15 Emerging trends in fuzzy systems

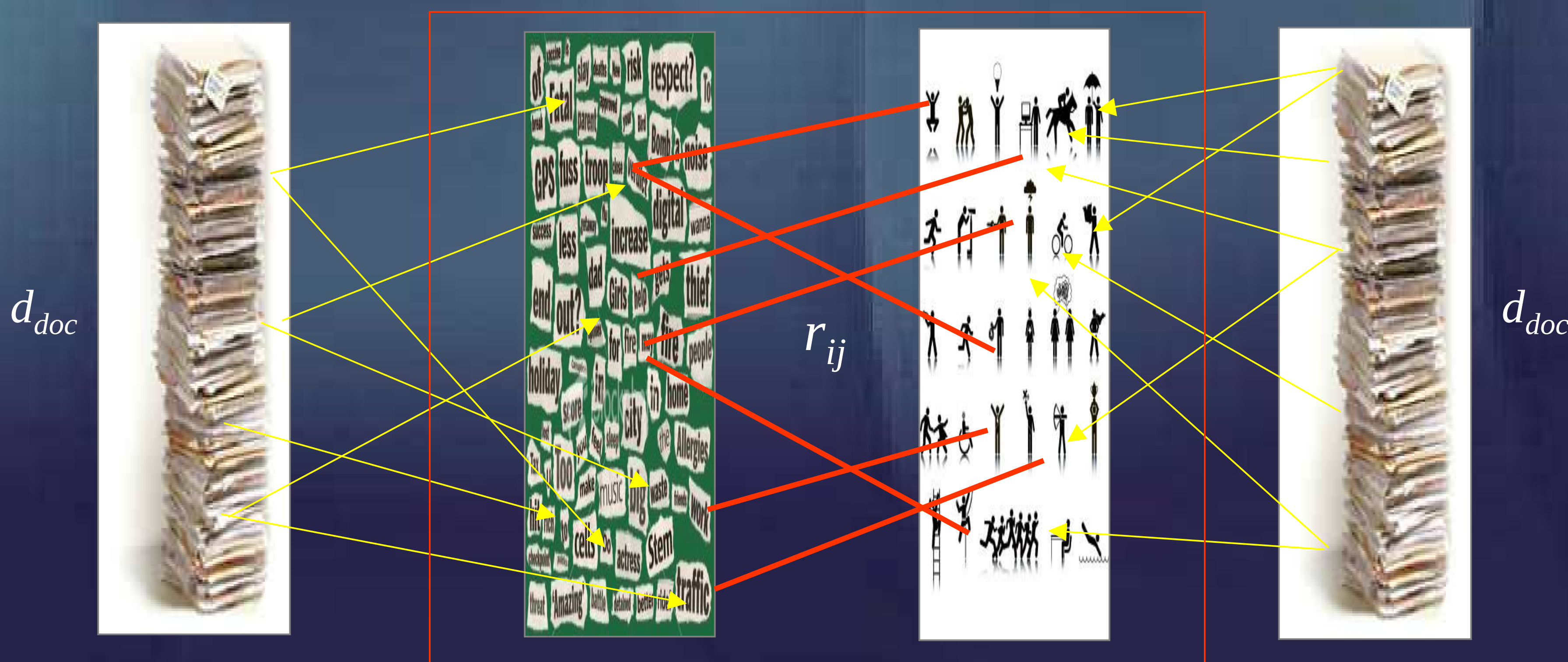
Fuzzy Systems Engineering
Toward Human-Centric Computing

Contents

15.1 Relational ontology in information retrieval

Fuzzy relational ontological model

Docs	Keywords	Concepts	Docs
$\mathbf{D}$	$\mathbf{K}$	$\mathbf{C}$	$\mathbf{D}$
$\{d_1, d_2,\ldots,d_{doc},\ldots d_u\}$	$\{k_1, k_2,\ldots,k_i,\ldots k_n\}$	$\{c_1, c_2,\ldots,c_j,\ldots c_m\}$	$\{d_1, d_2,\ldots,d_{doc},\ldots d_u\}$



$$R = \{(k_i, c_j) \mid k_i \in \mathbf{K},\ c_j \in \mathbf{C}\}$$

Pedrycz and Gomide, FSE 2007

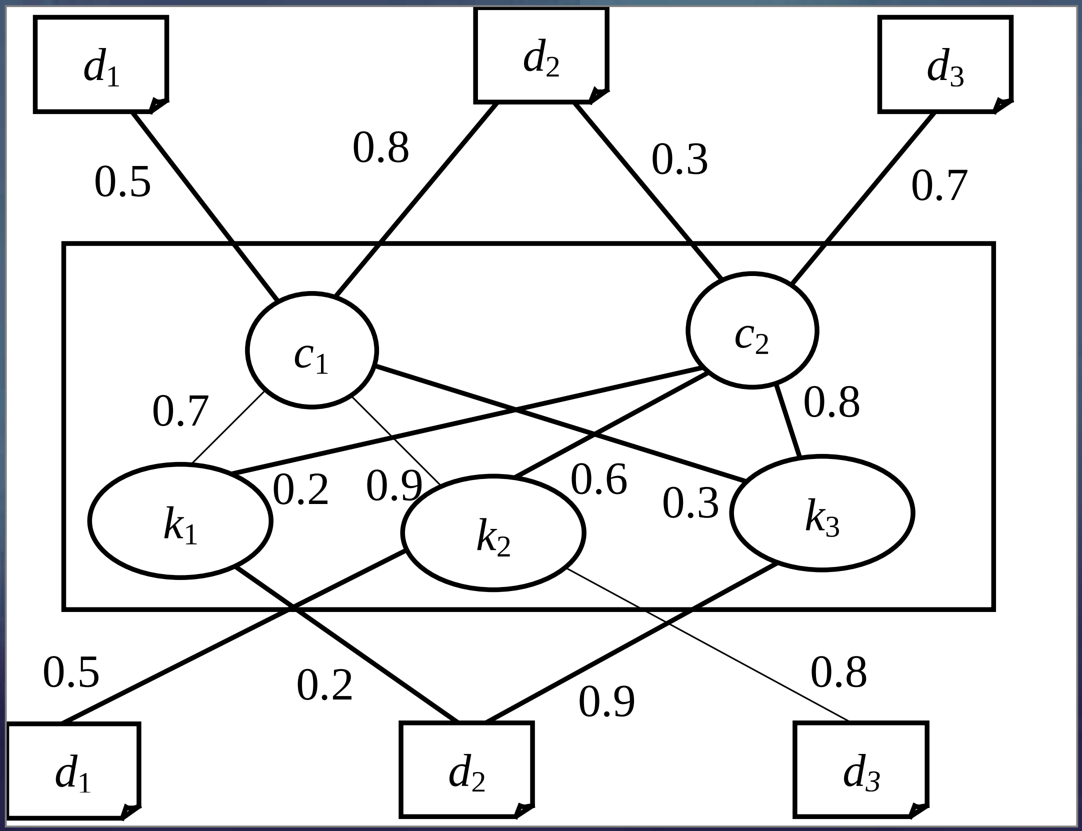
Relational ontology model

- Concerns with
  - two-layer ontology
    - category names
    - keywords
  - association between category and keywords (fuzzy relation)
- Fuzzy relational ontology
  - fuzzy relation $R = [r_{ij}]$ in $\mathbf{K}\times\mathbf{C}$
  - $\mathbf{K} = \{k_1, k_2, \ldots., k_n\}$
  - $\mathbf{C} = \{c_1, c_2, \ldots., c_m\}$
  - r_{ij} degree of association between k_i and c_j

$$R = \begin{array}{c} \\ k_1 \\ k_2 \\ \vdots \\ k_n \end{array} \begin{array}{c} \begin{array}{cccc} c1 & c2 & \cdots & cm \end{array} \\ \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1m} \\ r_{21} & r_{22} & \cdots & r_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nm} \end{bmatrix} \end{array}$$

Pedrycz and Gomide, FSE 2007

Example



d_1
d_2
d_3
0.5
0.8
0.3
0.7
c_1
c_2
0.7
0.8
0.2
0.9
0.6
0.3
k_1
k_2
k_3
0.5
0.2
0.9
0.8
d_1
d_2
d_3

