Agent advanced features for Automatic Negotiation and Coordination of joint Actions

by

Eugénio Oliveira
Faculty of Engineering - University of Porto
and LIACC
Portugal

OUTLINE

- **Aims** of the Presentation
- 1 The Negotiation Process:
  - Definitions and characteristics
- 2 Our Negotiation Model
  - Agents' features for dynamic scenarios
- 3 Agents' Decision-Making
  - Tactical behaviuor
  - Strategic behaviour
- 4 Agents' Learning Capability
  - Reinforcement Learning
  - Q-Learning Algorithm

OUTLINE

- Agent System for Electronic Commerce (App.1)
- 5 Agents for B2B
  - Virtual Enterprises Formation (App.2)
  - Negotiation Model enhancement: Multi-Criteria
  - Q-Negotiation
- 6 Agents' Coalition formation
  - Inter and Intra Coalitions negotiation Protocols
  - MAS for Distributed Resources Management (App.3)
- 7 Coordinating a team of Agents
  - Knowledge for Coordination (App.4)
- 8 Conclusions and Future research

Aims of the Presentation

- ✧ Which Agents' features for flexible Negotiation?
- ✧ What kind of Learning methods Agents to use?
- ✧ How to benefit from Negotiation in several Application domains?
- ✧ How to Coordinate a team of Agents

Negotiation: Classification

- 1 The Negotiation Process:
- Definitions:
  - The goal of the Negotiation process is the maximisation of the utility of a future decision.
  - Agents get involved in a Negotiation process if it is individually rational

- Negotiation process Characteristics:
  - **Participants:**
    - Bilateral Vs Multi-lateral
    - Single-sided Vs Double-sided
    - Coalitions permitted or not

Negotiation: Classification

- **Goods:**
  - Single-item Vs Multiple items
  - Multi-Attribute Vs Single-Attribute (price)

- **Process:**
  - Forward Vs Reverse
  - Single-stage Vs Multi-stage
  - Open-cry Vs Sealed-bid
  - Agent-mediated Vs Manual bidding
  - Integrative Vs Distributive

Negotiation Model: The Actors

- 2 Our Negotiation Model:
  - Case-studies:

    i) E. Commerce,

    ii) Resources Management,

    iii) Virtual Enterprises Formation

- Environments' characteristics:
  - Open, Dynamic, Competitive
- Cognitive Agents:
  - Intelligent (Flexible and Autonomous), Self-interested
  - Decision-Making capabilities, Inf. Privacy
  - Adaptive

Negotiation Model: Characteristics

- Participants:
  - Multi-lateral, Single-sided, Collusive
- Goods/Services:
  - Multi-Attribute
- Process:
  - Reverse and Forward, Single or Multi-stage
  - Sealed bid, Integrative and Distributive
  - Agent-mediated

Agents' Autonomy : Decision Making

- 3 Agents' Decision-Making

  i) for E Commerce

- **Tactic**: Linear combination of Functions that generate a value for a single negotiation issue.

  - **Time**-dependent (predictable factor)
  - **Resources**/Environment dependent (u.)
  - **Behaviour**-dependent (u.)

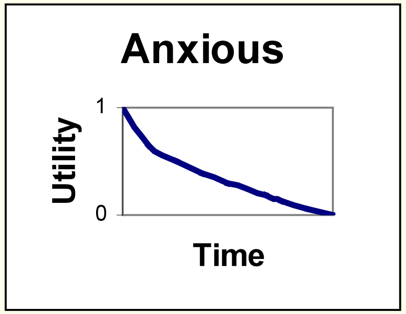
  See Faratin et al, "Negotiation decision functions for Autonomous Agents", IJRobots and Atonomous Systems, 24 (3-4), pp.159-182

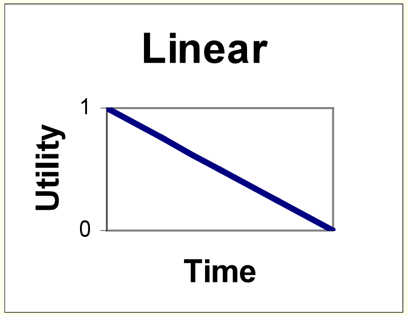
- Time-dependent agents concede as the deadline approaches

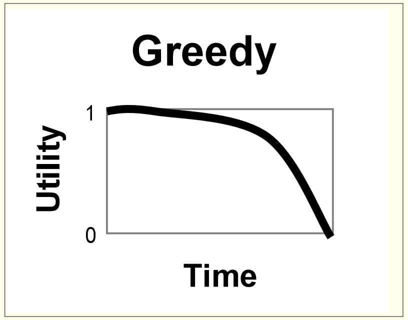
Decision Making: Tactics

- Resource/Environment dependent: proposals vary with the availability of resources or the number of opponents
- Behaviour dependent agents imitate the behaviour of their opponents

- Time-dependent tactics







Decision-Making: Tactics

- "Resource"-dependent tactics
  - depend on run-time factors: presence of opponents, previous deals,...
- Behaviour-dependent tactics:
  - depend on run-time factors: behaviour of opponents
    - Absolute Tit-for-Tat
    - Relative Tit-for-Tat

Decision-Making: Strategy

- **WHAT can be learned?**
  - Select the best proposals to make
  - Select the best Negotiation Protocol
  - Other Agents' models (No)

- Strategic Behaviour:
  - Strategy is the way weighted combinations of tactics change over time

Learning Agent methods

- 4 Agents' Learning Capability
  - Learning from the interaction with the environment

    ☝ **Reinforcement Learning**
  - Value Function: function of states (or state-action) that estimate how good is for an Agent to be in a state (or perform an action)

  - Dynamic Programming is used to compute optimal policies given a perfect model of the environment

  - Monte Carlo Methods require only learning from experience. Experience divided into episodes.

Reinforcement Learning

- Temporal Diference: a combination of MC and DP

- Reinforcement Learning is Temporal Diference Learning
  - On-Policy Vs Off-Policy methods:

- Q-learning - Off-Policy, Temporal Difference Control algorithm for reinforcement Learning

  - *Watkins, C.J., "Learning from delayed rewards" PhD Thesis, Cambridge University, UK.*

Q-Learning

- *Q* values associated with state-action pairs:

  $Q(s,a) = Q(s,a) + \alpha\, [r + \gamma\, max_{a'} Q(s',a') - Q(s,a)]$

- Where:
  - α: learning rate
  - r: reward achieved by executing *a* in *s*
  - γ: discount factor
  - $max_{a'} Q(s',a')$: maximum *Q* for actions in the next state

Q-Learning

- Action selection - exploitation vs. exploration
  - *ε -greedy*: a small probability ε of uniformly choosing a non-greedy action
- *Softmax*: a degree of exploration τ for choosing actions according to their ranking

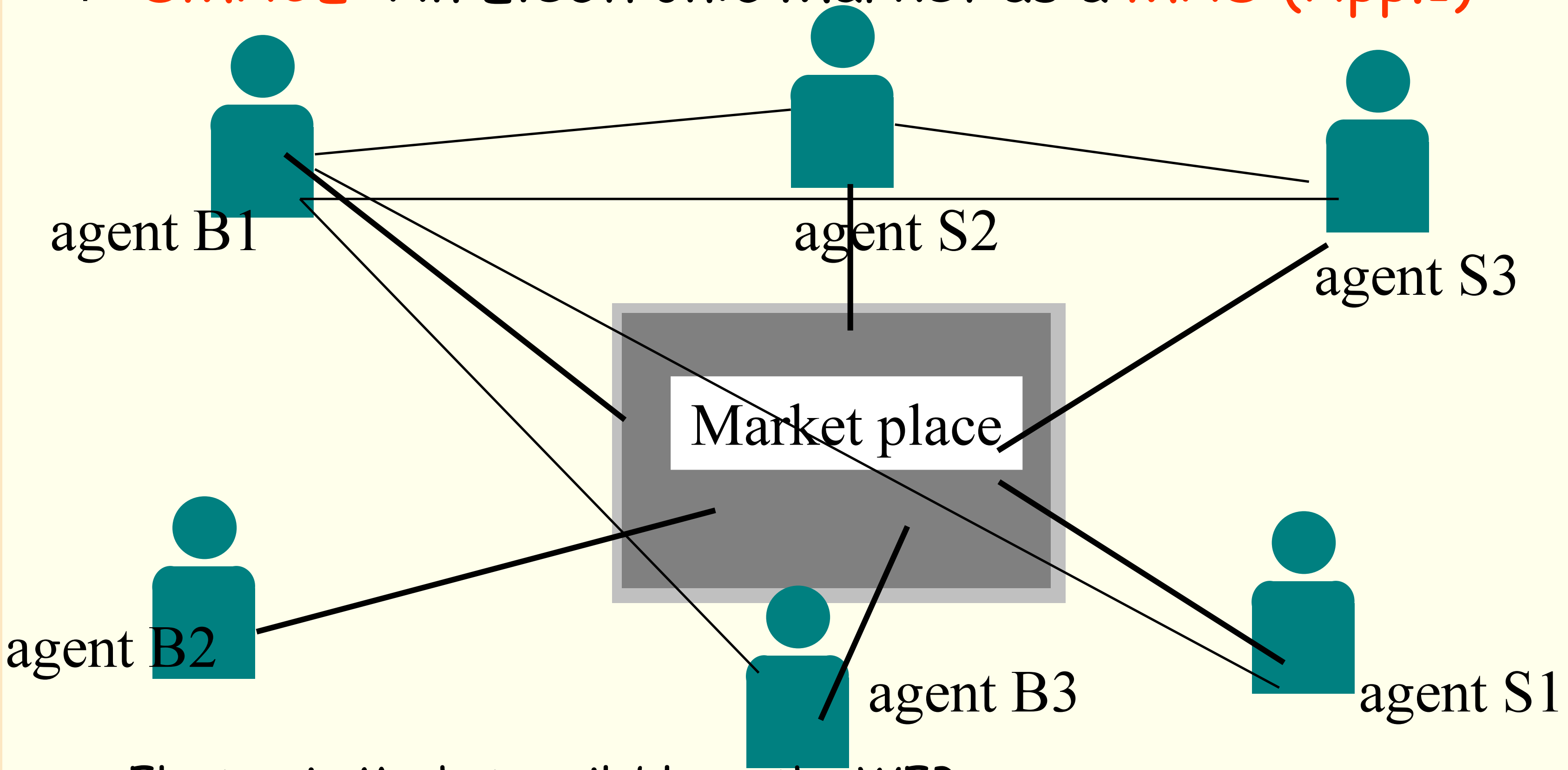
Choosing the best action (Softmax approach):

$$p_t(a) = \frac{e^{Q_t(a)/\tau}}{\sum_{b=1}^{n} e^{Q_t(b)/\tau}}$$

Where: p(a) - probability of choosing action a
τ - temperature ≥ 0

Agent System for Electronic Commerce

- SMACE: An Electronic Market as a MAS (App.1)



Electronic Market available on the WEB:
http://www.fe.up.pt/~eol/PROJECTS/SMACE_intro.html

Adaptive Agents for negotiation (App.1)

SMACE

Agents:

- User-defined
  - *MTA*
    - combination of 3 tactics, one from each family
    - user specifies the tactic weights
- Adaptive behaviour
  - *ABA*
    - dynamic combination of several tactics
    - uses Q- *Learning* to find out the best combination for each situation

Adaptive Agents for Negotiation (App.1)

