Service Management in Future Networks: The C3SEM Vision.

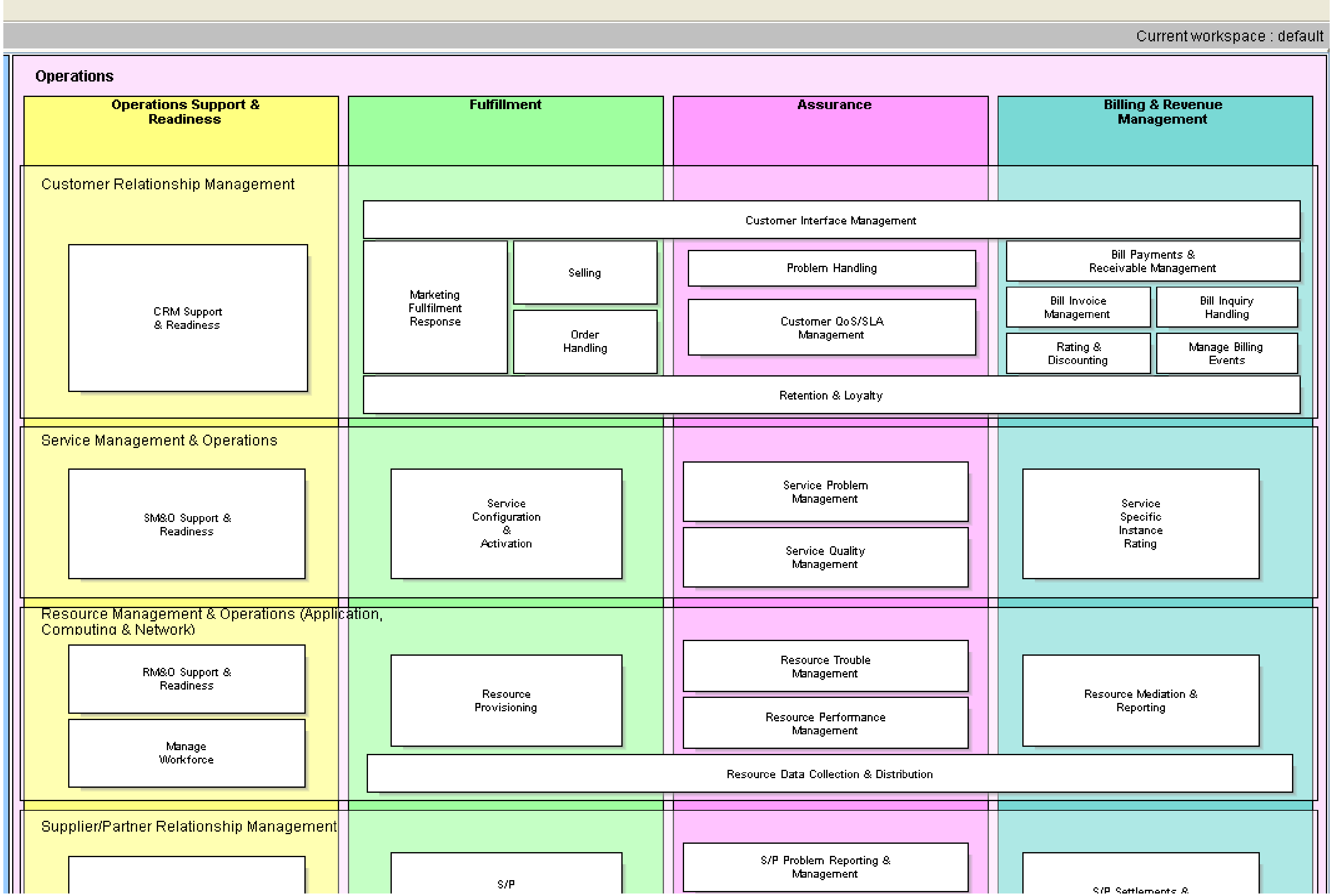
Joan Serrat
Universitat Politècnica de Catalunya

Talk outline

- Service Level Management (SLM)
- Highlights of the C3SEM Project
- SLM issues addressed in C3SEM
- Pricing as a means to enforce SLM
  - Background
  - Our pricing approach
  - Evaluation scenario
  - Results
  - Next steps

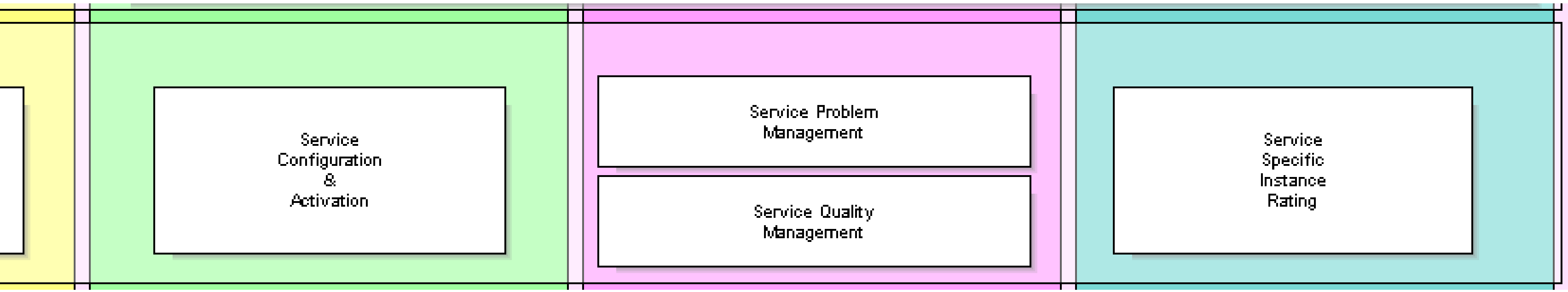
Service Level Management

- SLM under the eTOM framework[1] perspective (1)



(1) eTOM 7.5 Portal http://www.tmforum.org/BusinessProcessFramework/6775/home.html

SLM under the eTOM framework perspective(2)



- Definition:
  - "…focusses on the knowledge of the services (Access, Connectivity, Content, etc.) and includes all functionalities necessary for the management and operations of communications and information services required by or proposed to customers"
  - "The processes in this horizontal functional grouping are accountable to meet, at a minimum, targets set for Service Quality … as well as Service Cost"

The C3SEM project (1)

- Cognitive, Cooperative Communications and autonomous SErvice Management (C3SEM)
- 3 years duration starting January 2010
- Funded by MICINN
- Structure
  - *Subproject* 1(TEC2009-14598-C02-01). Connectivity and network planes. This subproject is executed by the University of Cantabria.
  - *Subproject 2* (TEC2009-14598-C02-02). **Autonomous Management** of **New Generation Services**. This subproject is executed by the Technical University of Catalonia