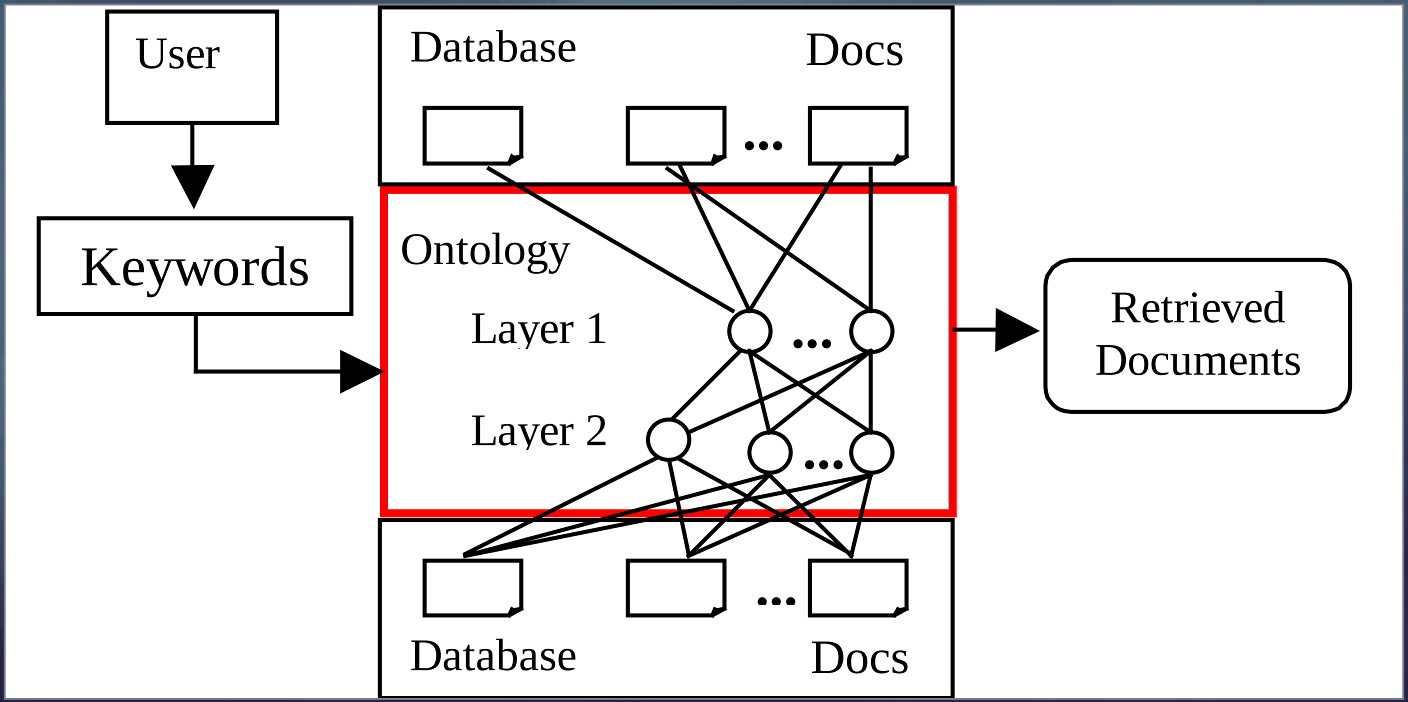
Information retrieval model

- IR = $< \mathbf{D}, \mathbf{Q}, V, F(q_i, d_{doc})>$

  D : set of document representation
  Q : set of query representation
  V : framework to represent docs, queries and their relations
  F : function associating a real number to (*doc*, *query*) pair

- $\mathbf{D} = \{d_1, d_2, \ldots, d_{\mathrm{doc}}, \ldots, d_{\mathrm{u}}\}$

Information retrieval system structure



Documents representation

$$T_k = \begin{array}{c c} & \begin{array}{cccc} k_1 & k_2 & \cdots & k_n \end{array} \\ \begin{array}{c} d_1 \\ d_2 \\ \vdots \\ d_u \end{array} & \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1n} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{u1} & \alpha_{u2} & \cdots & \alpha_{un} \end{bmatrix} \end{array}$$

$$T_c = \begin{array}{c c} & \begin{array}{cccc} c_1 & c_2 & \cdots & c_m \end{array} \\ \begin{array}{c} d_1 \\ d_2 \\ \vdots \\ d_u \end{array} & \begin{bmatrix} \beta_{11} & \beta_{12} & \cdots & \beta_{1m} \\ \beta_{21} & \beta_{22} & \cdots & \beta_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{u1} & \beta_{u2} & \cdots & \beta_{um} \end{bmatrix} \end{array}$$

Query representation

- $Q = \{k_i, c_j\}$; k_i, c_j may be linked by *and* or *or* operators

  Q is represented by vectors $\mathbf{x} \in \{0,1\}^n$ and $\mathbf{y} \in \{0,1\}^m$

$$x_i = \begin{cases} 1 & \text{if} \quad k_i \in Q \\ 0 & \text{otherwise} \end{cases} \qquad y_j = \begin{cases} 1 & \text{if} \quad c_j \in Q \\ 0 & \text{otherwise} \end{cases}$$

Information retrieval

- $\mathbf{x}$ vector query keywords
- $\mathbf{y}$ vector query keywords
- R fuzzy relational ontology

$$G_c = \mathbf{x} \circ R$$

$$G_k = R \circ \mathbf{y}$$

- $Q = (k_1 \vee k_2 \vee c_1) \wedge (k_3 \vee c_2)$
- $\mathbf{C} = \{c_1, c_2\}$, $\mathbf{K} = \{k_1, k_2, k_3\}$
- $F_c = [f_{c1}, \ldots, f_{cj}, \ldots, f_{cm}]$
- $F_k = [f_{k1}, \ldots, f_{kj}, \ldots, f_{kn}]$
- z_1 : threshold (chosen by design)

$$f_{cj} = \begin{cases} g_{cj} & \text{if} \quad g_{cj} > z_1 \\ 0 & \text{otherwise} \end{cases}$$

$$f_{ki} = \begin{cases} g_{ki} & \text{if} \quad g_{ki} > z_1 \\ 0 & \text{otherwise} \end{cases}$$

- Relevance degrees between docs and keywords

$$V_{DK} = T_k \circ F_k$$

- Relevance degrees between docs and concepts

$$V_{DC} = T_c \circ F_c^{\ T}$$

- Retrieval vector

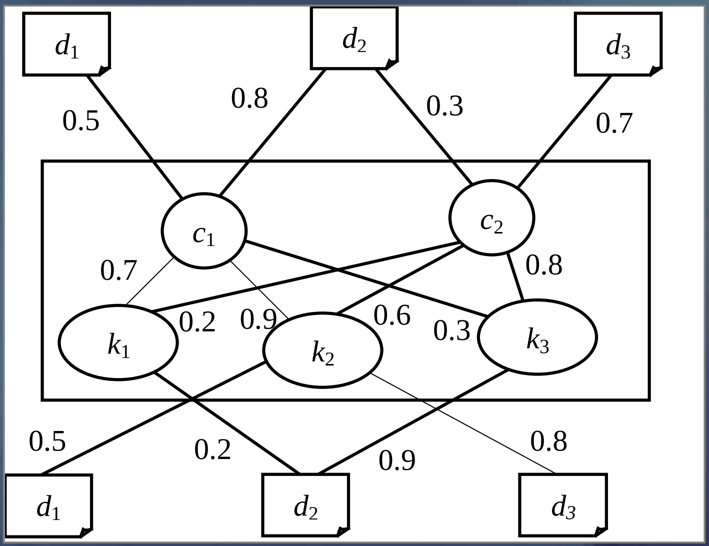
$$V_D = V_{DK} \cup V_{DC}$$

- Ordering V_D produces retrieval vector V

procedure INFORMATION-RETRIEVAL (Q) **returns** documents
input: query Q
local: thresholds: z_1, z_2
fuzzy relations: G_c, G_k

set $Q = \{k_i, c_j\}$
split Q: $Q_1 = \{k_i\}$ and $Q_2 = \{c_j\}$
construct queries vectors **x** and **y**
compute $G_c = [g_{cj}]$
compute $G_k = [g_{ki}]$
select categories with $g_{cj} > z_1$
select keywords with $g_{ki} > z_1$
find database documents related with the categories c_j
find database documents related with the keywords k_i
find database documents related with the categories presented in Q_2
find database documents related with the keywords presented in Q_1
if sub-queries AND connected **then** select the common documents
if sub-queries OR connected **then** select all documents
compute V_{DK}
compute V_{DC}
set $V_D = V_{DK} \cup V_{DC}$
set V = rank ordered V_D according to increasing values
retrieval documents for which V component values are greater than or equal to z_2
return documents

Example



$$R = \begin{bmatrix} 0.7 & 0.2 \\ 0.9 & 0.6 \\ 0.3 & 0.8 \end{bmatrix}$$

$$T_k = \begin{bmatrix} 0 & 0.5 & 0 \\ 0.2 & 0 & 0.9 \\ 0 & 0.8 & 0 \end{bmatrix}$$

$$T_c = \begin{bmatrix} 0.5 & 0 \\ 0.8 & 0.3 \\ 0 & 0.7 \end{bmatrix}$$

$Q = \{k_2 \ and \ c_1\}$

$z_1 = 0.65$

$z_2 = 0.4$

$Q_1 = \{k_3\}$ and $Q_2 = \{c_1\}$

$$Gc = x \circ R = [0 \quad 0 \quad 1] \circ \begin{bmatrix} 0.7 & 0.2 \\ 0.9 & 0.6 \\ 0.3 & 0.8 \end{bmatrix} = [0.3 \quad 0.8]$$

$$Gk = R \circ y = \begin{bmatrix} 0.7 & 0.2 \\ 0.9 & 0.6 \\ 0.3 & 0.8 \end{bmatrix} \circ \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.7 \\ 0.9 \\ 0.3 \end{bmatrix}$$

$$V_{DC} = T_C \circ F'_C = \begin{bmatrix} 0.5 & 0 \\ 0.8 & 0.3 \\ 0 & 0.7 \end{bmatrix} \circ \begin{bmatrix} 0 \\ 0.8 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.3 \\ 0.8 \end{bmatrix}$$

$$V_{DK} = T_K \circ F_K = \begin{bmatrix} 0 & 0.5 & 0 \\ 0.2 & 0 & 0.9 \\ 0 & 0.8 & 0 \end{bmatrix} \circ \begin{bmatrix} 0.7 \\ 0.9 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0.2 \\ 0.8 \end{bmatrix}$$

$$V_D = V_{DK} \cup V_{DC} = \begin{bmatrix} 0.5 \\ 0.3 \\ 0.8 \end{bmatrix}$$

$$\boxed{\mathrm{V} = \begin{bmatrix} 0.8 \\ 0.5 \\ 0.3 \end{bmatrix}}$$

Example

- 100 scientific papers on computational intelligence
- 61 words
- 6 concepts
- 55 keywords