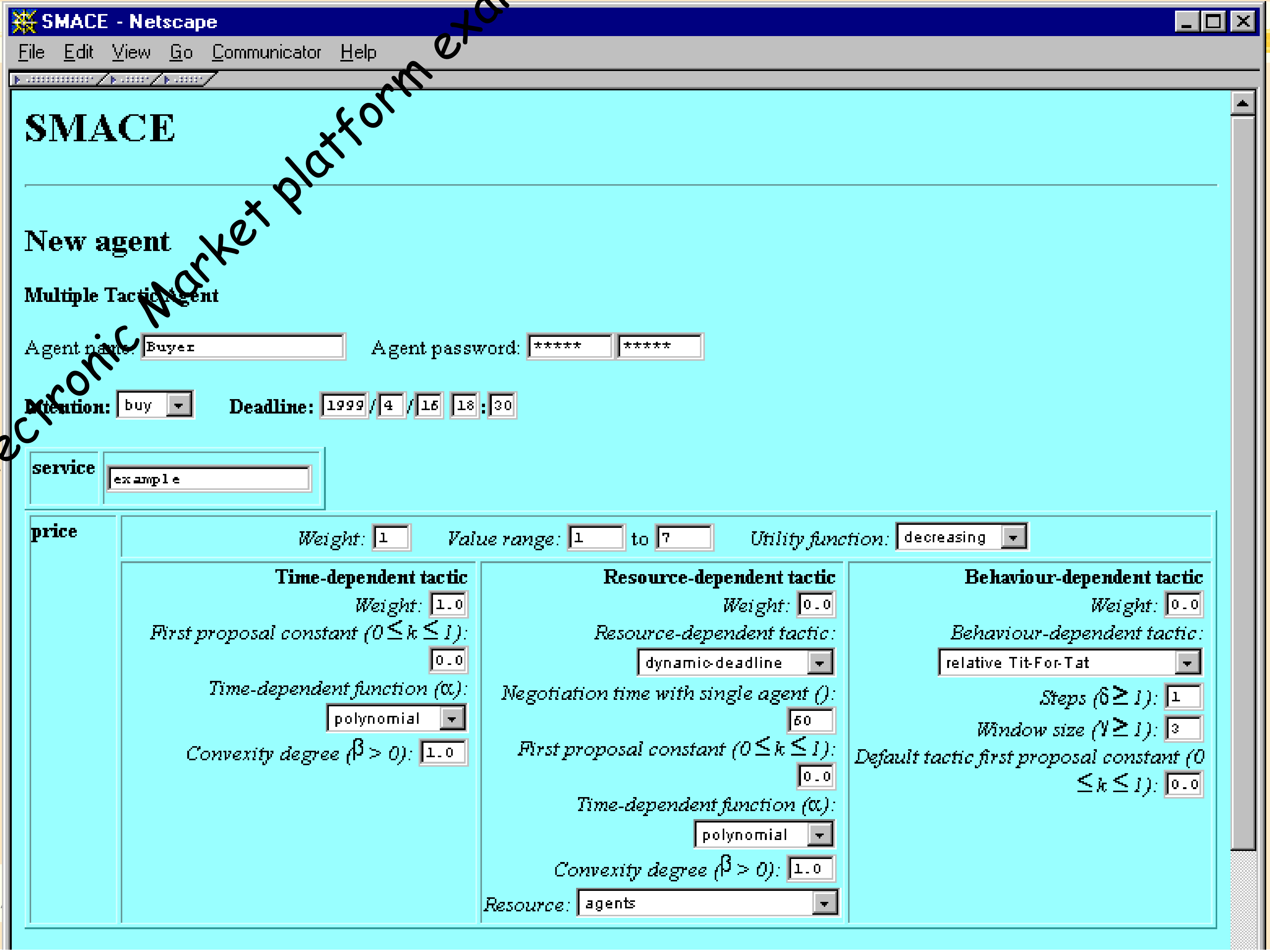
SMACE

SMACE is implemented in JAVA using JatLite
It consists of 3 layers:

- Infrastructure
  Agent templates to create new agents
  Market Place (augmented Jatlite Router)
- Plug &Trade
  Pre-defined Negotiating agents for exemplification
  MTA
  ABA
- User Interface
  providing HTML interfaces for both creating and monitoring agents' operation

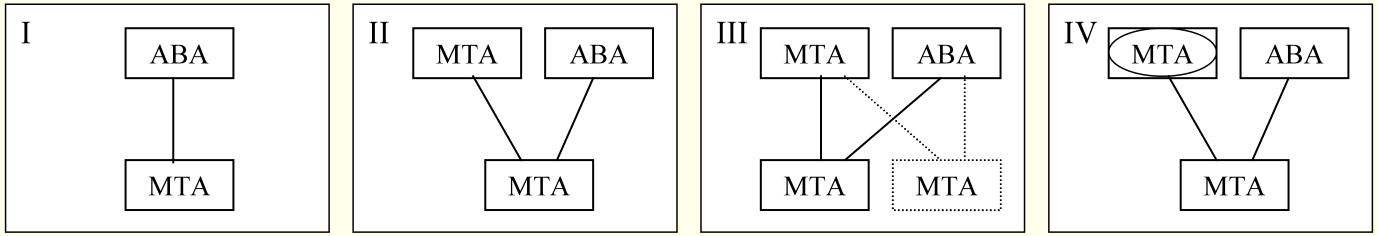
Adaptive Agents for Negotiation

Electronic Market platform example:



Adaptive Agents for Negotiation

- Experiments to test the learning skills of the adaptive agents (ABA)

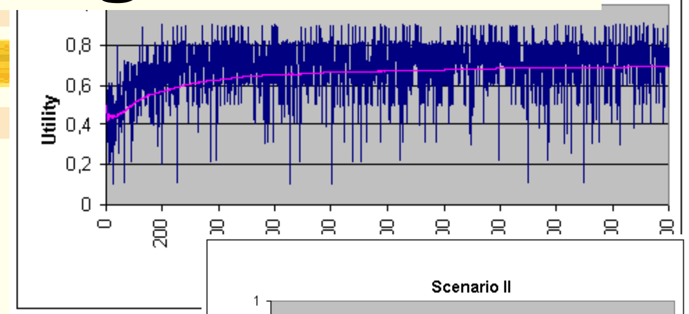


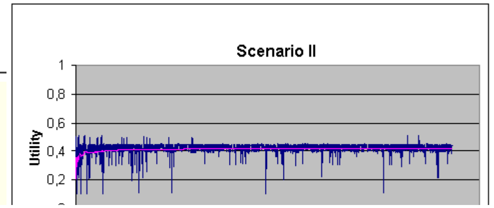
  - Scenario I: utility variation
  - Scenario II: utility and number of deals won
  - Scenario III: reaction to changing environment
  - Scenario IV: re-adaptation

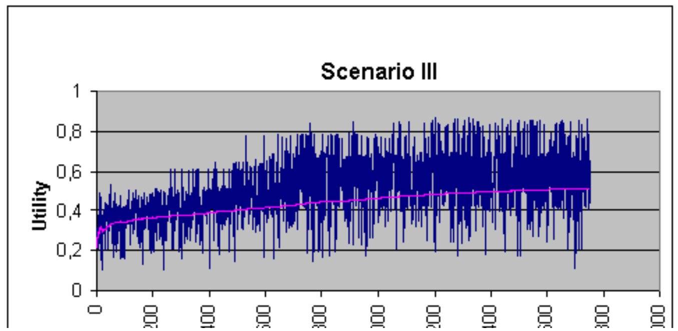
Adaptive Agents for Negotiation

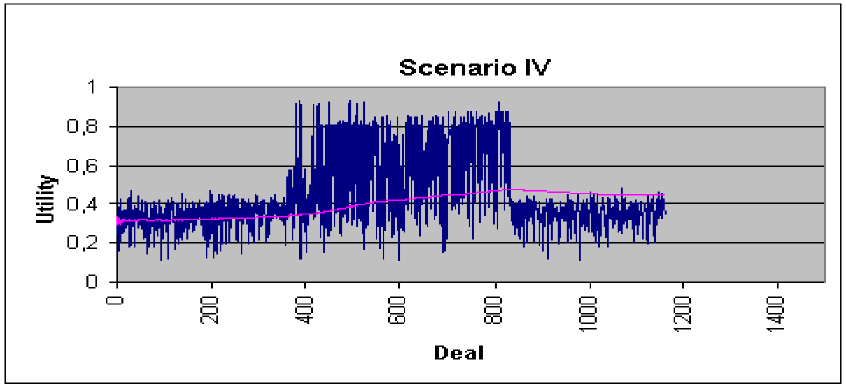
ABA Results

- Scenario I:
  - continuous increase of utility
- Scenario II:
  - majority of deals won, utility close to the best possible
- Scenario III:
  - got higher utilities with late arriving agent
- Scenario IV:
  - quick re-adaptation







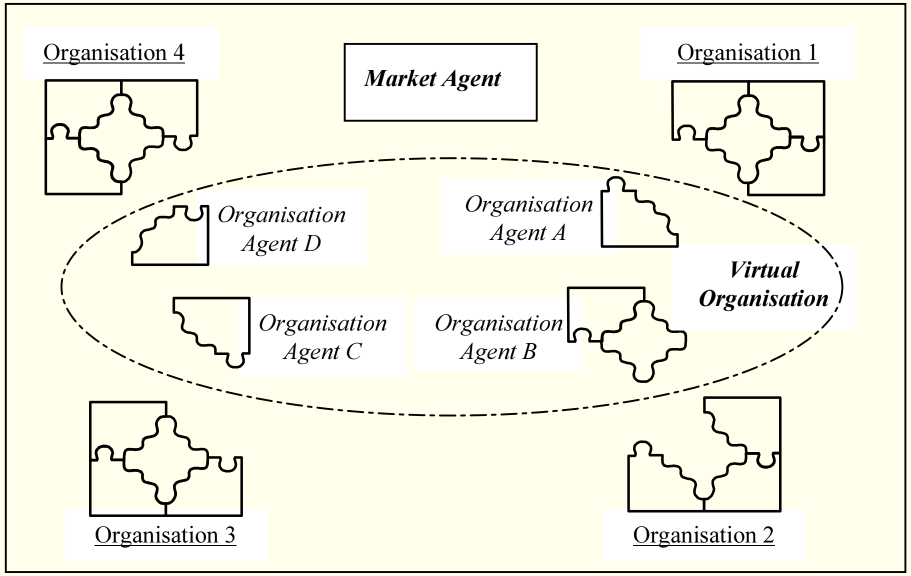


Virtual Enterprise Formation (App.2)

- 5 Agents for B2B
  - Virtual Enterprises Formation (App.2)
  - **Virtual Organisation definition:**
    Temporary network of independent companies playing different roles (suppliers, costumers, sellers, buyers,...) connected through a network in order to share skills and competencies to access new markets
  - VE Life Cycle:
    Identification of Needs, Partners selection, Operation, Dissolution

Virtual Enterprise Formation (App.2)

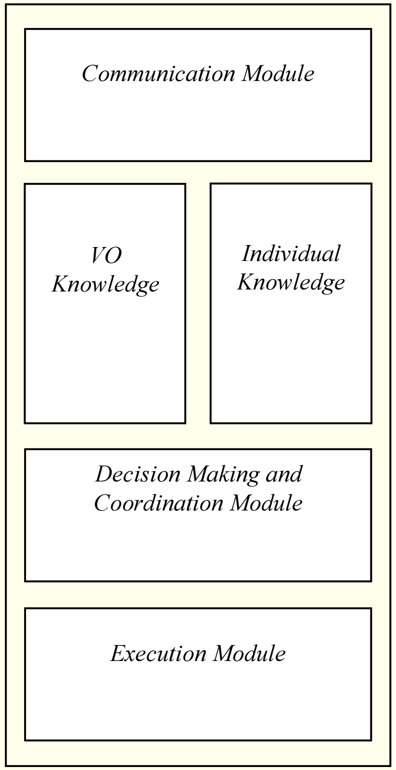
- MAS Architecture for VO formation



Virtual Enterprise Formation (App.2)