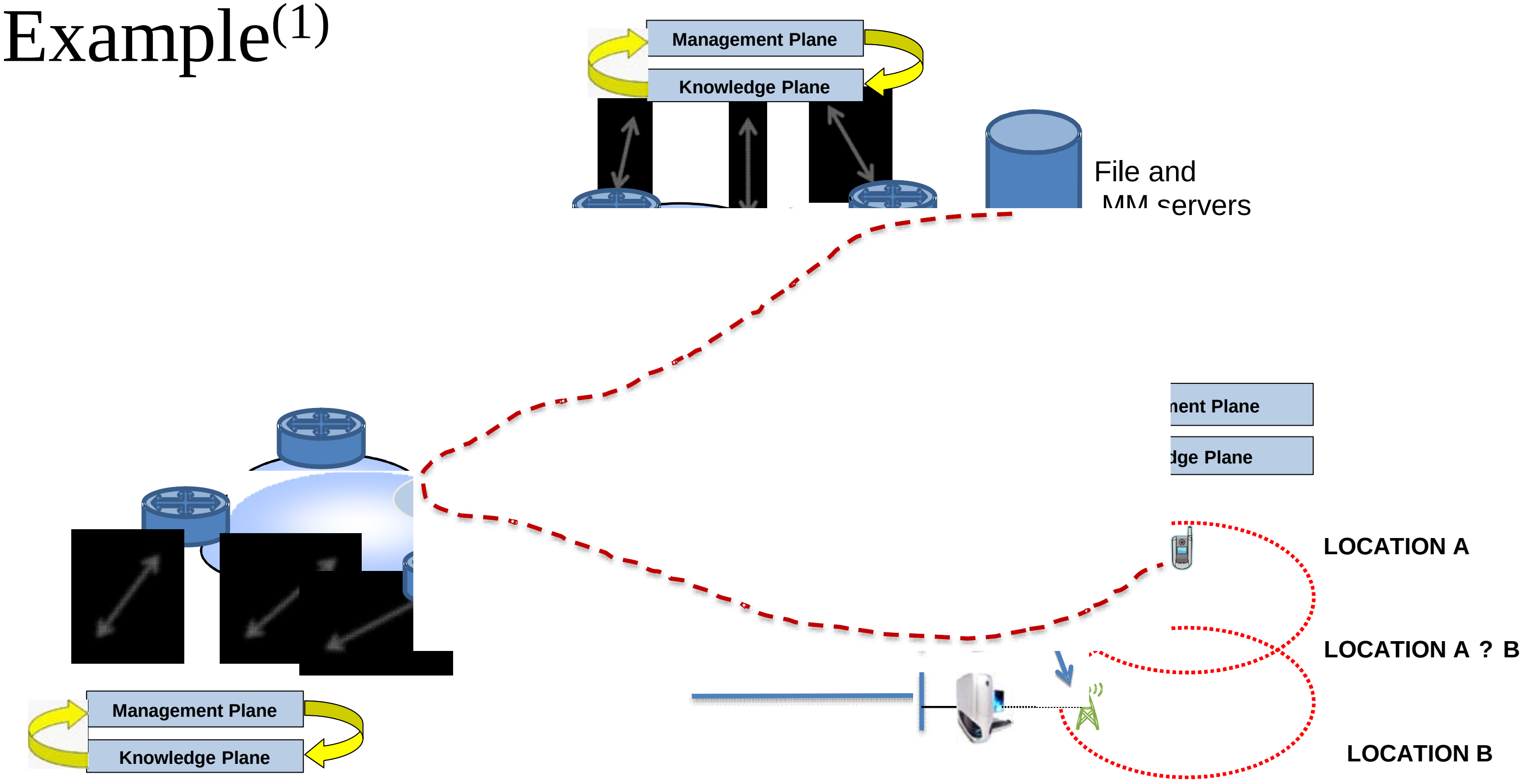
The C3SEM Project (2)

- Autonomous Management
  - Self-configuration
    must configure themselves in accordance with high-level policies representing service agreements, or business objectives.
  - Self-optimization
    must seek to improve their operation every time.
  - Self-healing
    Must detect, diagnose and repair problems caused by network or system failures.
  - Self-protection
    Must defend themselves as a whole reacting to or anticipating problems arising from malicious attacks (e.g. DoS) or cascading failures that remain uncorrected by self-healing measures.

- An interesting related paper on services autonomous management is by Yu Cheng et al. Toward an Autonomic Service Management Frammework: A Holistic Vision of SOA, AON and Autonomic Computing. IEEE Communications Magazine, May 2008

The C3SEM Project (3)

- New Generation Services
  - Seamless mobility, Resources Heterogeneity, Lots of different services, …
- Example[1]



(1) Deliverable D6.3 Final Results of the Autonomic Internet Approach http://ist-autoi.eu/autoi/d/autoi_deliverable_d6.3_-_final_results_of_the_autonomic_internet_approach.pdf

The C3SEM Project (4)

- How the C3SEM project contributes to cope with the challenges of Management of NGSs
  - Policy shaping mechanisms to manage the access network to satisfy a given set of business objectives(1)
  - Policy shaping mechanisms to create virtual communications infrastructure to satisfy a given set of business objectives(1)
  - Learning mechanisms for self-evolving management policies(2)
  - How to confer stability in systems driven by multiple control loops
  - **Efficient Dynamic Pricing Algorithms**

(1) J.Rubio-Loyola, et al. Business-driven Management of Differentiated Services 2010 IEEE/IFIP Network Operations and Management Symposium - NOMS 2010, pp.240-247
(2) R.Bagnasco, J.Serrat Multi-Agent Reinforcement Learning in Network Management 3rd International Conference on Autonomous Infrastructure, Management and Security (AIMS 2009) PhD paper

The Problem of Pricing Assignment

- How to charge for service in such a complex scenario
- How to negotiate with clients
- How to distribute pricing information
- How to make profit
- How to deliver quality of service
- How to balance/optimize resource utilization

Why Pricing is so much important

- Pricing has to be seen as a means by which the service provider communicates with users and give them incentives to make an efficient use of the network(1,2)

- It contributes to give value to services and to reach stability and robustness

(1) P. Koutsakis, Call Admission Control for wireless videoconference traffic based on the users' willingness to pay, EW 2010, April 2010
(2) J. Araujo et al. Towards cost-aware multipath routing, Scalabilty of Networks and Services, LNCS 5637, 2009

Pricing Strategies[1]

- Common telecom strategies
  - Fix price per time or traffic
  - Flat rate
- Auction
- Dynamic/Static
- Global/Local
- Distributed/Centralized

(1) D.Niyato et al. Competitive Pricing in Heterogeneous Wireless Access Networks: Issues and Approaches. IEEE Network, November/December 2008

Current Pricing Proposal at C3SEM

- Aimed characteristics
  - Decentralized
  - Rule-based
  - Dynamic
  - Driving principles
    - Profit maximization
    - Load balancing
    - QoS assurance

Intuitive rules about demand

- ***if*** *few_users* ***and*** *users_decreasing_slow* ***then*** *decrease_price_slow*

- ***if*** *lots_users* ***and*** *users_increasing_fast* ***then*** *increase_price_fast*

Intuitive rules about competition

- ***if*** *competitor_price_lower* ***and*** *competitor_price_decreasing_slow* ***then*** *decrease_price_slow*