Performance measures

$$\text{Recall} = \frac{\text{Number of relevant documents retrieved}}{\text{Total number of relevant documents in database}}$$

$$\text{Precision} = \frac{\text{Number of relevant documents retrieved}}{\text{Number of documents retrieved}}$$

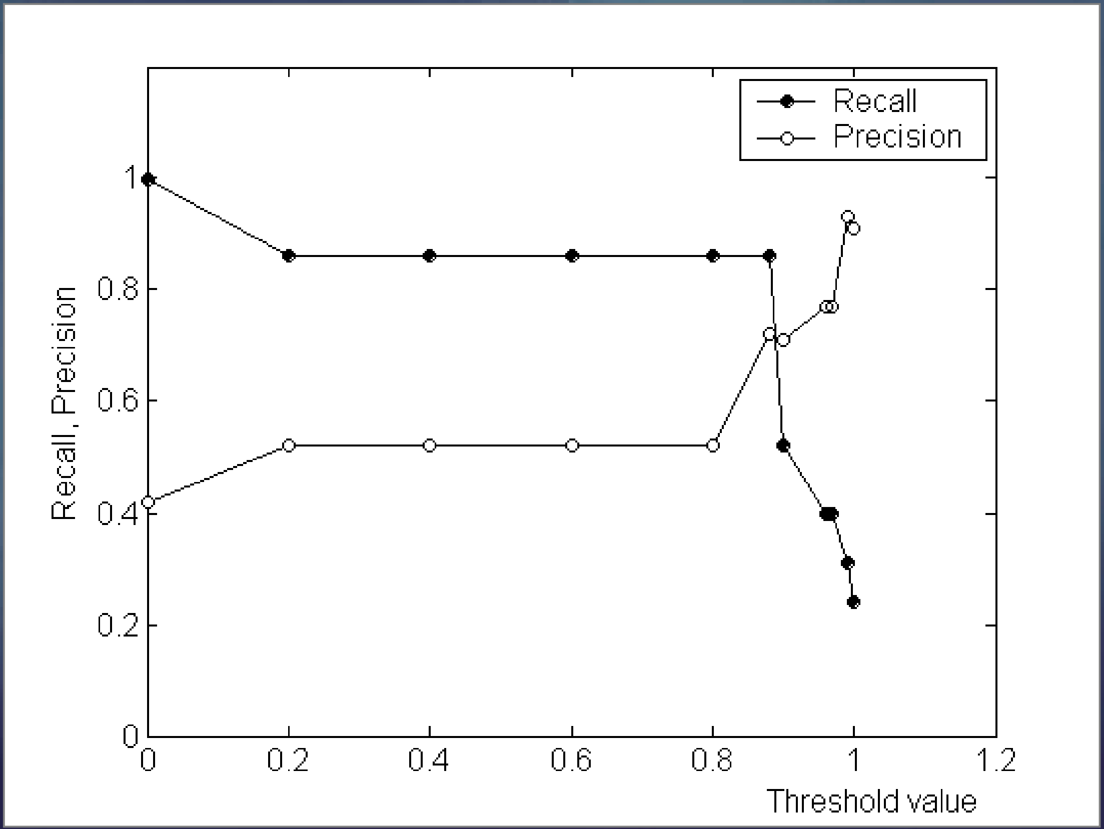
$$\overline{P}(r) = \sum_{i=1}^{N_q} \frac{P_i(r)}{N_q}$$

$\overline{P}(r)$: average precision at recall level r

N_q : number of queries

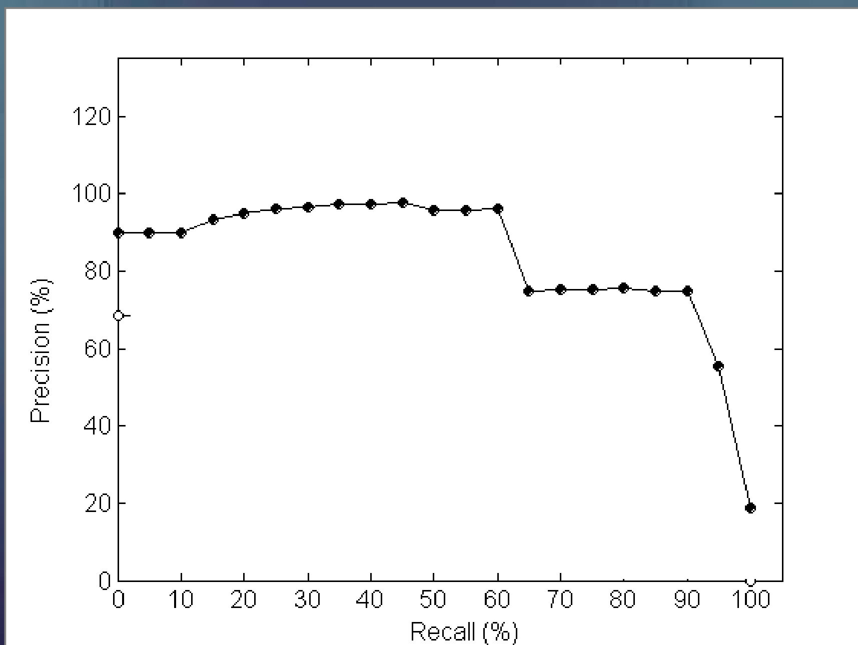
$P_i(r)$: precision at recall level r for the i-th query

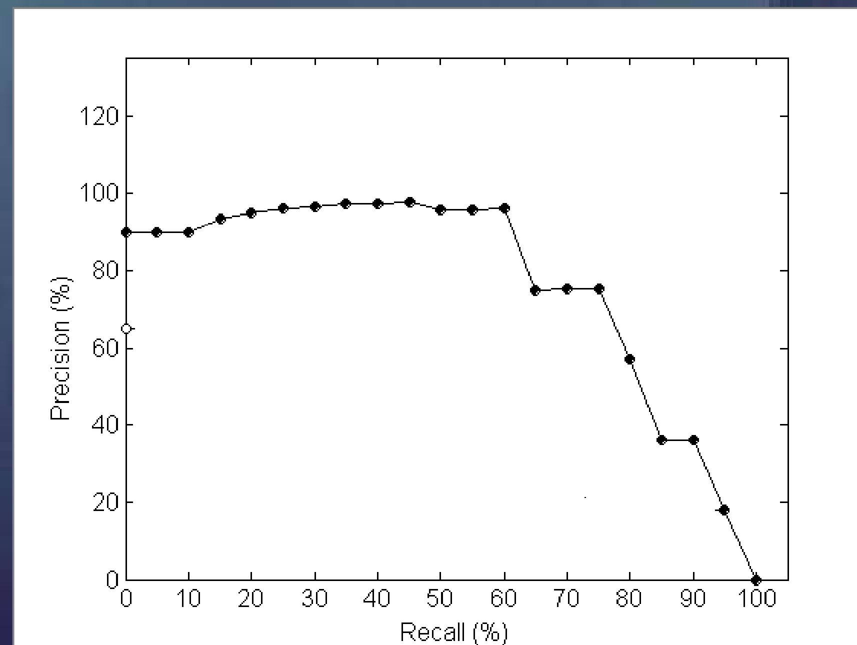
Threshold × recall/precision



Pedrycz and Gomide, FSE 2007

Precision × recall for composite queries





Pedrycz and Gomide, FSE 2007

15.2 Multiagent fuzzy systems

Agents and multiagents

- Agent: anything that can be viewed as perceiving its environment through sensors and acting upon that environment thorough actuators

- Rational agent: one that does the right thing

- Rationality depends on:

  – performance measure to define success
  – agent´s prior knowledge of the environment
  – actions that the agent can perform
  – agent percept history (sequence) to date

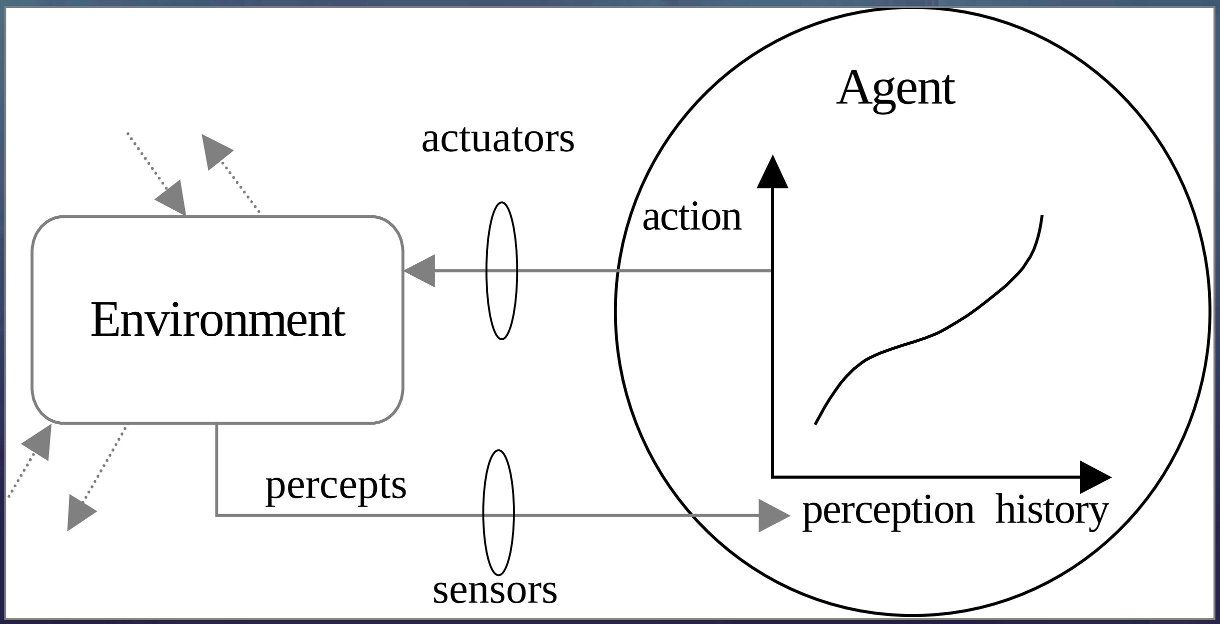
(Russel & Norvig, 2003)