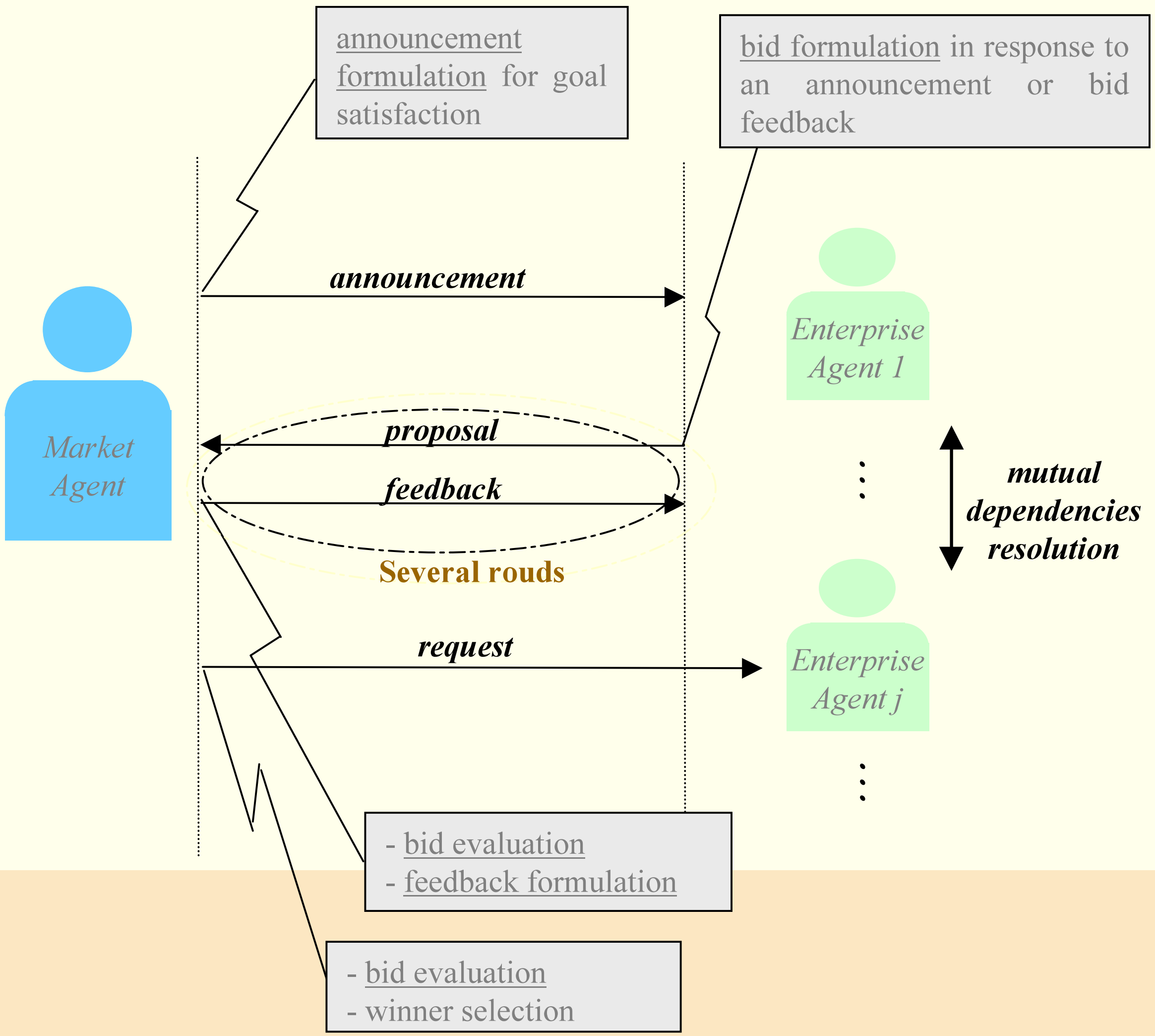
✧ Agents' Architecture

Organisation Agent



Market Agent

Negotiation Protocol diagram(App.2)



Q-Negotiation (App.2)

- Q-Negotiation algorithm:
  - Organiser (Market Agent)
    - Multi-issue evaluation
    - Qualitative feedback after bid evaluation
  - Respondent (Enterprise Agent)
    - Bid formulation using a Q-learning algorithm
    - Mutual dependencies resolution

keeping preferences private

Q-Negotiation (App.2)

- Market Agent (organiser):
  - Multi-issue bid evaluation:
    - Quantifies the proposal values' deviation from the optimal ones.

$$Ev = \frac{1}{Dev}, \qquad Dev = \frac{1}{k} * \sum_{i=1}^{k} \frac{i}{k} * dif\left(PrefV_i, V_i\right)$$

  - Qualitative feedback formulation:
    - Specifies the distance between the values included in the current proposal and the best one received so far, in terms of a qualitative comment (sufficient, bad, very_bad).
    - This means that there is a better proposal in the market

Q-Negotiation (App.2)

✧ Enterprise Agent (respondent)

✧ Bid formulation

- ✧ Uses an algorithm based on Q-learning.
- ✧ Reward values are calculated according to the received qualitative feedback.

$$r = \begin{cases} k & , \text{if winner} \\ \frac{k}{2} - \sum_i penalty & , \text{if not winner} \quad (0 \le penalty \le 1) \end{cases}$$

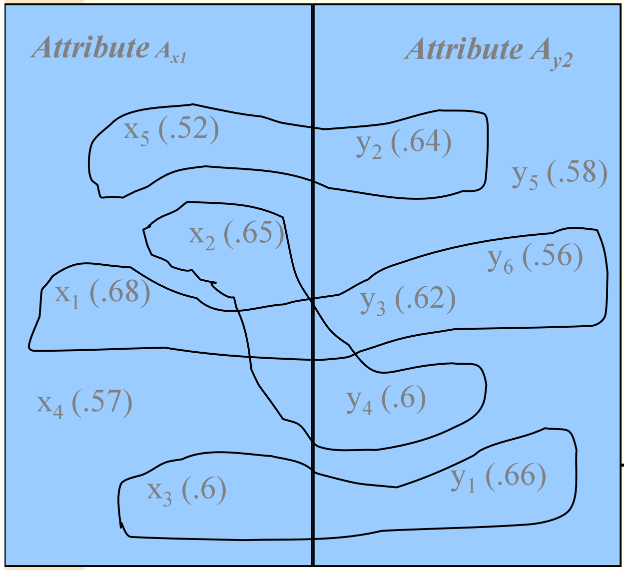
- ✧ Actions included in the exploration space are deduced according to the received qualitative feedback.
  - ✧state ≡ bid : $s = \langle v_1, v_2, \ldots, v_k \rangle$, v_x: value of attribute x
  - ✧action : $a = \langle a_1, a_2, \ldots, a_k \rangle$,$a_x \in$ {increase, decrease, mantain}

Q-Negotiation (App.2)

- Enterprise Agent's inter negotiation:
- Mutual dependencies resolution
  - Agents exchange between them alternative values for constrained attributes.
  - An alternative bid (*set of attributes' values*) has an utility, to that agent, that is lower than the previous one (decrement of the utility).
    - *Agents do not reveal the real utility value, but only the decrement of the utility value -> they keep that information private*
    - Final set of attributes' values that satisfy the constraints under negotiation is the one with the minimum joint decrement of the utility, that is, the minimum sum of the decrement of the utility of the negotiating agents.

Q-Negotiation (App.2)

✧ Mutual dependencies resolution



- E_1p: [(x_1,y_3),(x_1,y_6)] → du= [0,0]
- E_2e: [(x_1,y_3),(x_1,y_6)] → jdu= [.04,.1]
  E_2p: (y_1,x_3) → du=0
- E_1e: (y_1,x_3) → jdu= .08
  E_1p: (x_2,y_4) → du= .03
- E_2e: (x_2,y_4) → jdu= .09
  E_2p: (y_2,x_5) → du= .02
- E_1e: (y_2,x_5) → jdu= .18
  E_1p: (x_1,y_3) → jdu= .04
  E_2accepts: (x_1,y_3) → jdu= .04

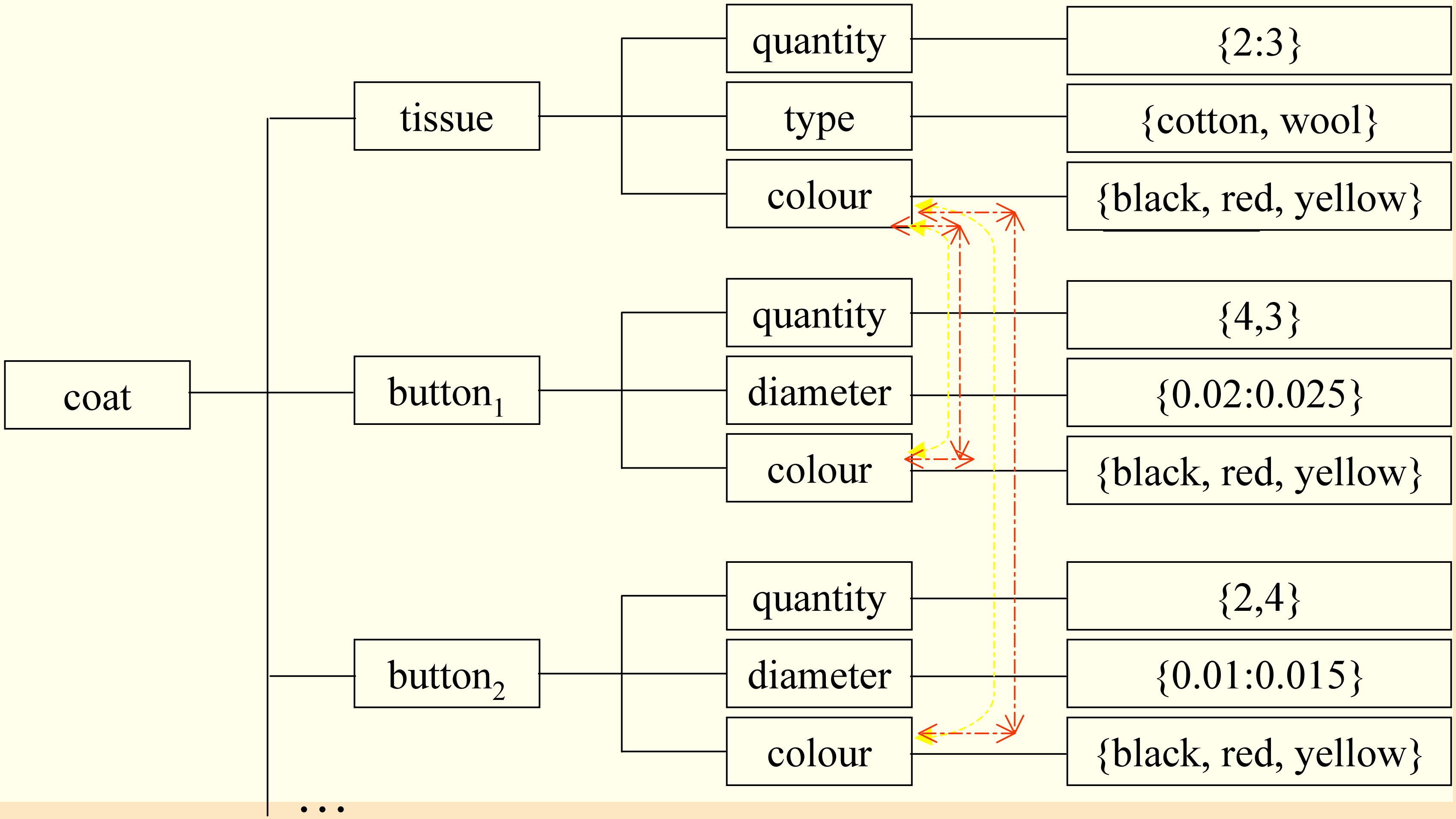
Q-Negotiation (App.2)

- Knowledge representation:

  - Ont = <Goods, Comp, Atr, Val, GCr, CAr, AVr, Deps>

  - Good, Comp, Atr, Val: set of goods', components', attributes' and values' identifiers.
    - GCr : $Good_i$ -> {Comp}, $\forall\ Good_i \in Good$
    relation that maps each good to a set of components.
    - CAr : $Comp_i$ -> {Atr}, $\forall\ Comp_i \in Comp$
    relation that maps each component to a set of attributes.
    - AVr : Atr_i -> Val_k, $\forall\ Atr_i \in Atr, \exists^1 Val_k \in Val$
    relation that maps to each attribute a value id.
    - Deps : $\{Dep_{ij}\}$, $Dep_{ij} = f(Val_{ki}, Val_{mj})$, $\forall\ Atr_i, Atr_j, \in Atr$
  dependencies between attribute' values.

Knowledge Representation



Knowledge representation (cont.)

- Ont = <Good, Comp, Atr, Val, GCr, CAr, Avr, Deps>
  - Good = {coat , car,... }
  - Comp = {tissue, button , wheel,...}
  - Atr = {quantity, colour, diameter, category,...}
  - Val = {{integer}, {string}, {real,continuous},...}

- GCr : coat → {tissue, $button_1$, $button_2$}
  - CAr : tissue → {quantity, colour, type}
    button → {quantity, colour, diameter}
  - AVr : quantity → {integer, discrete}
    colour → {string, discrete}
  - Deps : [(tissue, colour), (button,colour)] → []

Knowledge representation(cont.)