- ***if*** *competitor_price_higher* ***and*** *competitor_price_increasing_fast* ***then*** *increase_price_fast*

Design challenges

- Rules may conflict themselves. Therefore a *conflict resolution mechanism* is needed
  - Example:
    - ***if*** *few_users* ***then*** *decrease_price_slow*
    - ***if*** *competitor_price_higher* ***then*** *increase_price_fast*

- The adverbs “few”, “lot”, “slow”, “fast”, etc. are generic

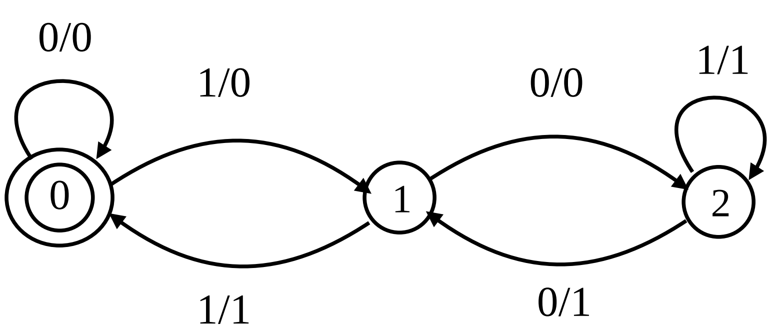
Conflict resolution mechanims

- Common in any multi-goal optimization system
- Solvable by:
  - Human intervention
  - Explicit priorities (on rules or their parts)
  - Heuristics
  - Machine learning
  - ….

Our approach is a different one based on Finite State Transducers

Finite State Transducers

- They are a finite state machine with output.
- On each edge, there are two labels.
- The left label match the input and the right label defines the output.
- Transducers can be seen as devices defining a class of relations over strings of symbols.



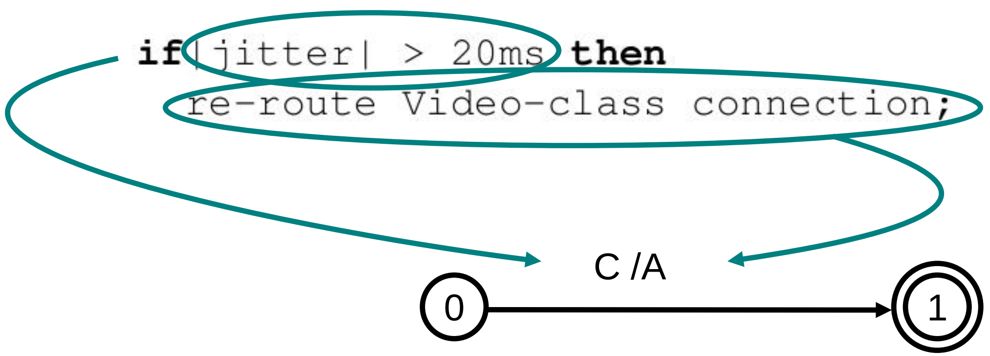
A symbol transducer representing division by 3

11 -> 01
110 -> 010

FST Features

- Extensive theory.
- Their implementations are light-weight and usually show good performance.
- They may have predicates instead of simple symbols as labels.
- Useful operations
  - Union, intersection, complement, composition, Kleene closure, determinisation, minimization.

Policies and Finite State Transducers[1]

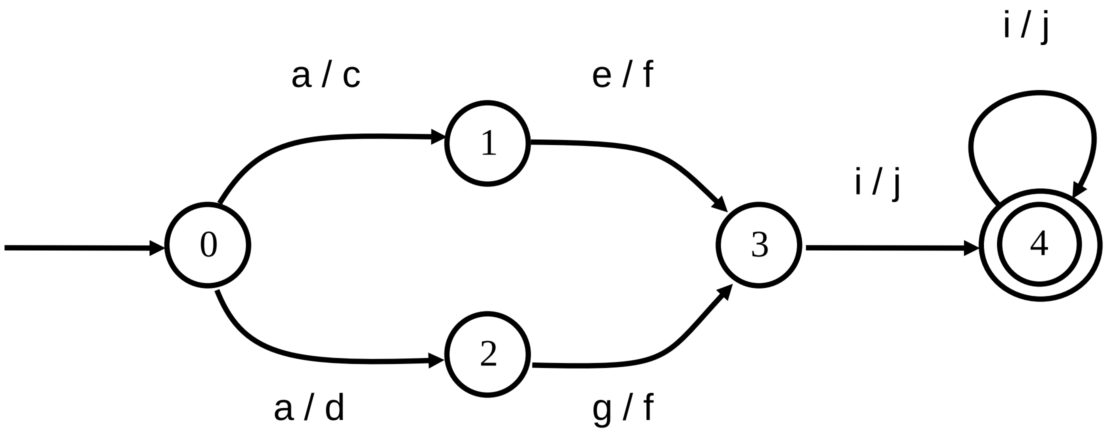


Translation From *if-then* to FSTs

- They are modeled as a graph with one edge for each event in the condition
- Elementary transducers for obligations/prohibitions, rights/dispensations and constraints
- Elementary FSTs representing simple rules can be combined to represent systems driven by sets of rules

(1) J.Baliosian, J.Serrat FINITE STATE TRANSDUCERS FOR POLICY EVALUATION AND CONFLICT RESOLUTION 5th IEEE International Workshop on Policies for Distributed Systems and Networks (Policy-04), pp.250-259 IBM T.J. Watson Research Centre, Yorktown Heights, New York June 07-09, 2004

A key FST operation: Determinisation

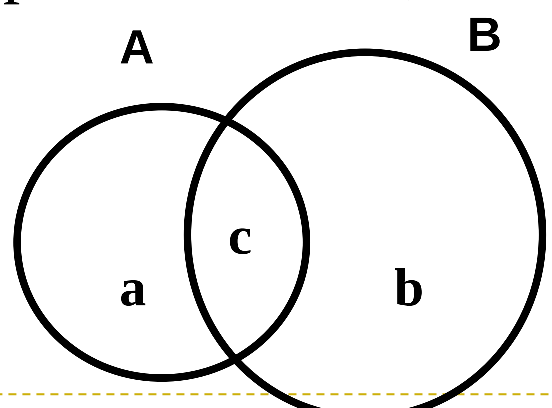


Ambiguous transducer

- Determinisation is the actual conflict resolution process
- Its aim is to have only one possible edge to choose in a given node
- *Predicates* are not enough: they were complemented by ***Tautness Functions***

Tautness Functions

- Are intended to represent how taut or tight a condition is around an event.
- Related to the concepts of “distance” and “nested domains”. Similar to *membership value* in fuzzy sets[1]
- It assigns a *real in* [-1,1] to the duple <condition,event>



$$\tau_A(a) > 0$$
$$\tau_A(b) < 0$$
$$\tau_A(c) < \tau_B(c)$$
$$\tau_{A\vee B} = max(\tau_A, \tau_B)$$
$$\tau_{A\wedge B} = min(\tau_A, \tau_B)$$
$$\tau_{\neg A} = -\tau_A$$

$$\tau_{A\rightarrow_\tau B} = \begin{cases} \tau_A, & if\ \tau_A < \tau_B \\ -1, & else \end{cases}$$