Agents

- Computer systems that is situated in some environment, and that is capable if autonomous action in this environment to meet its design objectives
- Agents: autonomy is central
- Intelligent agents
  - reactive
  - proactive
  - social ability
  - learning

(Wooldridge, 2002)

- Agent function: abstract mathematical (computable) description
- Agent function of an artificial agent is implemented by an agent program



Agent function
(strategy)

Structure of agents

- Agent = architecture + programs
- Agent programs
  - table driven
  - reflex
  - model-based
  - goal-based
  - utility-based
  - learning agents

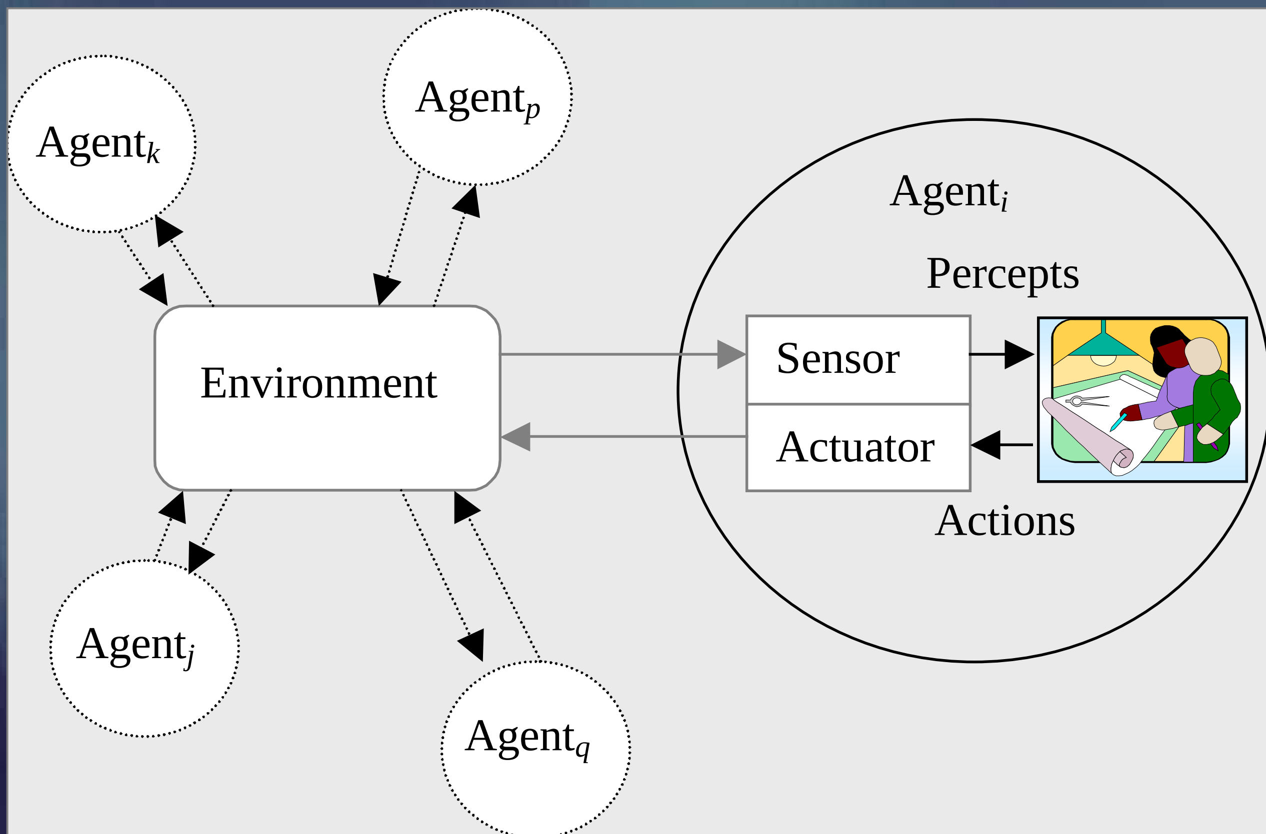
(Russel & Norvig, 2003)

Multiagents

- Systems composed of multiple interacting agents
- Multiagent systems concerns
  - individual gents
  - collection of these agents
  - interaction between agents
- Interactions involve
  - cooperation
  - negotiation
  - coordination

(Wooldridge, 2002)

Architecture of multiagent systems

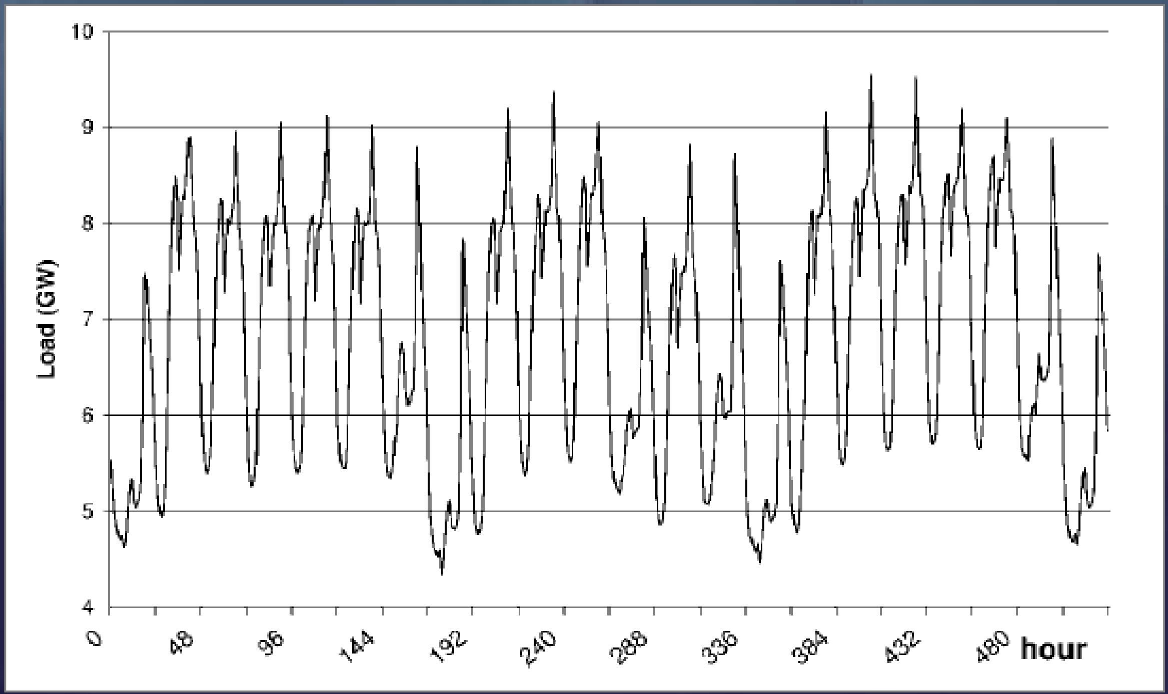


Electricity markets

- Power industry worldwide
  - auctions
    - resource allocation
    - systems coordination
- Power supplier in competing markets must
  - decide bidding strategy
  - maximize expected profit
  - market share

Demand

- Publicly known
- Auction is an ex-ante mechanism to allocate power hourly



Hourly load profile

Pedrycz and Gomide, FSE 2007

Running cost function

- Pool of thermal plants
- Cost of coal, gas, oil power plants

$$F(g_{jh}) = a + bg_{jh} + cg_{jh}^2$$

- Cost of nuclear plants: linear cost function
- Supplier cost function:

$$C(g_{jh}) = \alpha + \beta g_{jh} + \gamma g_{jh}^2$$

g_{jh} : active power supplied by plant j at hour h

Uniform price auction

1. Open the auction
2. Publish day ahead hourly load forecast
3. Accept bids from suppliers
4. Stop receiving bids
5. Apply a pricing procedure (e.g. merit order)
6. Publish the hourly price π_h, $h = 1,\ldots,24$
7. Inform each supplier the power to be produced for 24 h
8. Close the auction

- Power demand inelastic with price
- Auctioneer must assure D_{h} is met at all h
- Conservative agent bid pairs (MW, \$): (q_{jh}, p_{jh})
- Intelligent agent: free to choose price and amount

$$D_h = \sum_{j=1}^{T_h} g_{hj}$$

Supply = demand (MW)

$$P_{jh} = \pi_h g_{hj} - C_j(g_{jh})$$

Supplier profit ($)

$$MC_j = \left.\frac{\partial C_j(g_{jh})}{\partial g_{jh}}\right|_{g_{jh}=G_j}$$