- *Market Agent* *(Announcement)*
-

<Good, Comps, Deps, Deadline>

coat,

{ (tissue,{quantity, type, colour},

{{2:3}, {cotton,wool}, {black,red,green}}),

(...), (...) },

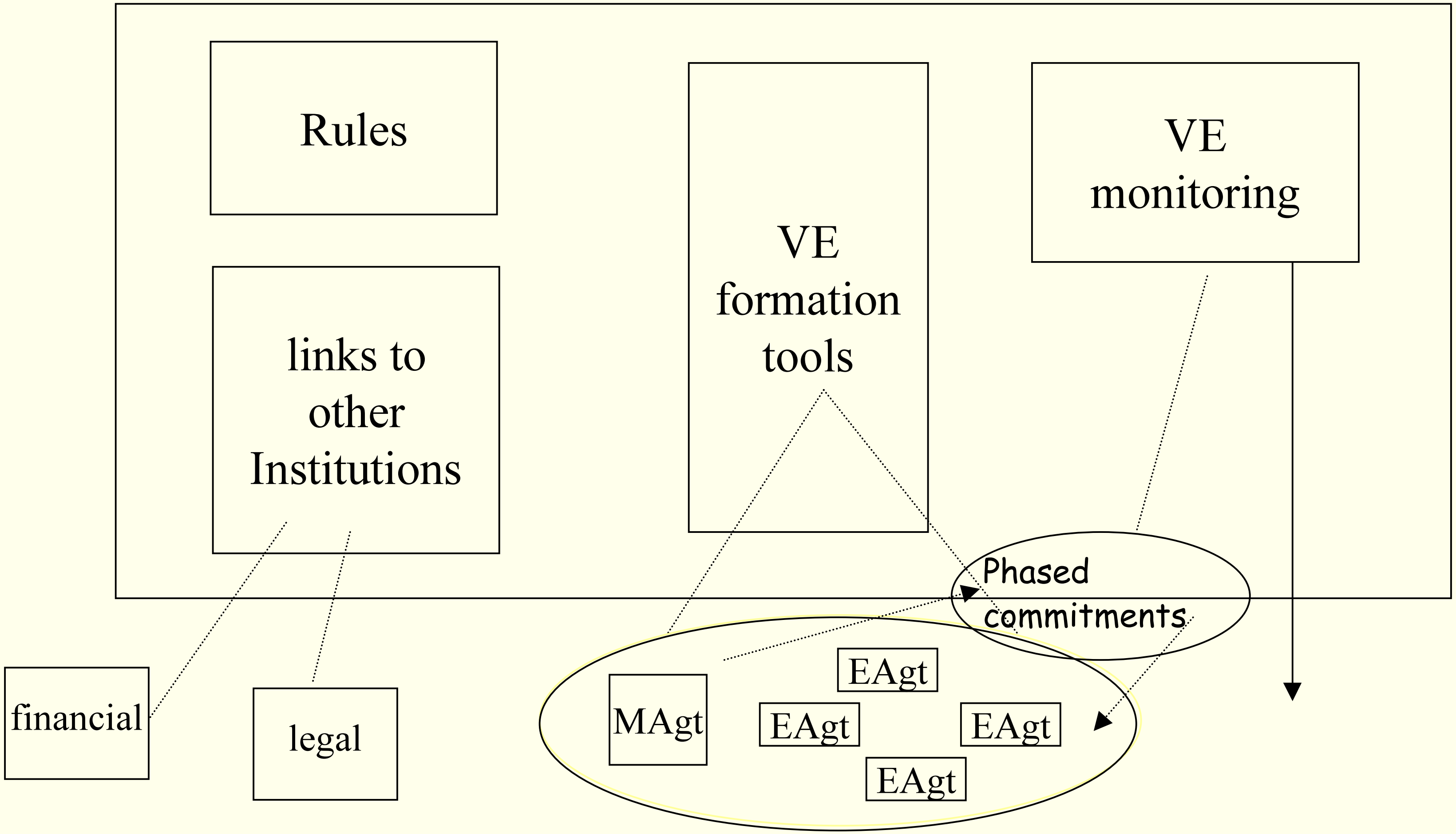
Dep_{ij} { ((tissue,colour), ($button_1$,colour),

[{{yellow},{yellow}},{{black,red},{black}}]),

(...) },

1234

Electronic Institution



Knowledge representation

- 6 Agents' Coalition formation
- The announcer selects from the 2^N-1 possible coalitions those that have the conditions to solve the problem (N=number of bidding agents) using some simple constraints to reduce the number of coalitions in the negotiation process:
  - rejection of all the coalitions that aren't powerful enough to solve the complete problem.
  - rejection of all the redundant coalitions (those that have any element that can be dropped keeping power enough to solve all the problem).
  - use of CPSel algorithm to reduce the number of possible coalitions.

MAS for Distributed Resources Management (App.3)

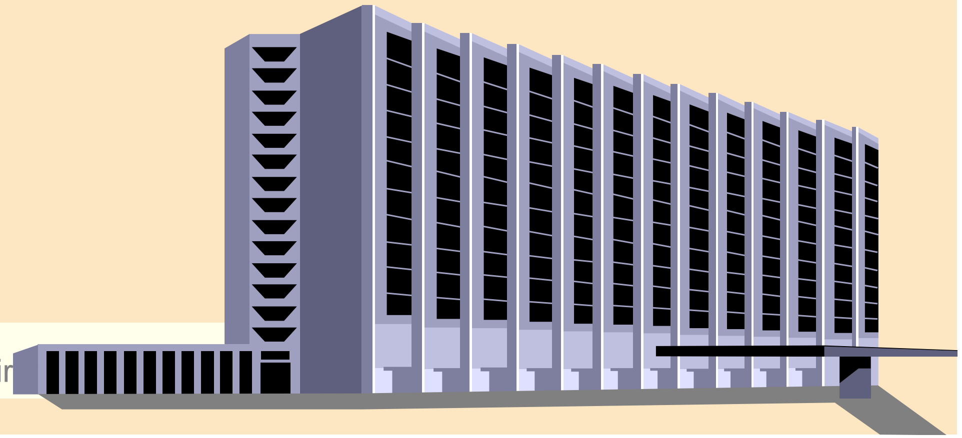
• MACIV

✧ **Management of Distributed resources in a Building Company:**

✧ The main objectives are:

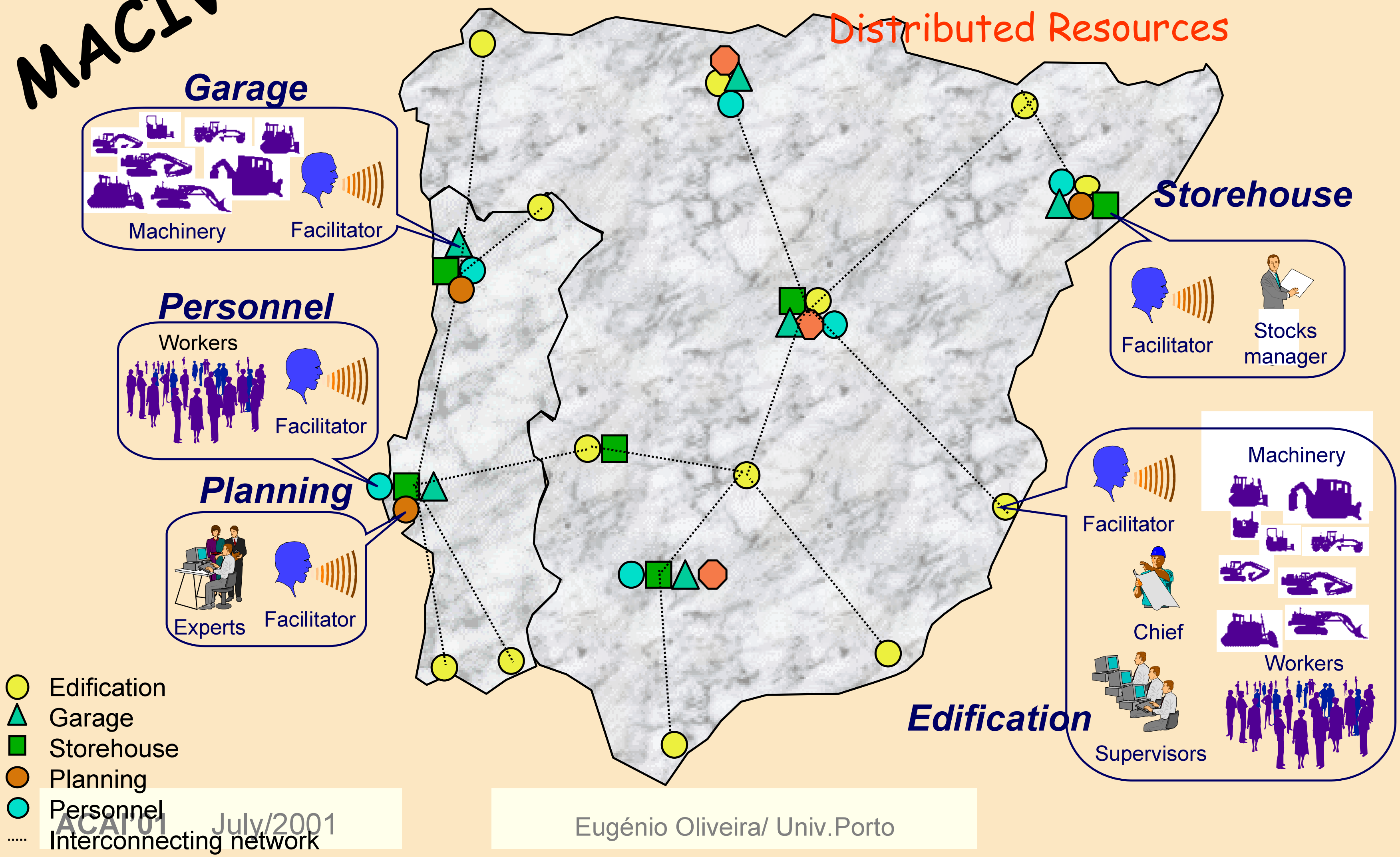
* To have access to knowledge about available resources
* To represent them as Agents that interact
•To make these Agents to Cooperate in a specific task

* To make Agents to negotiate and make coalitions to achieve the best possible price to execute the task



MAS for Distributed Resources Management (App.3)

- MACIV



MAS for Distributed Resources Management (App.3)

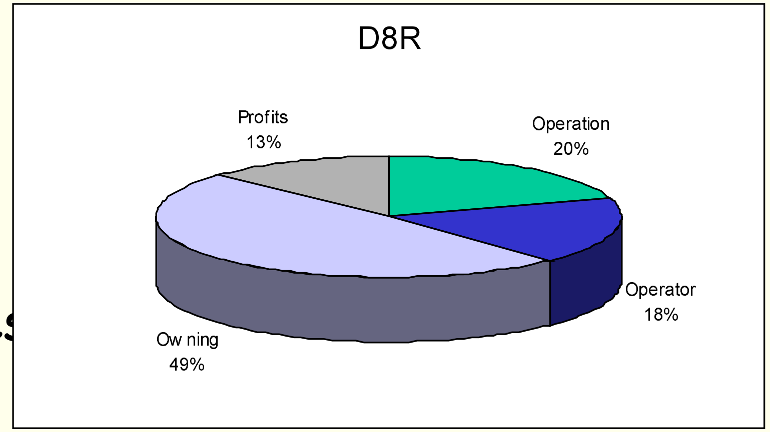
- MAS for Distributed Resources Management (App.3)
- Real-time resources management
  - Distributed application.
  - Correct proposal evaluation.
  - Agents utility evaluation.
  - Optimized planning.
  - Appropriated resource selection.
  - Inter-enterprises negotiation

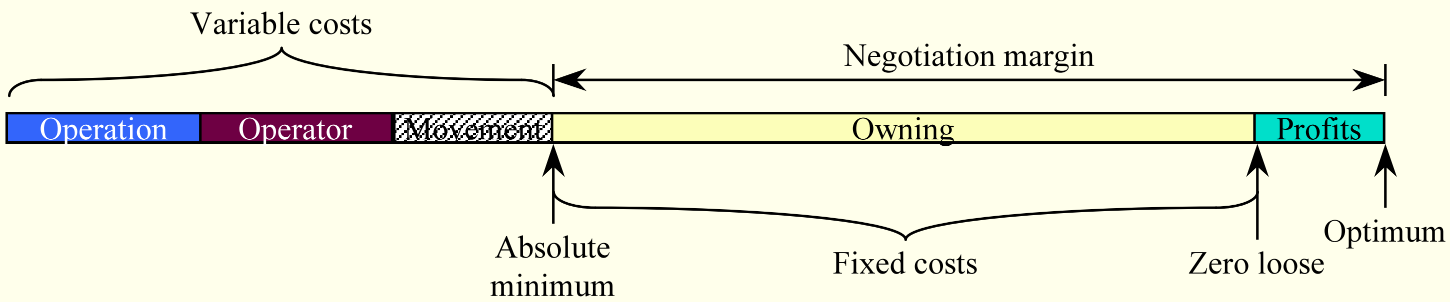
MAS for Distributed Resources Management (App.3)

