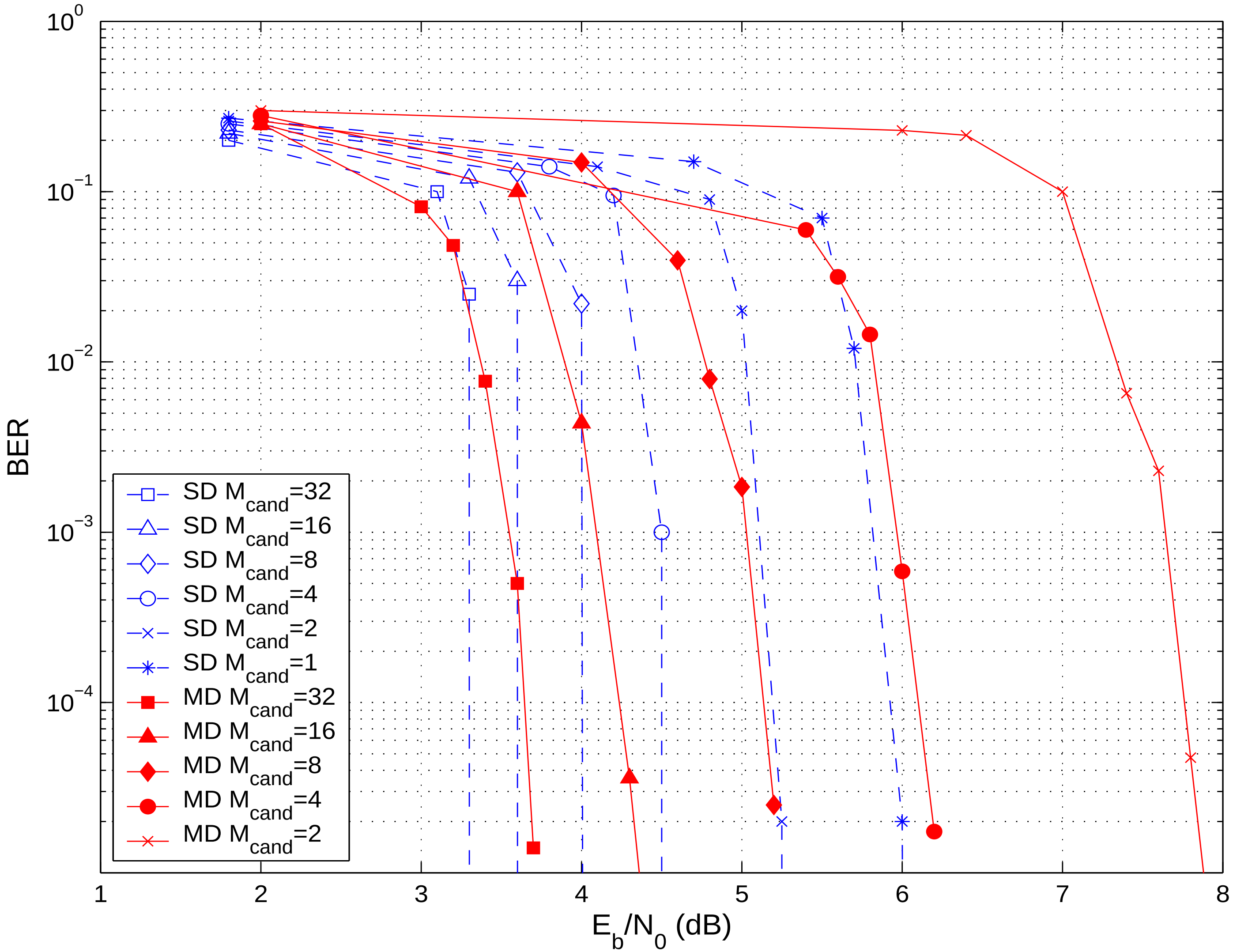
$$\begin{aligned}
L(c_{ij}|\mathbf{y}) &= \max_{\mathbf{c}\in\mathbb{C}_{ij,+1}} \left\{ -\frac{1}{2\sigma^2}||\mathbf{y}-\mathbf{B}\mathbf{x}(\mathbf{c})||^2 + \sum_{(kl)} \ln P(c_{kl}) \right\} \\
&\quad - \max_{\mathbf{c}\in\mathbb{C}_{ij,-1}} \left\{ -\frac{1}{2\sigma^2}||\mathbf{y}-\mathbf{B}\mathbf{x}(\mathbf{c})||^2 + \sum_{(kl)} \ln P(c_{kl}) \right\} \\
&= \max_{\mathbf{c}\in\mathbb{C}_{ij,+1}} \left\{ -\frac{1}{2\sigma^2}\mathcal{M}(\mathbf{x(c)}) \right\} - \max_{\mathbf{c}\in\mathbb{C}_{ij,-1}} \left\{ -\frac{1}{2\sigma^2}\mathcal{M}(\mathbf{x(c)}) \right\}
\end{aligned}$$