Marginal cost ($/MW)

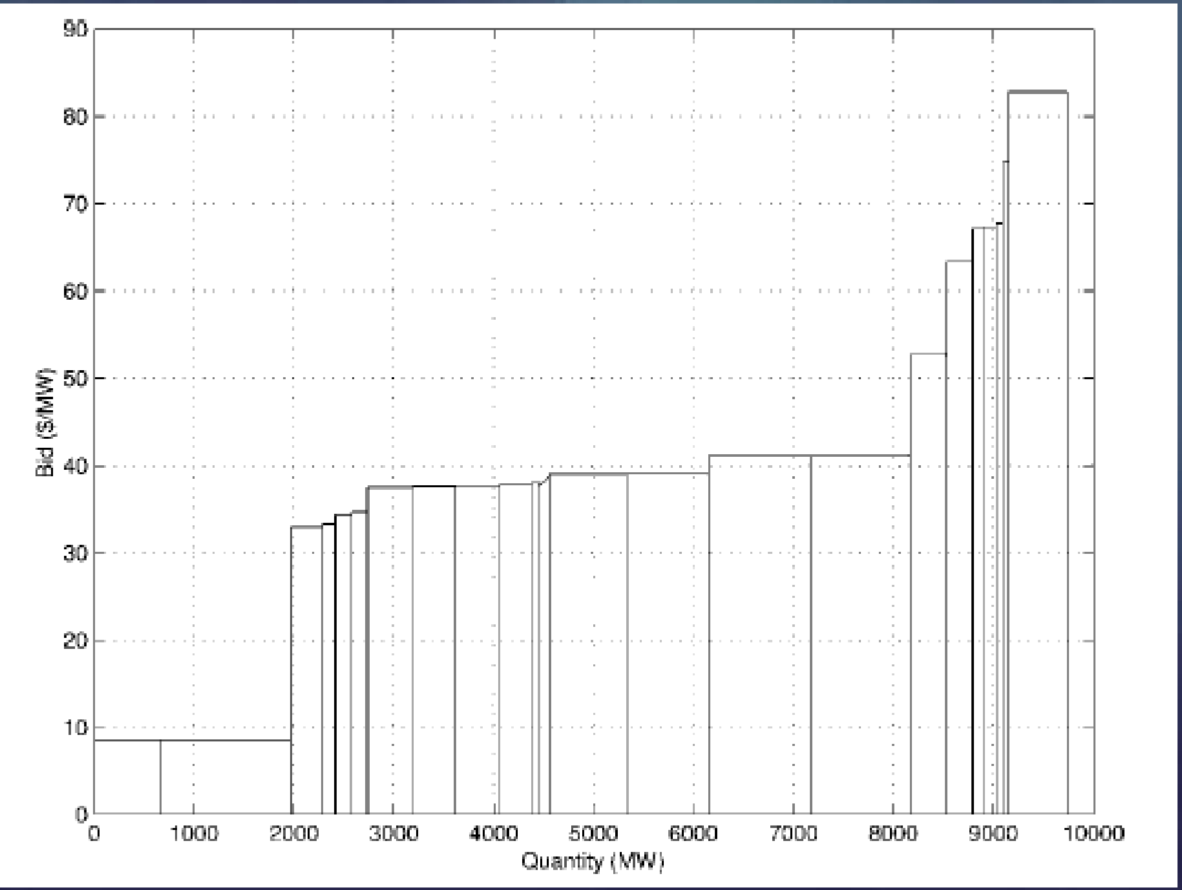
Market players

Table 1 Thermal plants characteristics

Plant	Type	G_j	$MC_j(G_j)$	$C_j(.)$
Angra 1	Nuclear	657	8.5	$8.5g$
Angra 2	Nuclear	1309	8.5	$8.5g$
P.Medici 3-4	Coal	320	32.95	$865.3 + 28.914g + 0.0063g^2$
P.Medici 1-2	Coal	126	33.33	$343.34 + 28.53g + 0.01905g^2$
TermoBahia	Gas	171	34.38	$580.54 + 30.985g + 0.00992g^2$
TermoCeara	Gas	153	34.72	$505.29 + 30.558g + 0.0136g^2$
Canoas	Gas	450	37.54	$1575.22 + 33.869g + 0.00408g^2$
N.Fluminense	Gas	426.6	37.63	$1484.92 + 33.759g + 0.00454g^2$
Araucaria	Gas	441.6	37.70	$1505.65 + 33.782g + 0.00443g^2$
Tres Lagoas	Gas	324	37.76	$1115.29 + 33.595g + 0.00643g^2$
Corumba	Gas	79.2	38.03	$278.97 + 33.256g + 0.03016g^2$
Juiz de Fora	Gas	103	38.73	$323.68 + 33.088g + 0.0274g^2$
Ibirite	Gas	766.5	39.07	$3632.08 + 31.966g + 0.00463g^2$
TermoRio	Gas	824.7	39.11	$3904.05 + 31.912g + 0.00436g^2$
Argentina I	Gas	1018	41.04	$4459.61 + 32.775g + 0.00406g^2$
Argentina II	Gas	1000	41.05	$4379.82 + 32.774g + 0.00414g^2$
J.Lacerda C	Coal	363	52.64	$1547.15 + 45.962g + 0.00919g^2$
J.Lacerda B	Coal	262	63.30	$1407.65 + 56.198g + 0.01356g^2$
J.Lacerda A1-2	Coal	100	67.10	$549.89 + 57.895g + 0.04605g^2$
J.Lacerda A3-4	Coal	132	67.35	$728.6 + 57.65g + 0.03674g^2$
Charqueadas	Coal	69.1	67.72	$414.59 + 60.037g + 0.05559g^2$
FAFEN	Gas	57.6	74.78	$417.18 + 66.857g + 0.06879g^2$
Uruguaiana	Gas	582	82.77	$4306.82 + 76.729g + 0.00519g^2$

Pedrycz and Gomide, FSE 2007

Market supplier function

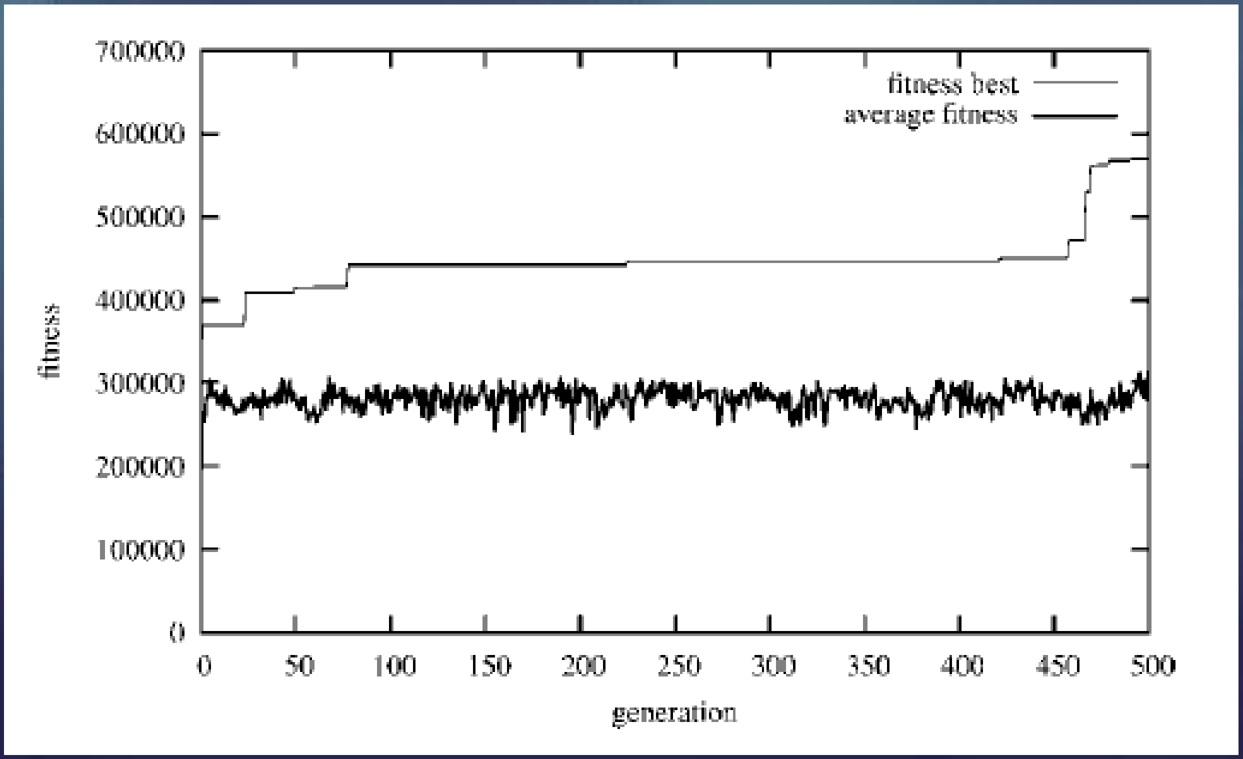


agents bid marginal prices at full capacity)

Intelligent agent ≡ genetic fuzzy system

```
procedure GFRBS-ALGORITHM (X,Y,f) returns a rule base
input : universes X,Y
        fitness function: f
local:  population: set of individuals
        crossover rate, mutation rate
          max: maximum number of generations
          INITIALIZE(population, number individuals)
          repeat
            evaluate each individual using f
            select parents in population using relative fitness
            apply crossover and mutation on parents
            create new population
          until number generations ≥ max
          return rule base
```