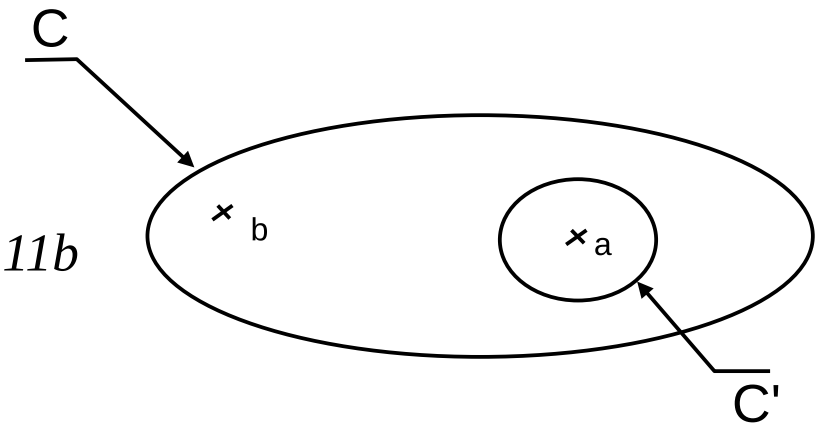
$$\tau_{A\rightleftarrows_\tau B} = \begin{cases} \tau_A, & if\ \tau_A = \tau_B \\ -1, & else \end{cases}$$

(1) N.Baldo et al. Fuzzy Logic for Cross-Layer Optimization in Cognitive Radio Networks. IEEE Communications Magazine, April 2008

Examples of TFs

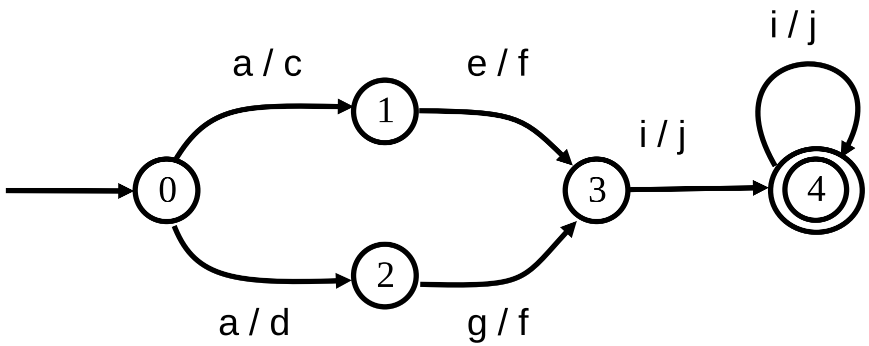
- The most straightforward example is when a domain is “inside” another domain. ***C'*** is tauter than ***C*** on the event **a**.

Condition C:
A terminal connected with a wireless interface
Condition C':
A terminal connected with a IEEE 802.11b interface

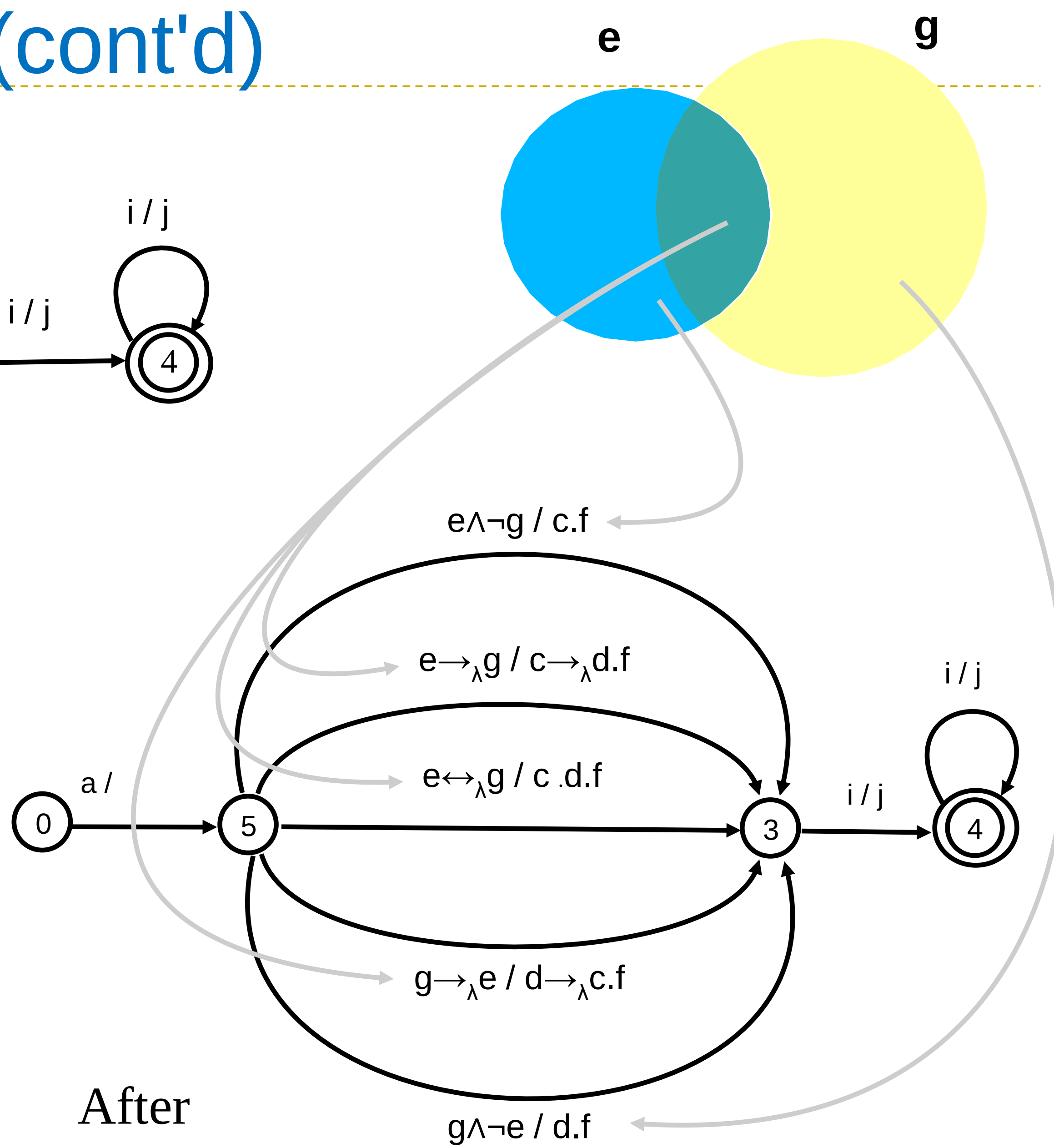


$\tau_{C'}(a) = 0.8$
$\tau_{C}(a) = 0.4$
$\tau_{C'}(b) = -0.7$
$\tau_{C}(b) = 0.3$

Determinisation (cont'd)



Before



After

The Overall Process

1. Translate rules to transducers
2. Compute the union of all transducers representing rights and obligations
3. Subtract the transducers representing prohibitions and dispensations
4. Compose the resulting transducer with each constraint transducer
5. Determinise the resulting transducer to solve conflicts

Summary on the TFFST Model

- A formal model based on a new entity called TFFST was developed for conflict detection and resolution of **static** and **dynamic conflicts**.
- This model should work **efficiently** and **independently of technology**.
- The most costly processes for conflict resolution are carried **off-line**.
- The **runtime** process is of **lineal order** on the amount of incoming events.