List sphere APP detectors

- Deep first algorithm
  - fixed diameter and selection of all the codewords Boutros, Gresset, Brunel and Fossorier 03
  - fixed diameter but search of only the M_{cand} best codewords Hochwald and Ten Brink 03. Sorting for each new candidate.
  - reduce the diameter once M_{cand} codewords have been selected
- Breadth first algorithm
  - fixed diameter and selection of all the codewords : T algorithm Le Ruyet, Bertozzi and Özbek 04
  - keep M_{cand} paths at each level : a list of M_{cand} good codewords is provided.Reid Grant and Kind 03, Le Ruyet Bertozzi and Özbek 04

Simulation results

- List Sphere Detectors versus M APP Detectors



- $N_t = N_r = 8$
- QPSK mod
- 9216 info bits
- turbo code 3GPP

R=1/2, 8 iterations

- 4 global iterations

Soft input soft output MMSE detector

- Computes an affine estimate $\hat{\mathbf{x}} = \mathbf{A}\mathbf{y} + \mathbf{b}$ such that :

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{A},\mathbf{b}} ||\mathbf{x} - \hat{\mathbf{x}}||^2$$

- Solution : $\mathbf{A} = \text{cov}(\mathbf{x}, \mathbf{y})\text{cov}(\mathbf{y}, \mathbf{y})^{-1}$ and $\mathbf{b} = E(\mathbf{x}) - \mathbf{A}\mathbf{B}E(\mathbf{x})$

$$\hat{x}_q = \frac{\sigma_x^2}{2}\mathbf{B}_q^T(\mathbf{B}\text{cov}_q(\mathbf{x}, \mathbf{x})\mathbf{B}^T + \sigma_n^2\mathbf{I}_{2N})^{-1}(\mathbf{y} - \mathbf{B}\text{E}_q(\mathbf{x}))$$

where $\mathbf{B}_q$ is the qth column of $\mathbf{B}$,

$$\text{E}_q(\mathbf{x}) = (\text{E}(x_1), \ldots, \text{E}(x_{q-1}), 0, \text{E}(x_{q+1}), \ldots, \text{E}(x_{2N}))^T$$

$$\text{cov}_q(\mathbf{x}, \mathbf{x}) = \text{diag}(\text{cov}(x_1, x_1), \ldots, \text{cov}(x_{q-1}, x_{q-1}), \frac{\sigma_x^2}{2}, \text{cov}(x_{q+1}, x_{q+1}), \ldots$$
$$\ldots, \text{cov}(x_{2N}, x_{2N}))$$

Soft input Soft output MMSE detector

We assume that the estimates $\hat{x}_q$ are the outputs of equivalent gaussian channels :

$$\hat{x}_q = \mu_q x_q + \eta_q$$

where

$$\mu_q = \frac{\sigma_x^2}{2} \mathbf{B}_q^T (\mathbf{B}\mathrm{cov}_q(\mathbf{x}, \mathbf{x})\mathbf{B}^T + \sigma_n^2 \mathbf{I}_{2N})^{-1} \mathbf{B}_q$$

$$\nu_q^2 = \frac{\sigma_x^2}{2} (\mu_q - (\mu_q)^2)$$