Example



fitness best
average fitness
700000
600000
500000
400000
300000
200000
100000
0
fitness
0
50
100
150
200
250
300
350
400
450
500
generation

Rule base

$10010 \Rightarrow 01010101\ 111011$
$11111 \Rightarrow 01100000\ 110011$
$11011 \Rightarrow 11100010\ 010110$
$00010 \Rightarrow 01010101\ 011011$
$00011 \Rightarrow 01011010\ 010111$
$01010 \Rightarrow 11110110\ 011111$

If X_1 is $(A_{11}\ or\ A_{14})$
then Y_1 is $(C_{12}\ or\ C_{14}\ or\ C_{16}\ or\ C_{18})$
and Y_2 is $(C_{21}\ or\ C_{22}\ or\ \mathrm{C}_{23}\ or\ C_{25}\ or\ C_{26})$

First rule

- Best bid strategy: two weeks test period
  - profit 36.7% higher
  - 91.3% more energy produced than conservative strategy
- Rule-base semantics
  - when demand is low and price below its marginal cost at full capacity, the agent bids lower price and a quantity that minimizes loss
  - increases price when he has opportunity to be the marginal supplier

15.3 Distributed fuzzy control

Resource allocation

$$\max \; J(\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n)$$

$$s.t. \;\; \sum_{i=1}^{n} \mathbf{u}_i \le r$$

$$\mathbf{u}_i \ge 0 \;, i = 1, \ldots, n$$

Economic system

- Agents
  - producers
  - consumers
- Price system
- commodities

Artificial market

- Agents
  - producers: maximize profit
  - consumers: maximize utility
- resource vector $\mathrm{u} \in \mathrm{R}^{\ell}$
- consumer agents: n , $f_i(u_i^c)$, $u_i^c \in \mathrm{R}^{\ell}$ (f_i = utility)
- producer agents: m, $f_j(u_j^p)$, $u_j^p \in \mathrm{R}^{\ell}$ (f_j = profit)

Optimal market allocation

$$\max_{\mathbf{u}_i^c} \quad f_i(\mathbf{u}_i^c)$$

$$\max_{\mathbf{u}_i^p} \quad f_j(\mathbf{u}_j^p)$$

$$s.t. \ \sum_{i=1}^{n} \mathbf{u}_i^c = \sum_{j=1}^{m} \mathbf{u}_j^p$$

$$\mathbf{u}_i^c, \mathbf{u}_j^p \geq 0, \ i = 1, \ldots, n; \ j = 1, \ldots, m$$

Control systems and economy

$$\mathbf{x}_i(t+1) = f_i(\mathbf{x}(t), \mathbf{u}_i(t), t)$$

$$\mathbf{y}_i(t) = g_i(\mathbf{x}(t), t), \;\; \mathbf{x}_i(0) = \mathbf{x}_{io}$$

$$\mathbf{x} \in R^q, \; \mathbf{u}_i \in R^{\ell}, \; \mathbf{y}_i \in R^s$$

Coupled dynamic systems

$$\min_u F_i(|u_1(t)|, \ldots |u_n(t)|, x_o)$$

$$s.t. \;\; \mathbf{x}_i(t+1) = f_i(\mathbf{x}(t), \mathbf{u}_i(t), t)$$

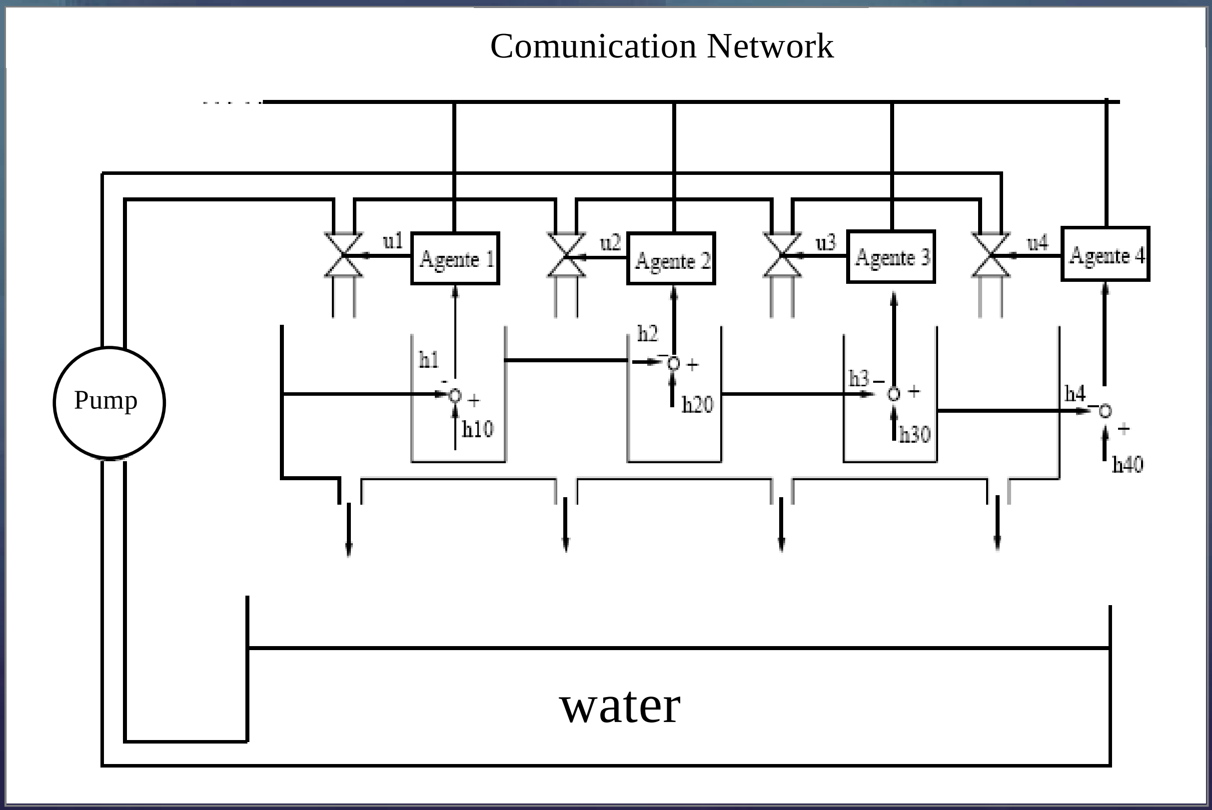
$$\mathbf{y}_i(t) = g_i(\mathbf{x}(t), t)$$

$$|\mathbf{u}_i(t)| \geq 0$$

$$\mathbf{x}_i(0) = \mathbf{x}_{io}$$

Distributed resource allocation problem

Fuzzy market-based control



$\mathbf{h}_s \equiv$ set-points

$\mathbf{u}_o \equiv$ operation point

$\Delta h_i(t)$ input variable for i^{th} agent

$u_i(t) = u_{io} + \Delta u_i(t)$ decision of i^{th} agent

Market-based control algorithm

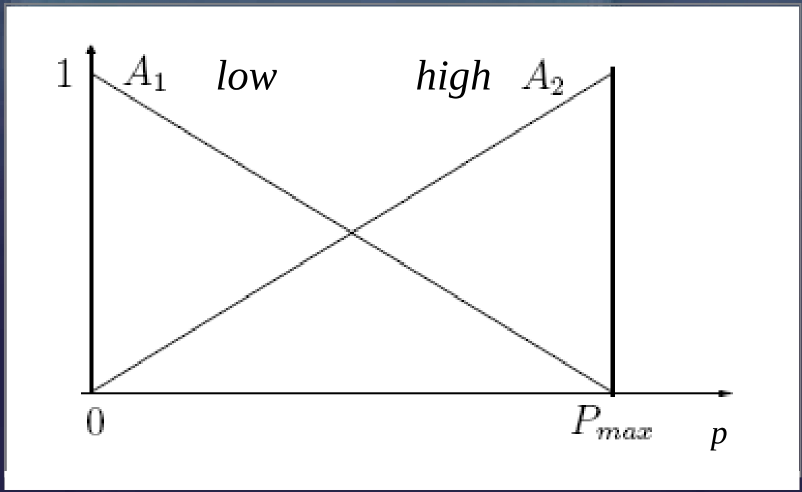
procedure DISTRIBUTED-FUZZY-AUCTIONEER (*d*,*s*) **returns** price
input : demand: *d*
supply: *s*