MACIV

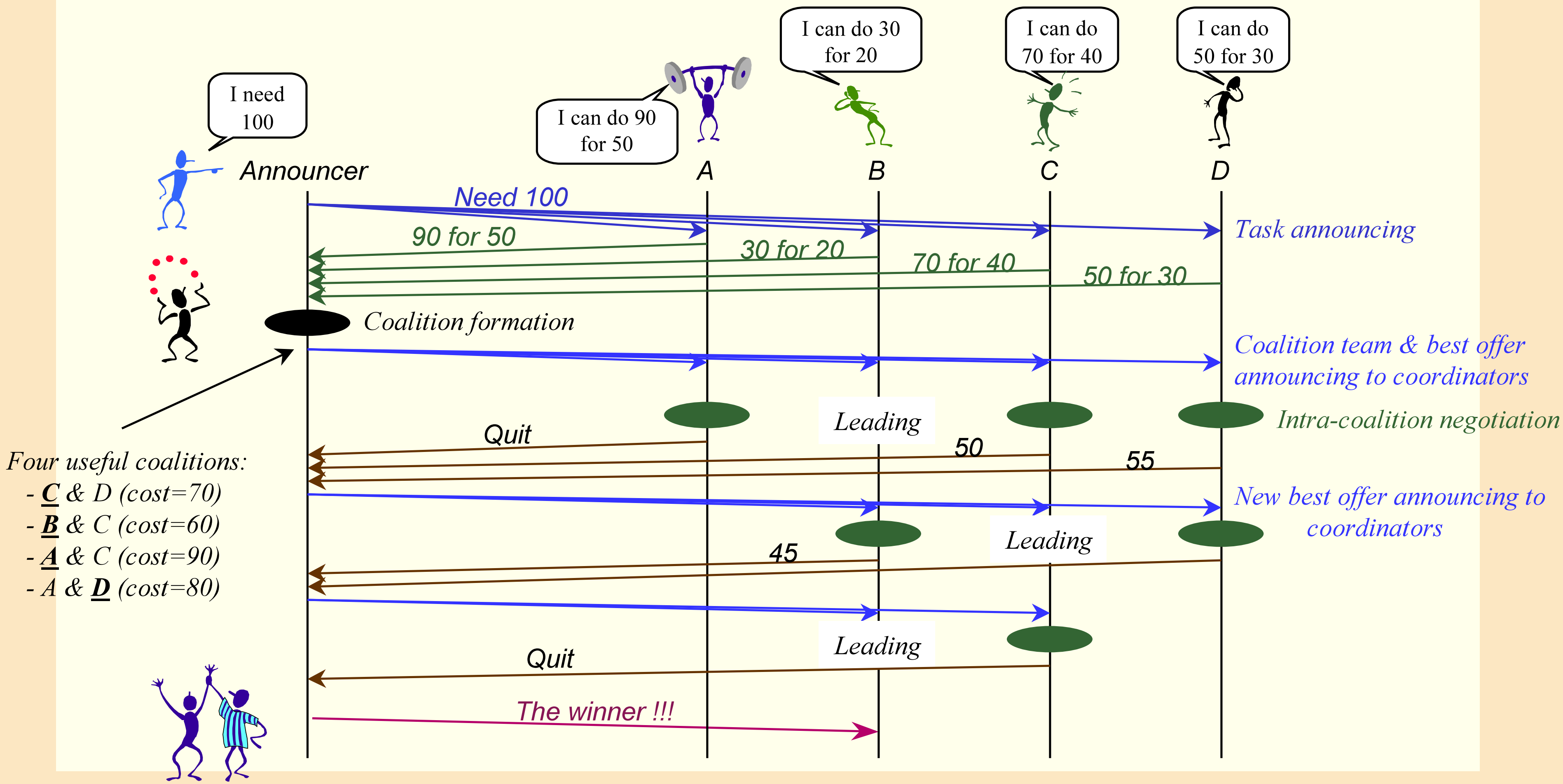
Agents activity cost calculation

- MAS for Distributed Resources Management (App.3)
- Five components:
  - Fixed costs: Owning cost
  - Variable costs:
    - Operation cost
    - Operator cost
    - Displacement (if neces
  - Profits

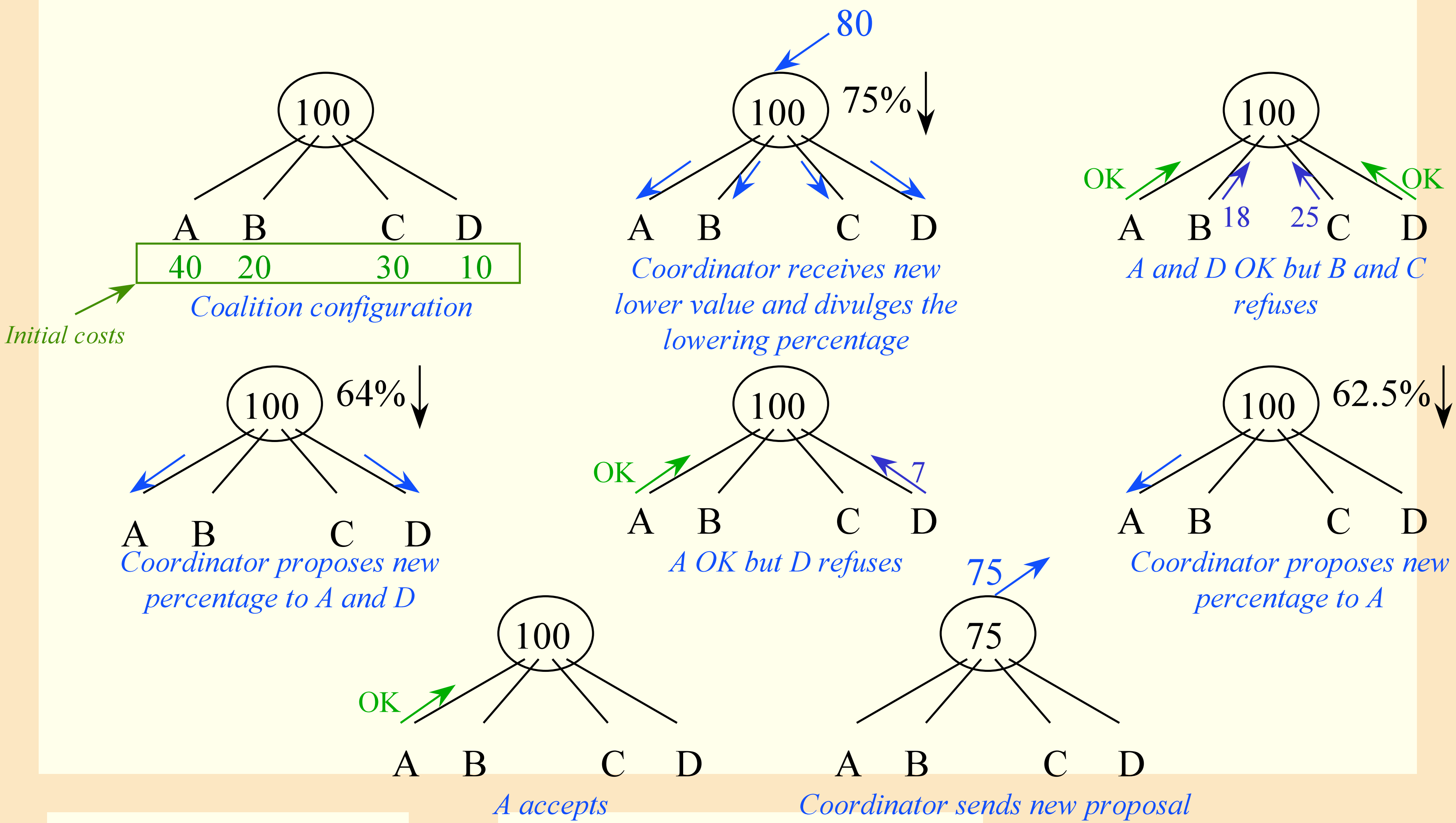




Inter-coalition negotiation

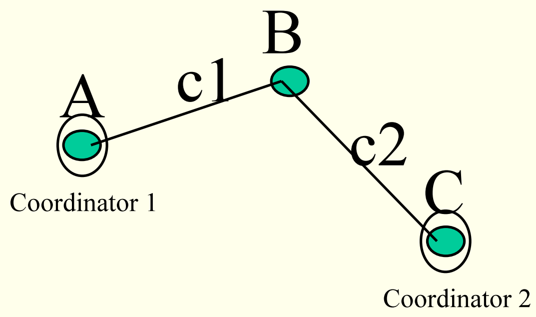


Intra-coalition negotiation



Intra-coalition negotiation

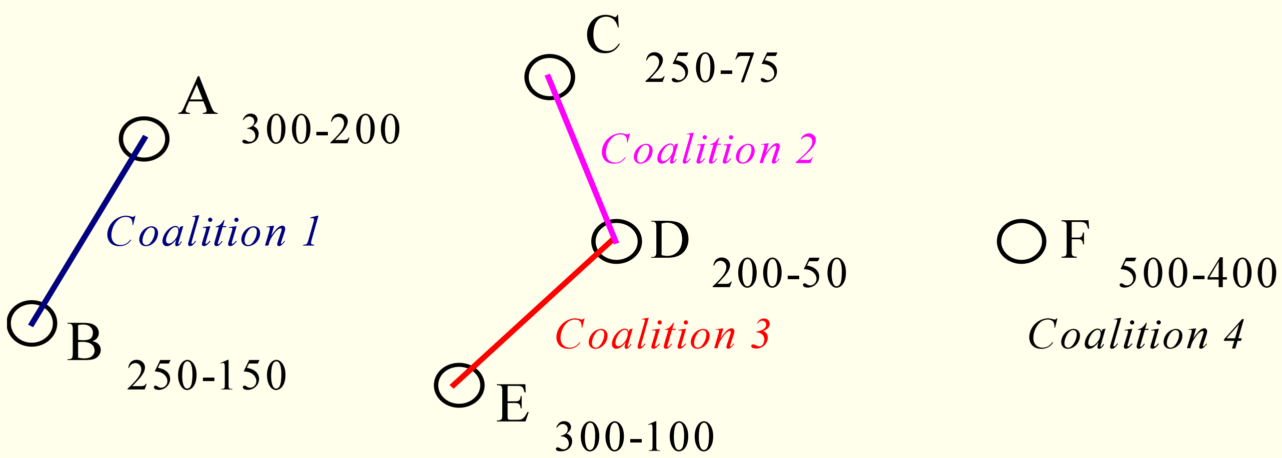
✧ The self-depreciation problem:



- When an agent participates in more than one competing coalitions, auto-depreciation should be avoided
- The - frozen costs solution has been adopted: the leading coalition participants freeze their own costs.