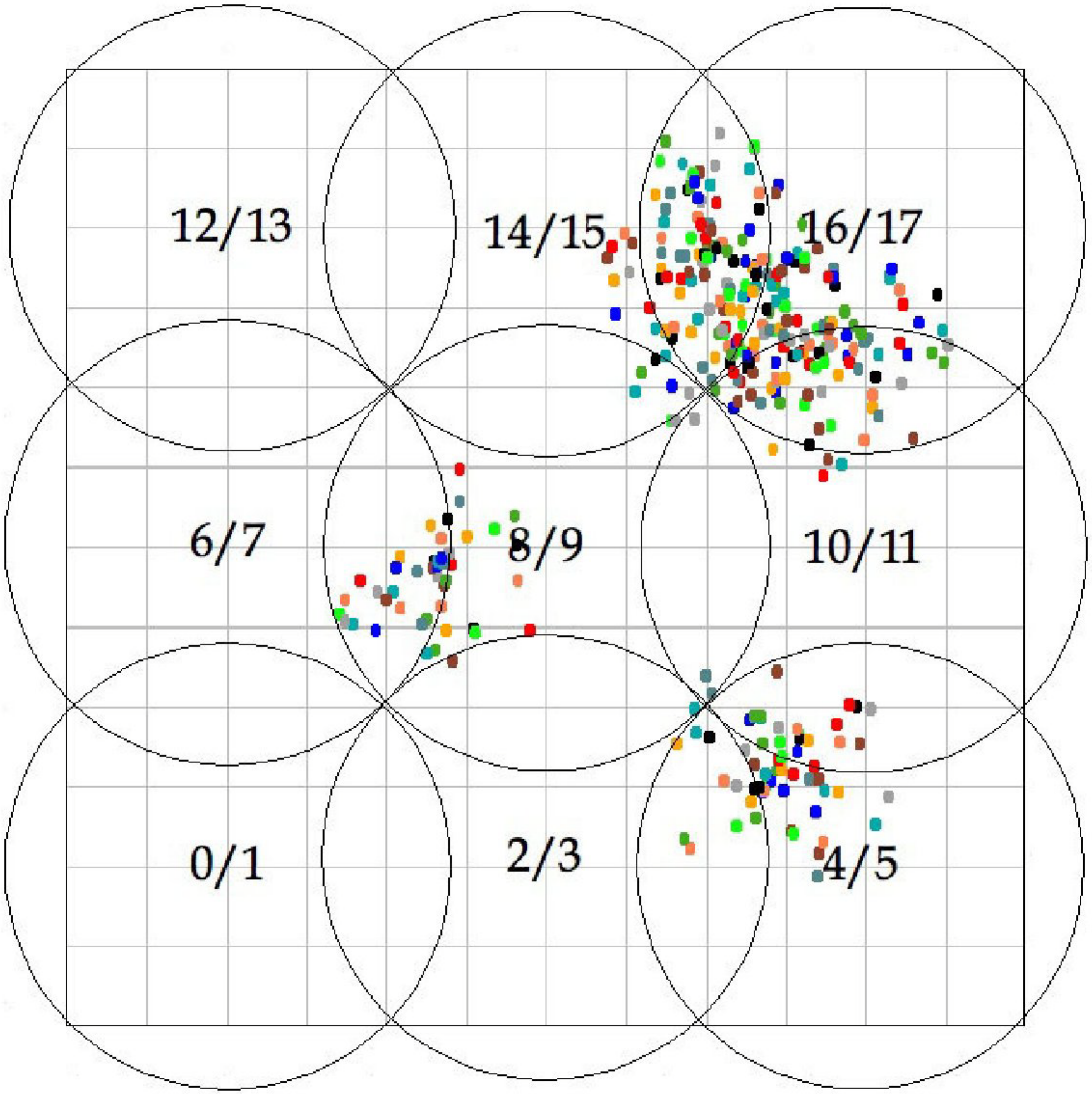
Evaluation scenario

- Two ISP competing
- Deployed over the same urban zone
- With identical network resources
- One ISP uses the TFFST-rule-based solution (our solution)
- The other ISP uses a static flat rate pricing scheme
- Users are aware of the price of each ISP and decide to connect with the ISP of lowest price

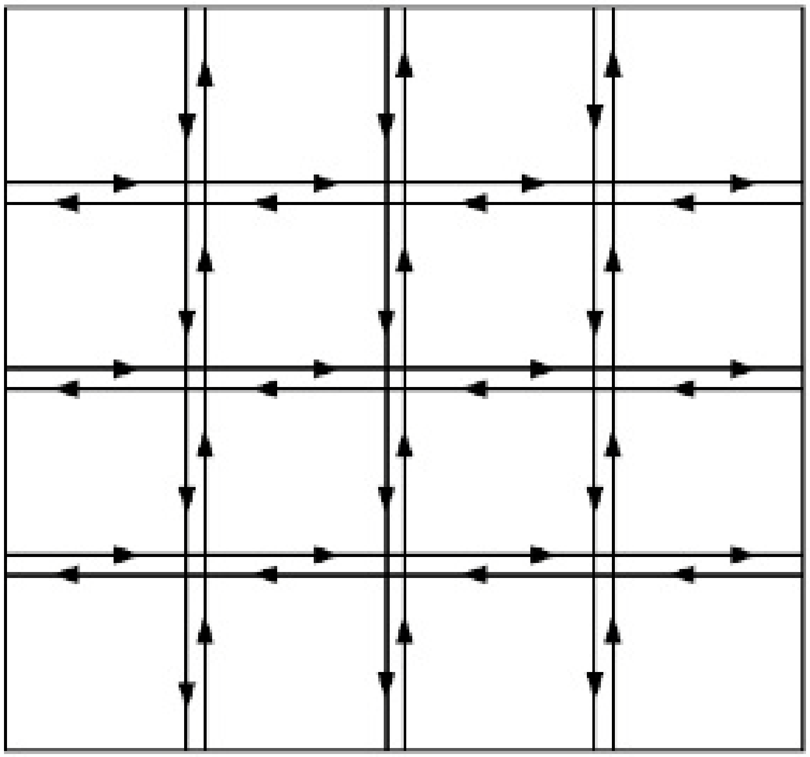
Simulation setup

- A square of 1200x1200 m
- 9 Access Points per ISP
- Initial price for both ISP 50 to 55
- 3,5 Mbps per AP
- AP coverage area: a circle with radius 285 m
- 450 mobile users
- Manhattan /Random walk mobility models
- Streets every 100 m
- Pedestrian velocity
- Every user ask for 50 to 150 Kbps.