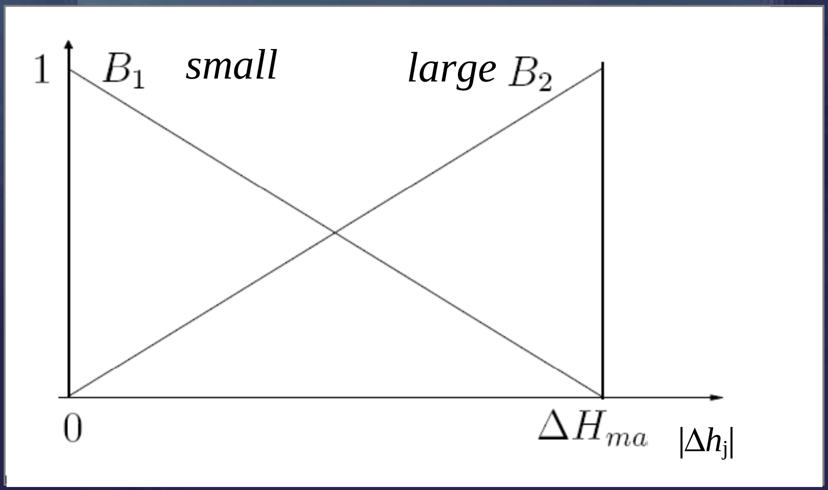
for each agent **do**
if demand agent **then** get demand
if consumer agent **then** get supply
compute equilibrium price
run auctions for the training period
store individual fitness ϕ
remove individual from the market
return price

$$r_{pp}(t) + \sum_{j=1}^{m} r_j^p(t) = r_{cp}(t) + \sum_{i=1}^{q} r_i^c(t)$$

Consumer agent

If price is *low* and deviation is *small* then demand is g_{do}
If price is *low* and deviation is *large* then demand is g_{dmax}
If price is *high* then demand is g_{do}





$$g_{do}(p(t), \Delta h_i(t)) = 0, \ \forall p(t), \Delta h_i(t)$$

$$g_{d\max}(p(t), \Delta h_i(t)) = r_{d\max}, \ \forall p(t), \Delta h_i(t)$$

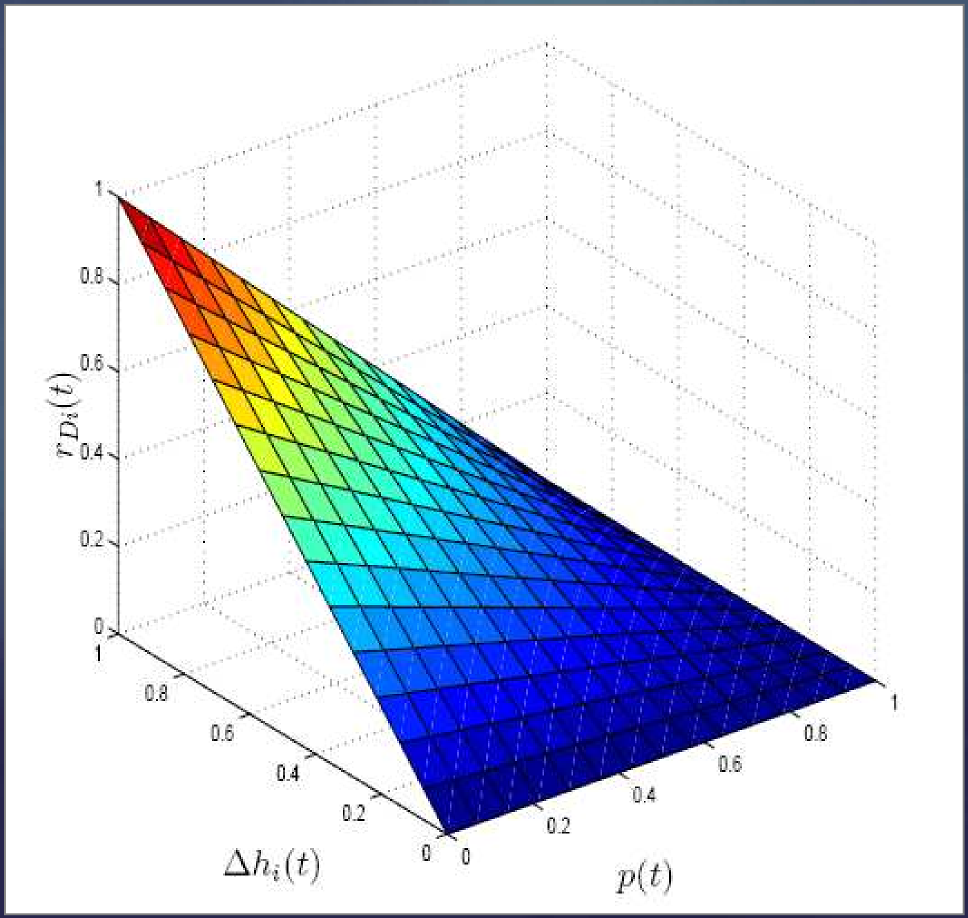
$$\Downarrow$$

$$r_i^c(t) = (k_{d1} - k_{d2}\, p(t))\Delta h_i(t)$$

$$k_{d1} = (r_{d\max} / \Delta H_{\max})$$

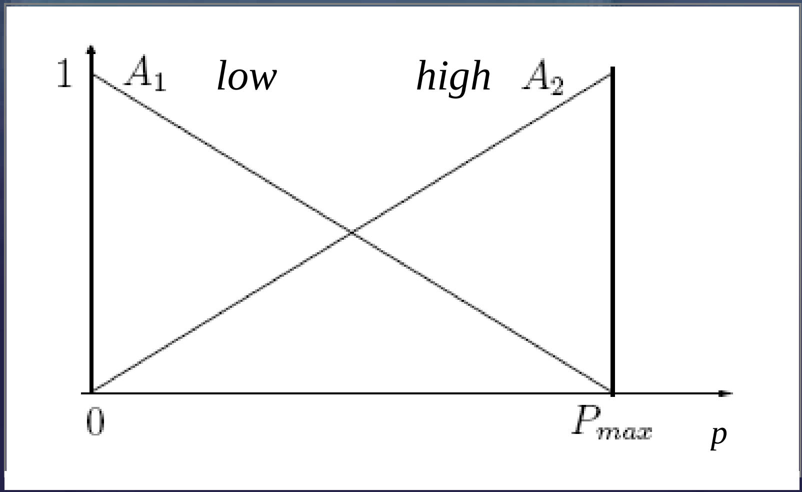
$$k_{d2} = (k_{d1} / P_{\max})$$

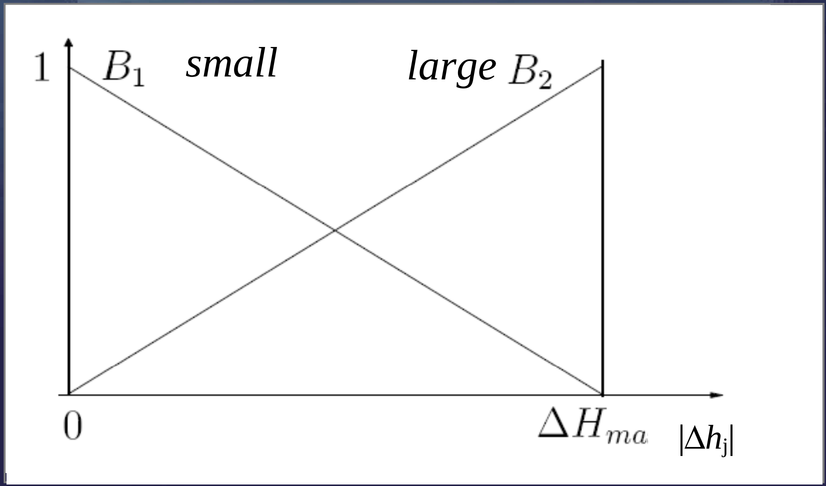
Demand function of a consumer agent



Producer agent

If price is *low* then supply is g_{so}
If price is *high* and deviation is *small* then supply is g_{so}
If price is *high* and deviation is *large* then supply is g_{smax}





$$g_{so}(p(t), \Delta h_i(t)) = 0, \ \forall p(t), \Delta h_i(t)$$

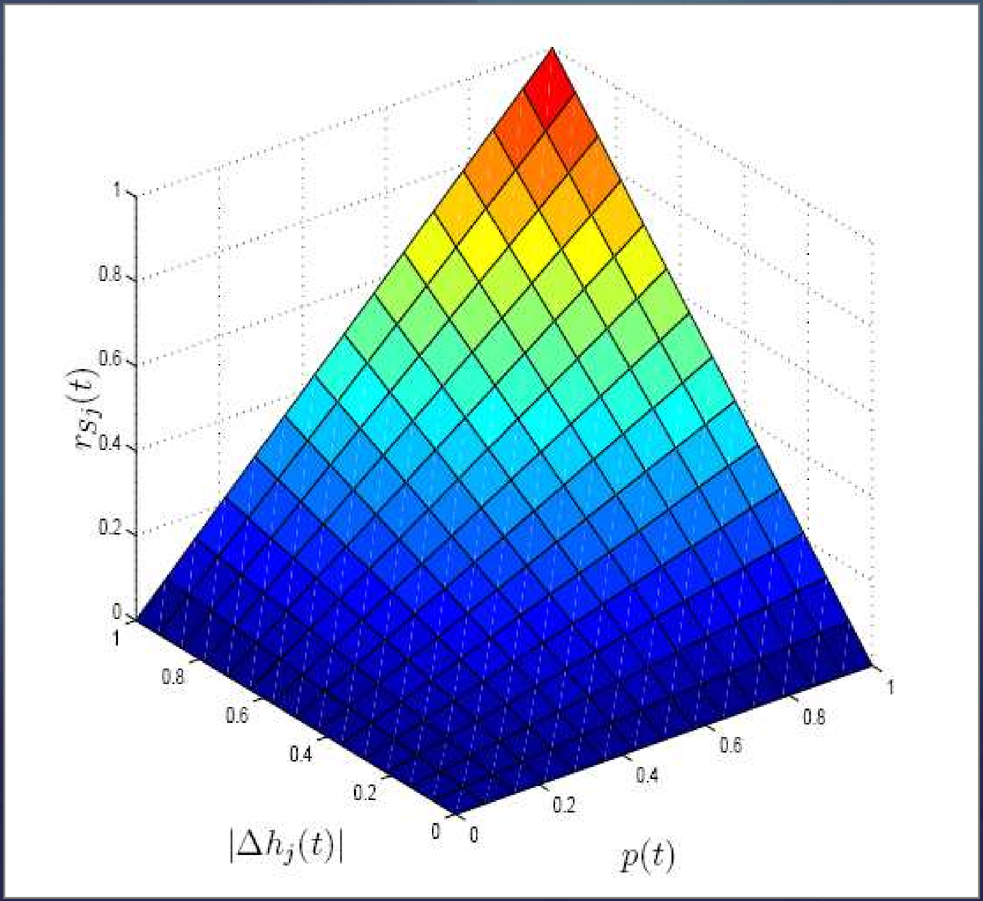
$$g_{s\max}(p(t), \Delta h_i(t)) = r_{s\max}, \ \forall p(t), \Delta h_i(t)$$

$$\Downarrow$$

$$r_j^p(t) = k_s p(t) \,|\, \Delta h_i(t) \,|$$

$$k_s = (r_{s\max} / (P_{\max} \Delta H_{\max}))$$

Supply function of a producer agent



Market equilibrium

Pump as a permanent producer : $r_{pp}(t) = \delta p(t)$

Pump as a permanent consumer: $r_{cp}(t) = \alpha_1\left(1 - \frac{p(t)}{\alpha_2}\right)$