A negotiation example

✧



- 6 agents in 4 coalitions (D belongs to the leading coalition with C)
- X-Y : possible variation of agents' proposals
- Agent D does not contribute for improving coalition 3 proposal

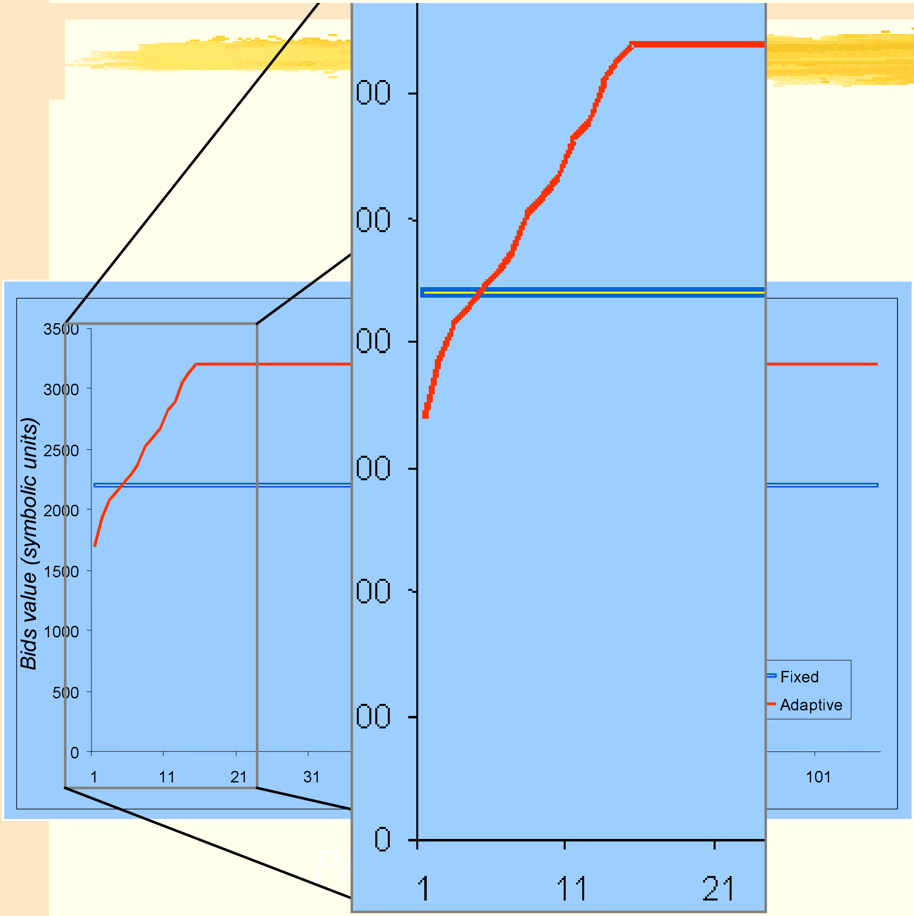
A negotiation example

✧ Price evolution along the negotiation process

Coalition	1	2	3	4	5	6
A,B	550	*400*	-	*Quit*		
C,D	*450*	-	*350*	-	*250*	-
D,E	500	400	350	*300*	-	*Quit*
F	500	400	*Quit*			
A	300	225				
B	250	175				
C	250	-	200		100	
D	200	200	150,175	150	150	
E	300	200	175	150		

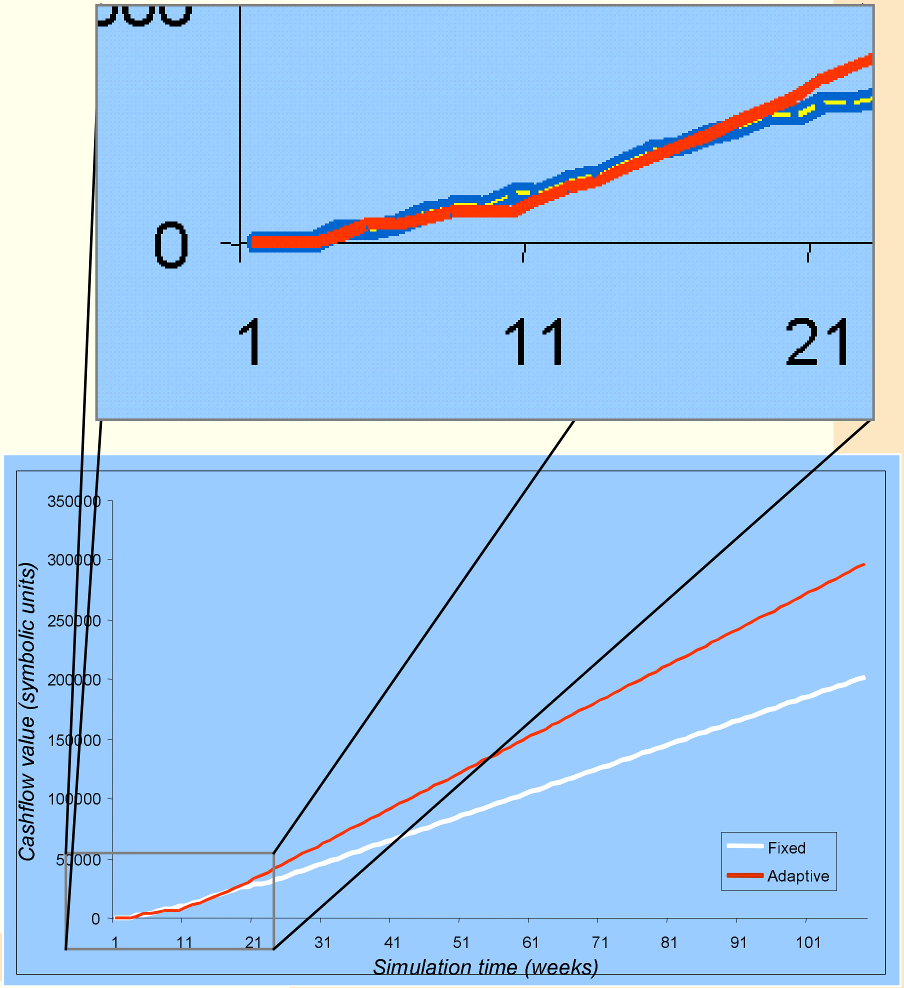
Coalition	Minimum value
A,B	350
C,D	125
D,E	150
F	400

Adaptive vs. fixed price (I)



Adaptive agent

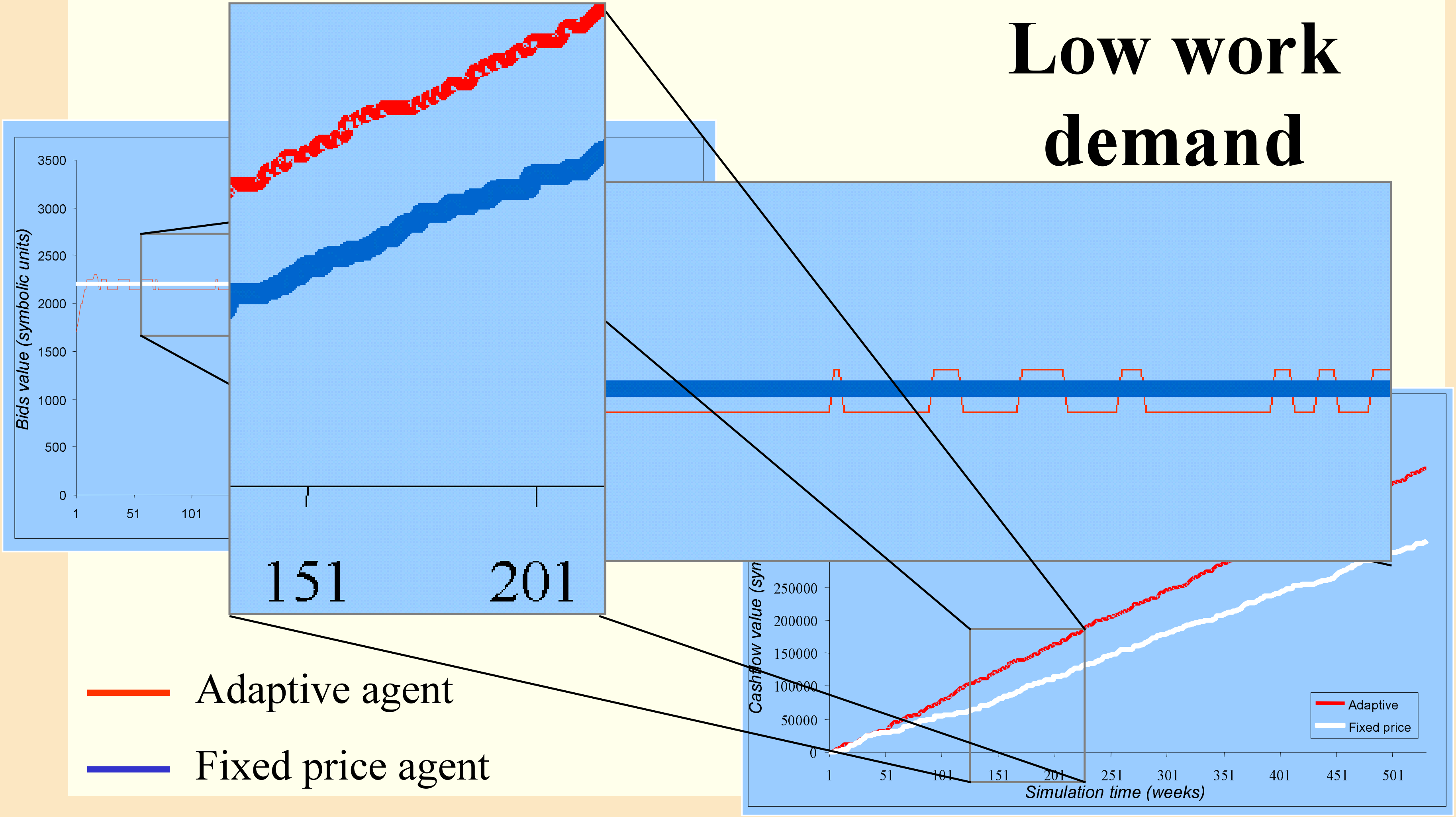
Fixed price agent



Cash flow

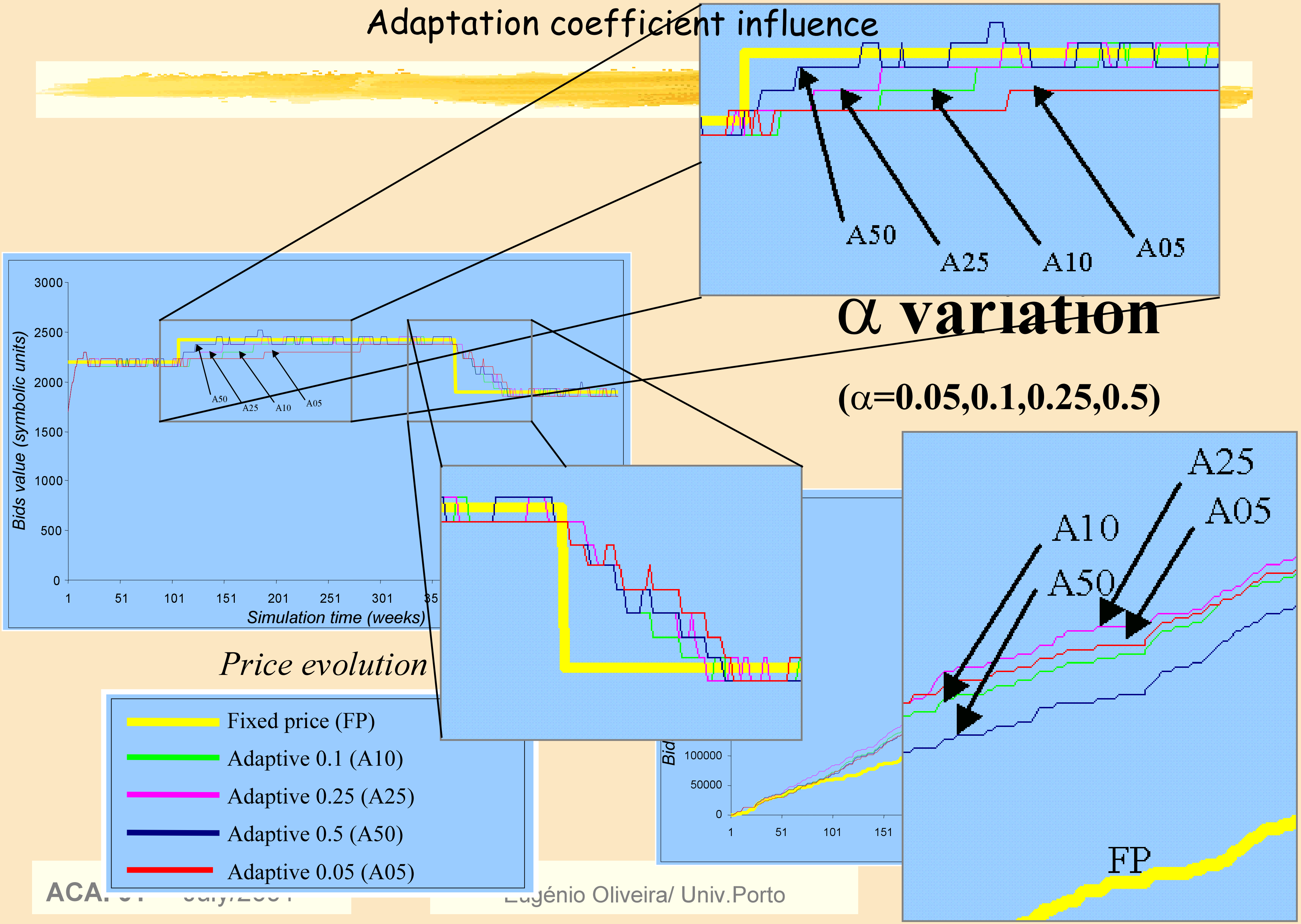
Adaptive vs. fixed price (II)