Access Points distribution



Users' mobility model



Rules with respect to Service Demand

1. ***if*** *few_users* ***and*** *users_steady* ***then*** *decrease_price_slow*
2. ***if*** *few_users* ***and*** *users_decreasing_slow* ***then*** *decrease_price_slow*
3. ***if*** *few_users* ***and*** *users_decreasing_fast* ***then*** *decrease_price_fast*
4. ***if*** *few_users* ***and*** *users_increasing_slow* ***then*** *keep_price*
5. ***if*** *few_users* ***and*** *users_increasing_fast* ***then*** *keep_price*
6. ***if*** *mid_users* ***and*** *users_steady* ***then*** *increase_price_slow*
7. ***if*** *mid_users* ***and*** *users_decreasing_slow* ***then*** *decrease_price_slow*
8. ***if*** *mid_users* ***and*** *users_decreasing_fast* ***then*** *decrease_price_fast*
9. ***if*** *mid_users* ***and*** *users_increasing_slow* ***then*** *keep_price*
10. ***if*** *mid_users* ***and*** *users_increasing_fast* ***then*** *increase_price_slow*
11. ***if*** *lots_users* ***and*** *users_steady* ***then*** *keep_price*
12. ***if*** *lots_users* ***and*** *users_decreasing_slow* ***then*** *decrease_price_slow*
13. ***if*** *lots_users* ***and*** *users_increasing_slow* ***then*** *increase_price_slow*
14. ***if*** *lots_users* ***and*** *users_increasing_fast* ***then*** *increase_price_fast*
15. ***if*** *lots_users* ***and*** *users_decreasing_fast* ***then*** *decrease_price_fast*

Rules with respect to the Competitor

1. ***if*** *competitor_price_lower* ***and*** *competitor_price_decreasing_slow* ***then*** *decrease_price_slow*
2. ***if*** *competitor_price_lower* ***and*** *competitor_price_decreasing_fast* ***then*** *decrease_price_fast*
3. ***if*** *competitor_price_lower* ***and*** *competitor_price_steady* ***then*** *decrease_price_slow*
4. ***if*** *competitor_price_lower* ***and*** *competitor_price_increasing_slow* ***then*** *decrease_price_slow*
5. ***if*** *competitor_price_lower* ***and*** *competitor_price_increasing_fast* ***then*** *decrease_price_fast*
6. ***if*** *competitor_price_higher* ***and*** *competitor_price_decreasing_fast* ***then*** *decrease_price_slow*
7. ***if*** *competitor_price_higher* ***and*** *competitor_price_steady* ***then*** *increase_price_slow*
8. ***if*** *competitor_price_higher* ***and*** *competitor_price_increasing_slow* ***then*** *increase_price_fast*
9. ***if*** *competitor_price_higher* ***and*** *competitor_price_increasing_fast* ***then*** *increase_price_fast*

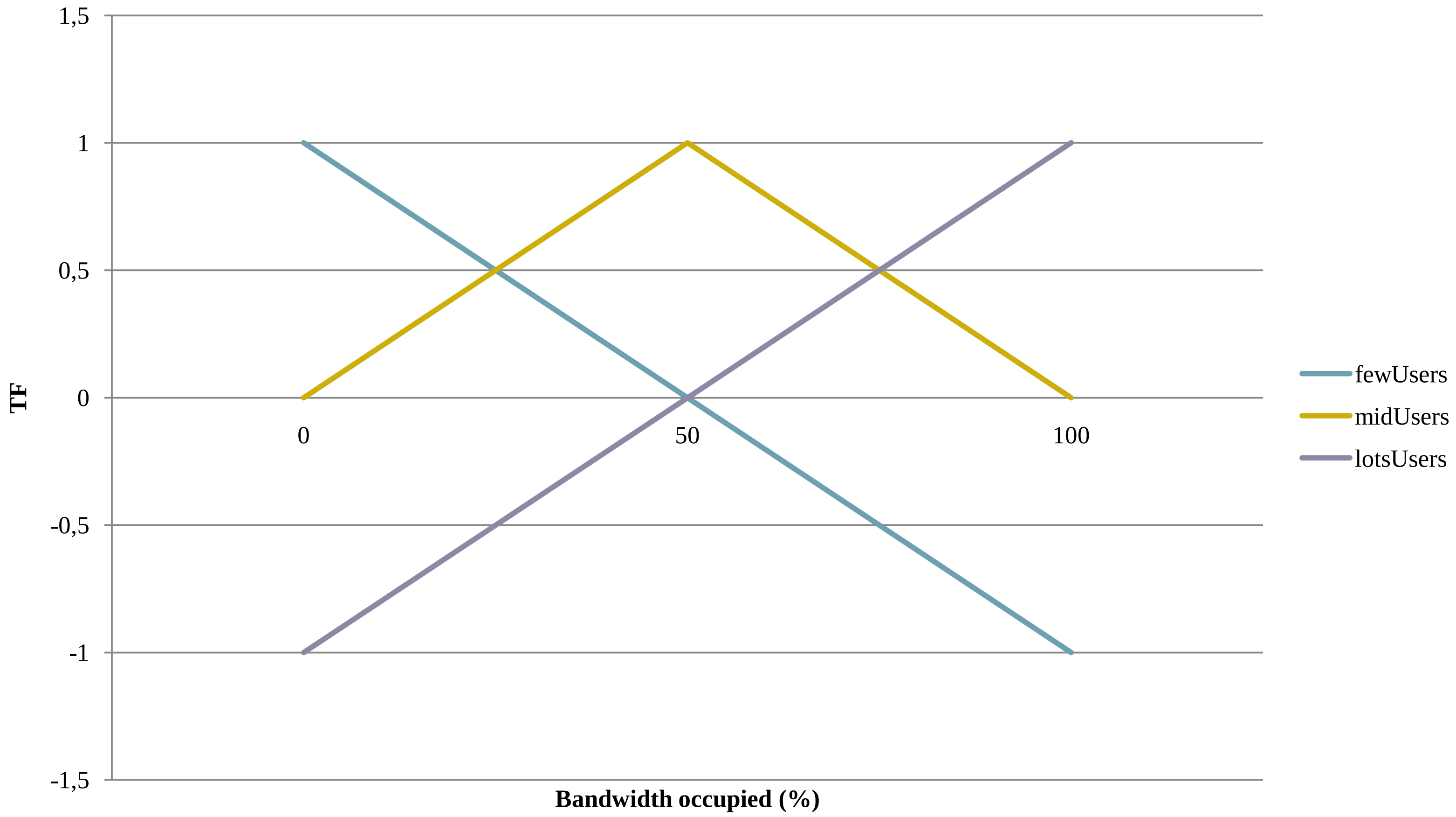
Tautness Functions have to be defined on conditions related to the Service Demand

- *fewUsers*
- *midUsers*
- *lotsUsers*
- *usersSteady*
- *usersDecreasingSlow*
- *usersDecreasingFast*
- *usersIncreasingSlow*
- *usersIncreasingFast*