Equilibrium prices

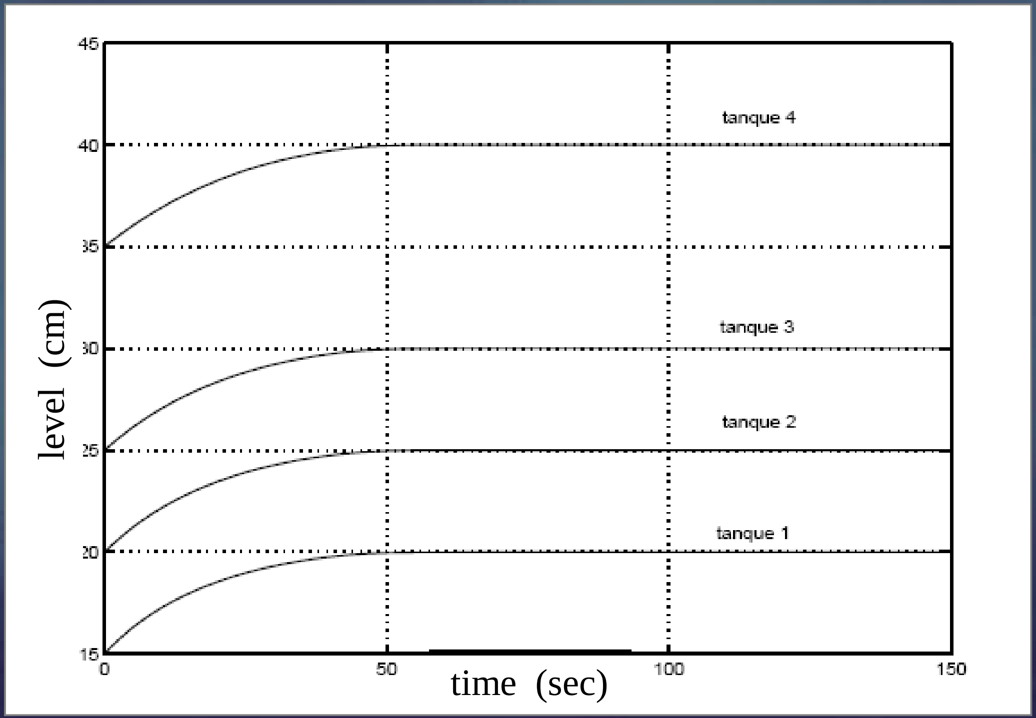
$$p(t) = \frac{\alpha_1 + k_{d1}\sum_{i=1}^{q}\Delta h_i(t)}{\delta + (\alpha_1/\alpha_2) + k_s\sum_{j=1}^{m}|\Delta h_j(t)| + k_{d2}\sum_{i=1}^{q}\Delta h_i(t)}$$

Controls

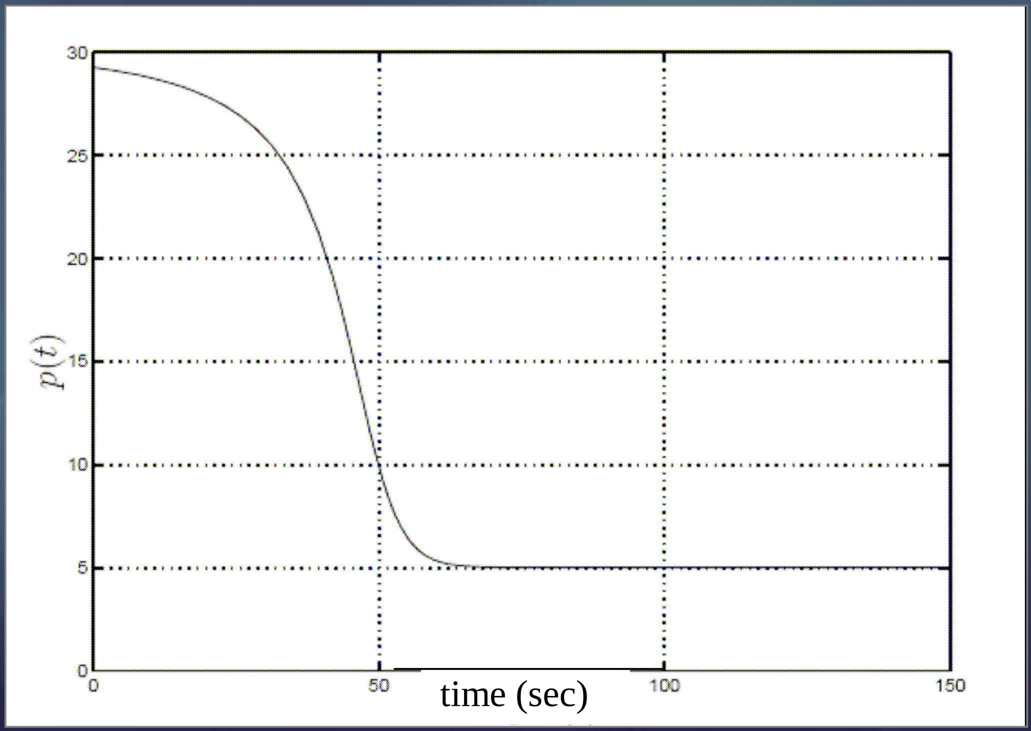
$\Delta u_i(t) = (k_{d1} - k_{d2}p(t))\Delta h_i(t)$ consumer

$\Delta u_s(t) = k_p p(t))\Delta h_j(t)$ producer

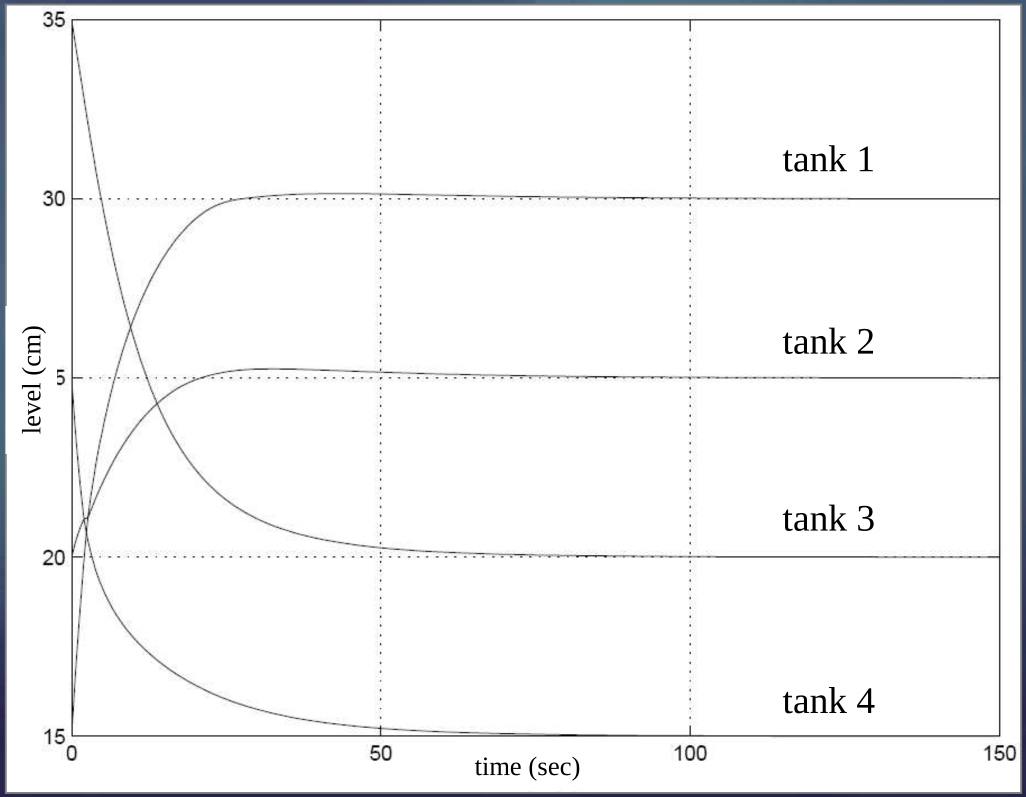
Example 1: Tank level



Equilibrium prices



Example 2: Tank levels



Equilibrium prices