Low work demand



Cash flow

Adaptation coefficient influence

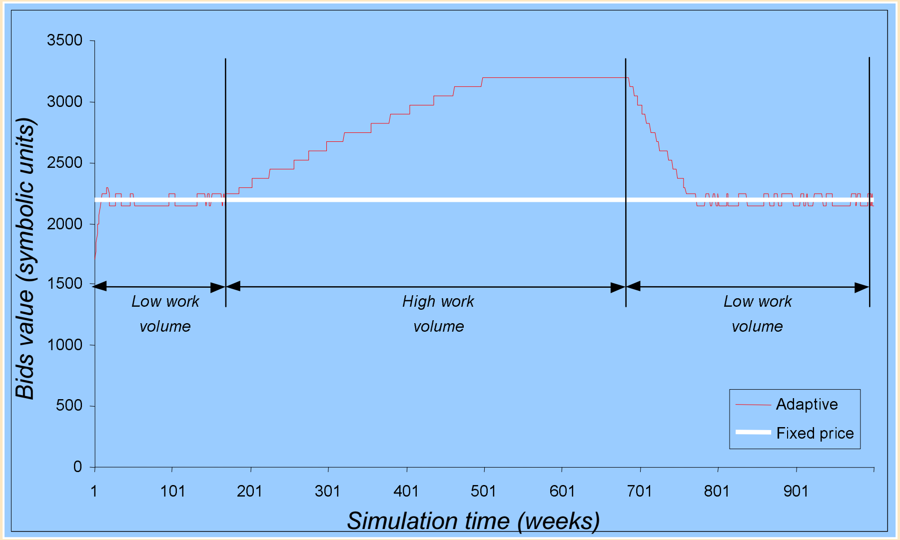


Price evolution

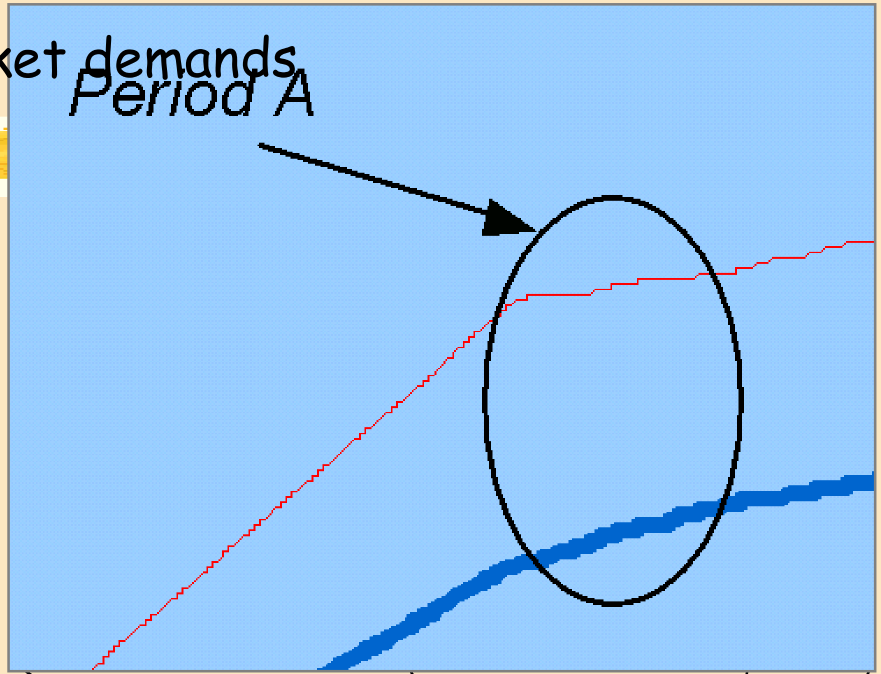
α variation

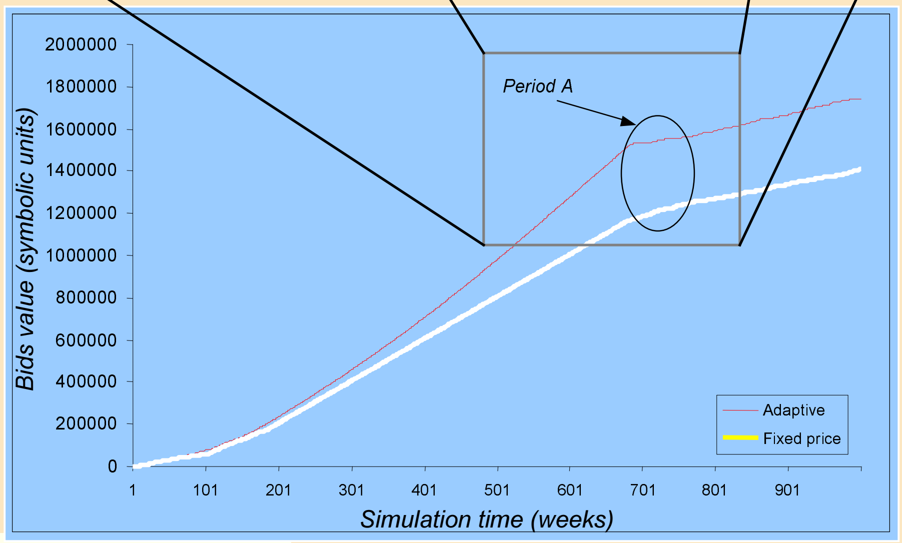
(α=0.05,0.1,0.25,0.5)

Influence of market demands



Price evolution





Cash flow

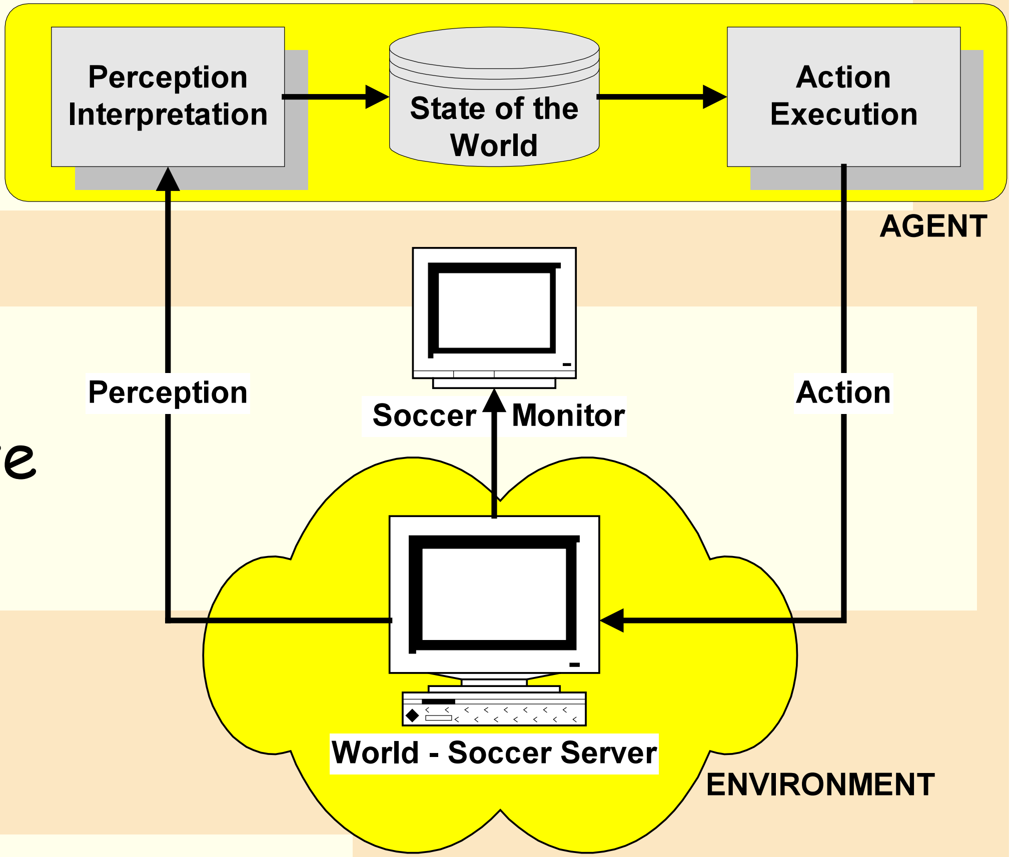
Adaptive agent

Fixed price agent

Agents' Team Coordination

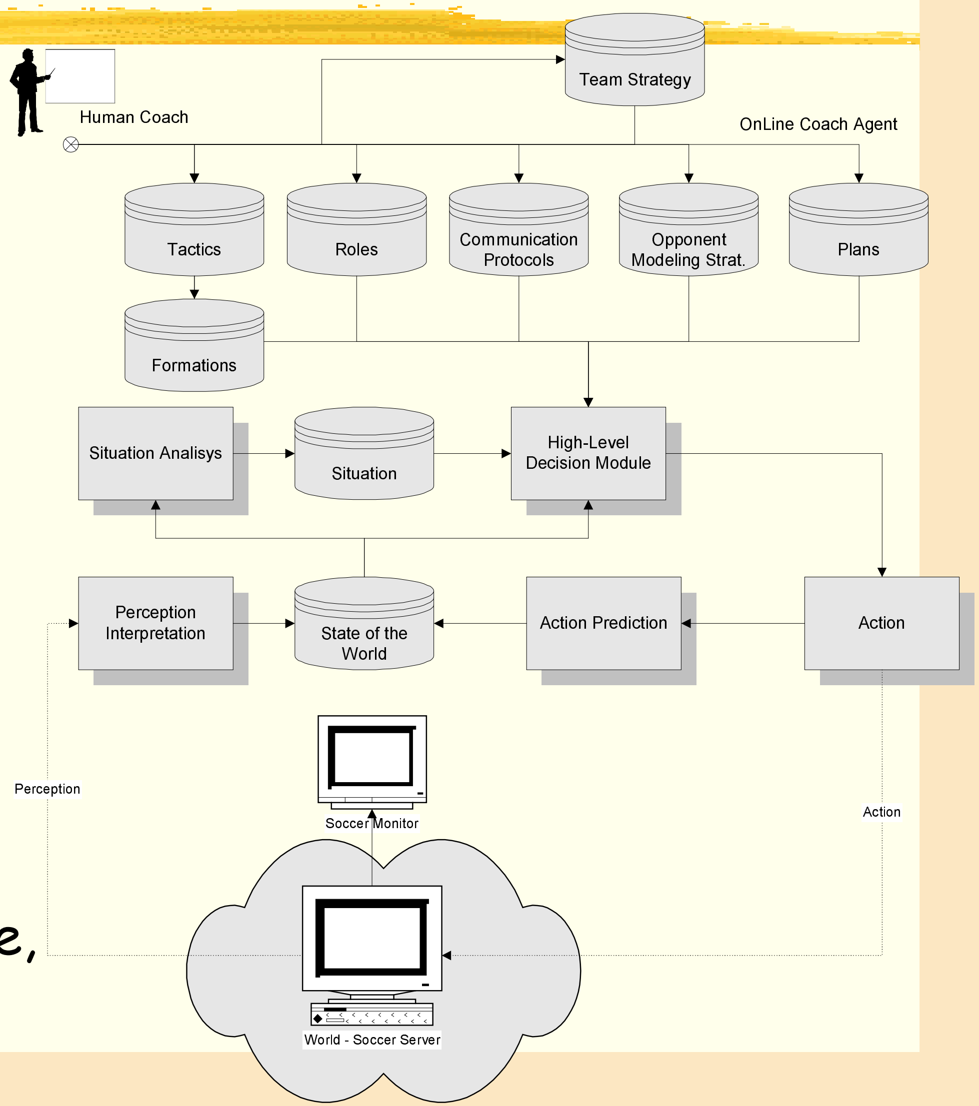
- 7 Coordinating a team of Agents
- How to Build Basic Agents for RoboSoccer?
  See Luis Reis Homepage: http://www.cerem.ufp.pt