Tautness Functions have to be defined on conditions related to the Competitor

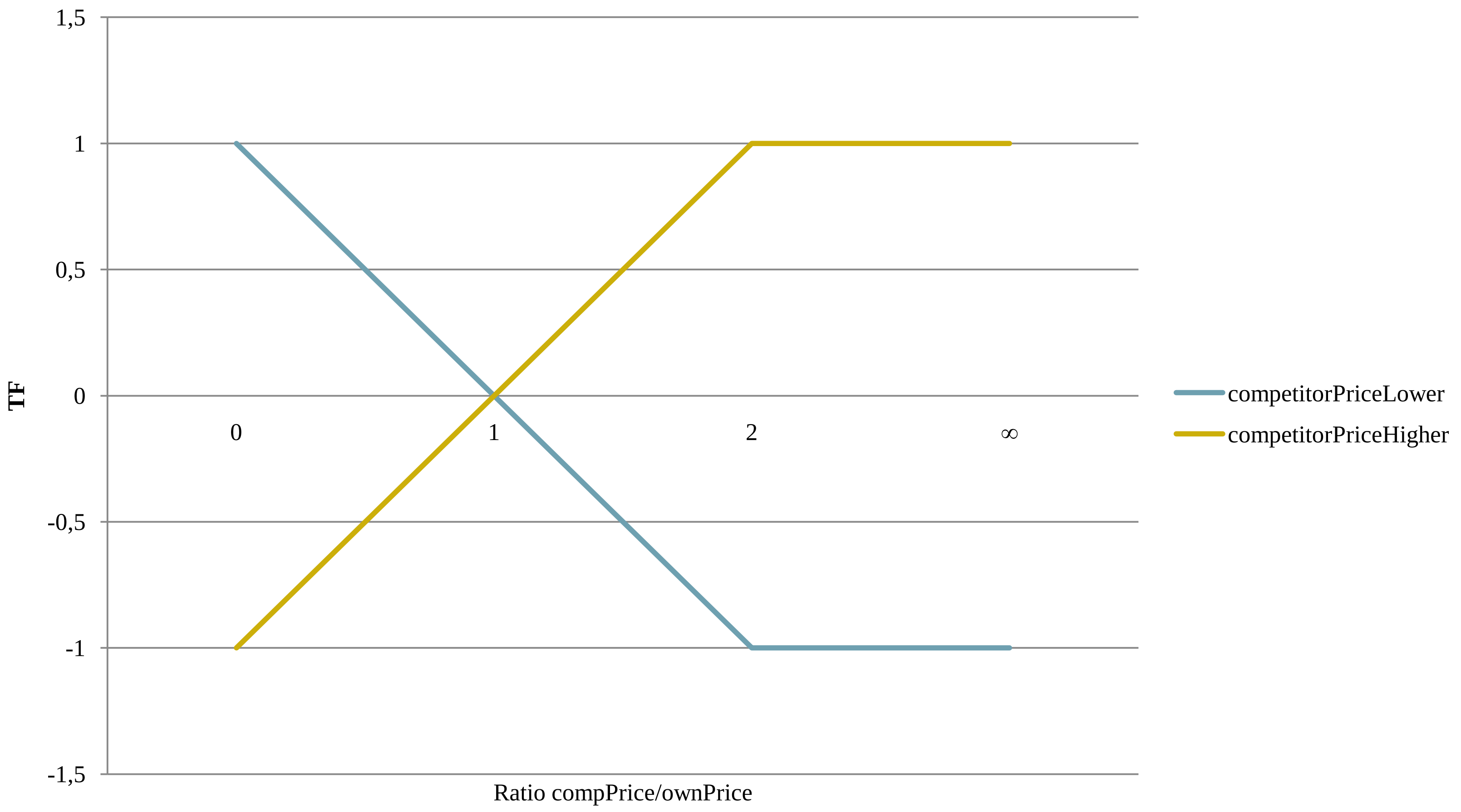
- *competitorPriceLower*
- *competitorPriceHigher*
- *competitorPriceSteady*
- *competitorPriceDecreasingSlow*
- *competitorPriceDecreasingFast*
- *competitorPriceIncreasingSlow*
- *competitorPriceIncreasingFast*

Proposed Tautness Functions (1)



Functions fewUsers, midUsers and lotsUsers

Proposed Tautness Functions (2)

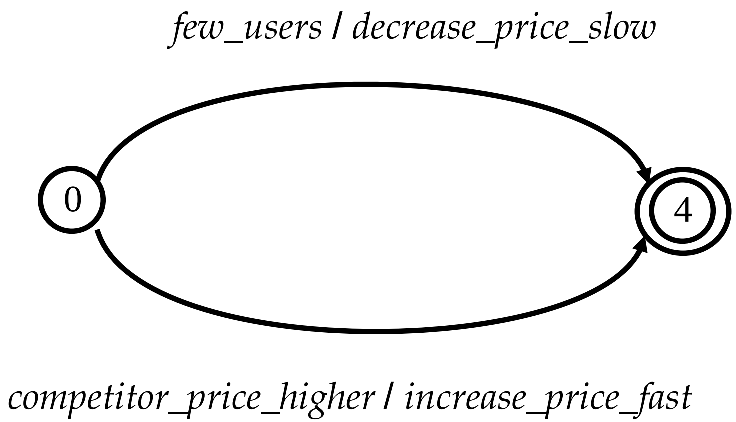


Functions competitorPriceLower and competitorPriceHigher

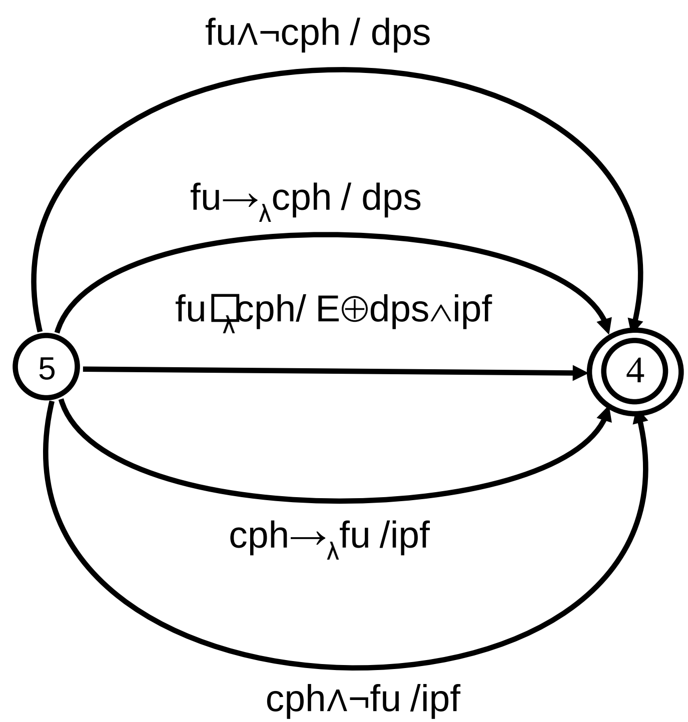
A conflict resolution example (1)