  - Perception
  - World State Update
  - Action Execution



F.C.Portugal Project

✧ **Agent Architecture:**

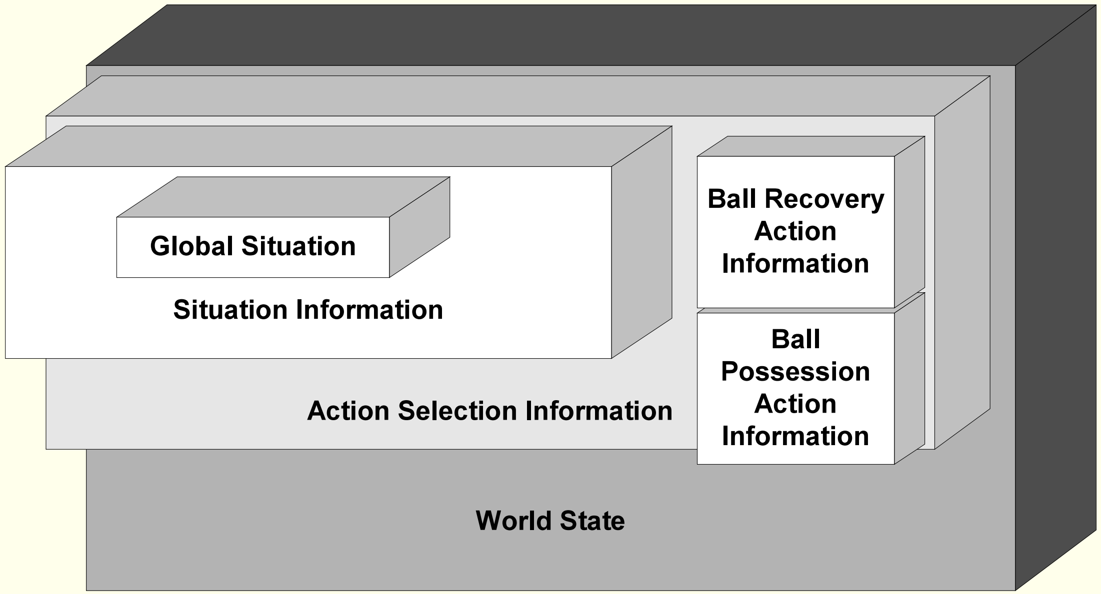


Improvements:

Multi-Level World State
STRATEGY, SBSP, DPRE, ADVCOM, SLM
High-Level Decision Module,
Optimum Kick

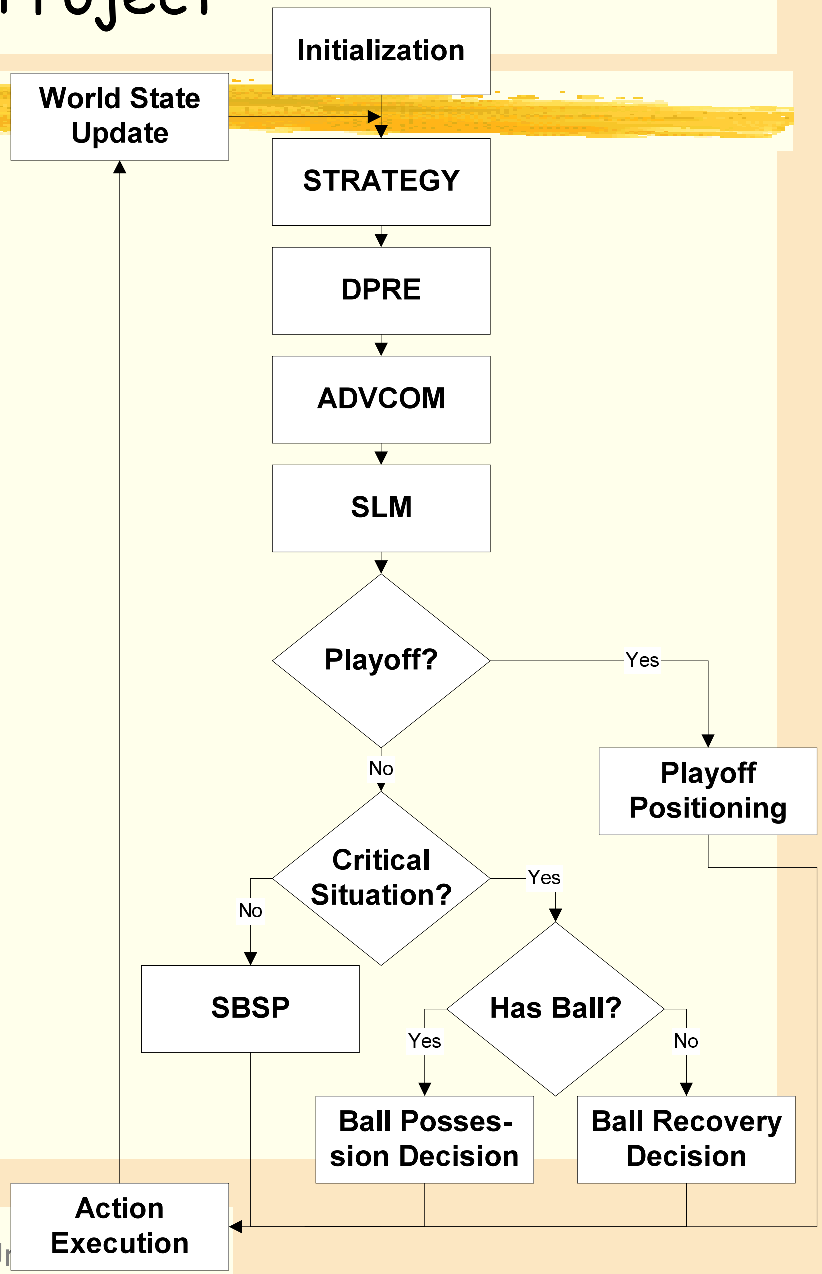
F.C.Portugal Project

- Multi-Resolution Information Structure
  - Update
    - Visual, Aural and Physical Information
    - Action Effect Prediction
    - Team-mate and Opponent Modelling



F.C.Portugal Project

✧ *Decision Control Flow*



F.C.Portugal Project

- Coordination of a Team of Homogeneous / Heterogeneous Agents in Adversarial Environments
  - SBSP - Situation Based Strategic Positioning
  - DPRE - Dynamic Positioning and Role Exchange

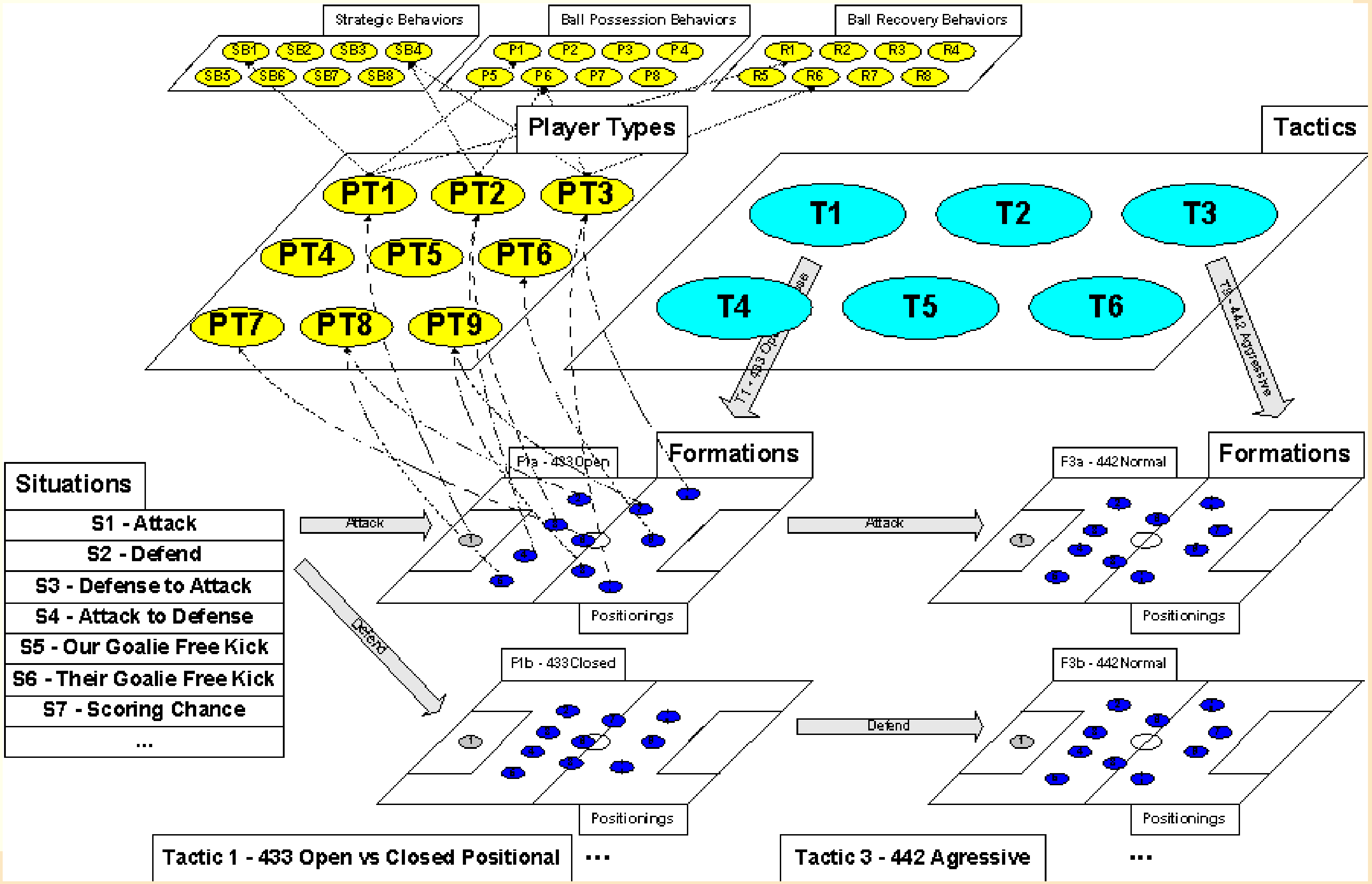
- STRATEGY Concept for Competing against other Team with Opposite Goals
- ADVCOM- Communication in Multi-Agent Systems
- SLM- Intelligent Perception : Strategic Looking Mechanism
- Optimisation Techniques – Optimum Kick, Smart Dribble
- Learning and Opponent Modelling in Adversarial Environments
- Relying on Soccer knowledge– Individual Decisions (Playoff, Ball Possession and Ball Recovery), Marking

F.C.Portugal Project
Team Strategy Formalisation

- **Team Strategy**
  Tactics composed by Formations used in situations
  (defend, attack, defence->attack, ...)
  Player Types
  (Strategic, Ball Possession and Ball Recovery Characteristics)
  Communication Protocols, Perception Strategies,
  Team-mate and Opponent Modelling Techniques (not yet)

- **Formations**
  Dynamic Positioning of each player
  Each formation positioning has a given Player Type

F.C.Portugal Project



✧ Formations and Player Types

✧ Strategic Situation: SBSP – Strategic Positioning

✧ Active Situation (with/without Ball): Active Behaviour

- Ball Possession Actions:
  - Shoot to the Goal
  - Pass (rapidly) the Ball
  - Forward the Ball (to a given point)
  - Dribble with the Ball
  - Hold the Ball
- Decision Matrixes used to Select the Best Action!

✧ DPRE - Dynamic Positioning and Role Exchange

- Dynamic exchange of positioning and behaviours (player types) based on utility:
  - Distances from players positions to the positioning strategic positions
  - Positioning importance and adequacy of agents to that positioning
- Since Agents are homogeneous, DPRE improves the team collective performance
- Energy levels and Field covering are increased
- Enables the team to play well with 5 or 6 players!
- Dynamic Covering!

Conclusions

- 8 Conclusions and Future Research
- ***Agent´s features for Negotiation***:
  - *Multi-issue criteria evaluation including qualitative feedback*
  - *Q-Negotiation through the use of both:*
    - *Tactical and*
    - *Strategic Reasoning based on Q-learning algorithm*
    - *Mutual Dependencies resolution*
  - *Application Examples:*
    - *SMACE- Multi-lateral CDA Electronic Market*
    - *MACIV- MAS for Distributed Resources Management*

Conclusions

- *Simple Protocol for Agents' coalition Selection*
- *Electronic Institutions to encompass Virtual Enterprise Life-Cycle:*
  - *Virtual Enterprise Formation*

- *Knowledge intensive use for Agents' coordination in adversarial environments*
- *Dynamic decision based on rich World state evaluation*

Future Research

- ***Electronic Institutions:***
  - *Phased-Commitments, Decommitments and penalties*
  - *Knowledge representation:*
    - *Ontology-Services: Ontologies composition, fusion (and translation)*
    - *Enhancing KR in the Ontology (by including mutual constraints)*
    - *Ontology editor and instantiations for specific domain (textile, car comp.)*
  - *New Learning approaches:*
    - *K-Level Agents?*
    - *Using Inductive Learning (by examples) for improving Tactics*
  - *Others:*
    - *Agents' coordination On-line learning (FCPortugal Robosoccer team)*
    - *Learning User's Preferences for Electronic Commerce*
    - *Emotional Agents in Multi-Media, Distributed Truth maintenance ...*

Agent technology for tomorrow ?