- ***if*** *few_users* ***then*** *decrease_price_slow*
- ***if*** *competitor_price_higher* ***then*** *increase_price_fast*

A conflict resolution example (2)

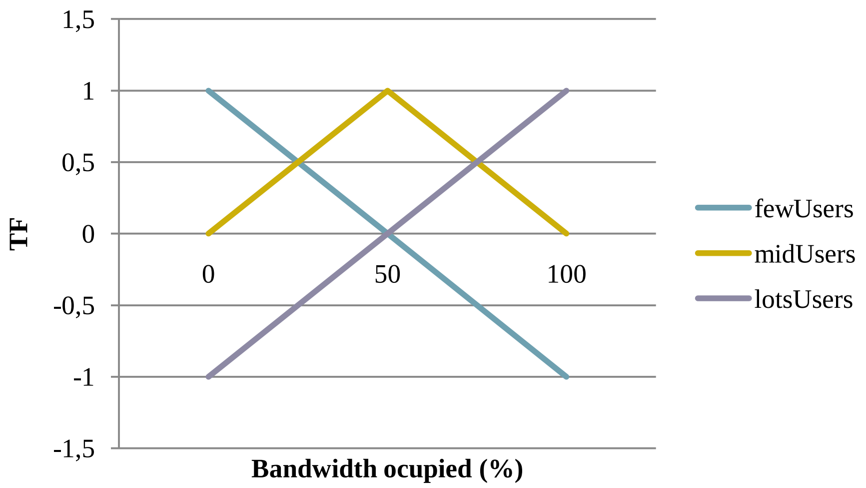


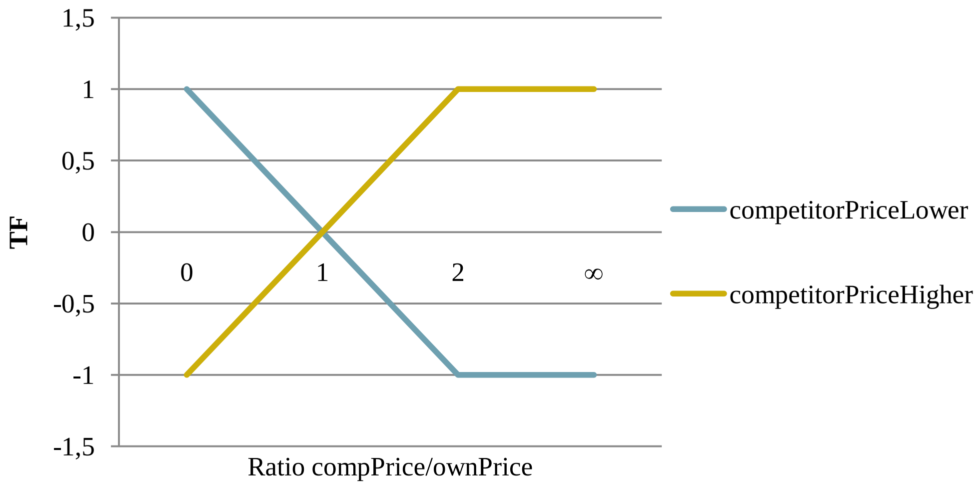
Before



After

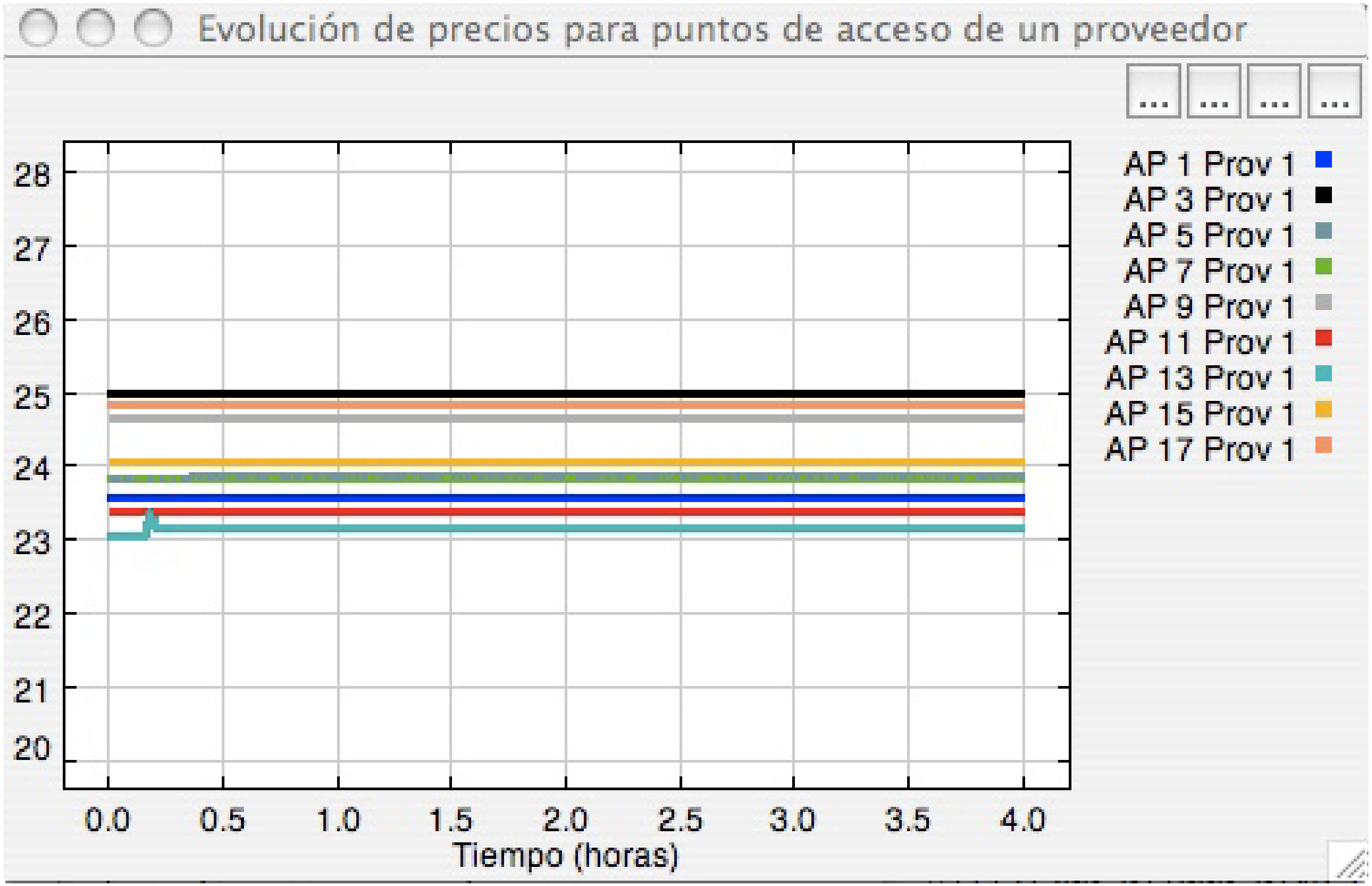
A conflict resolution example (3)





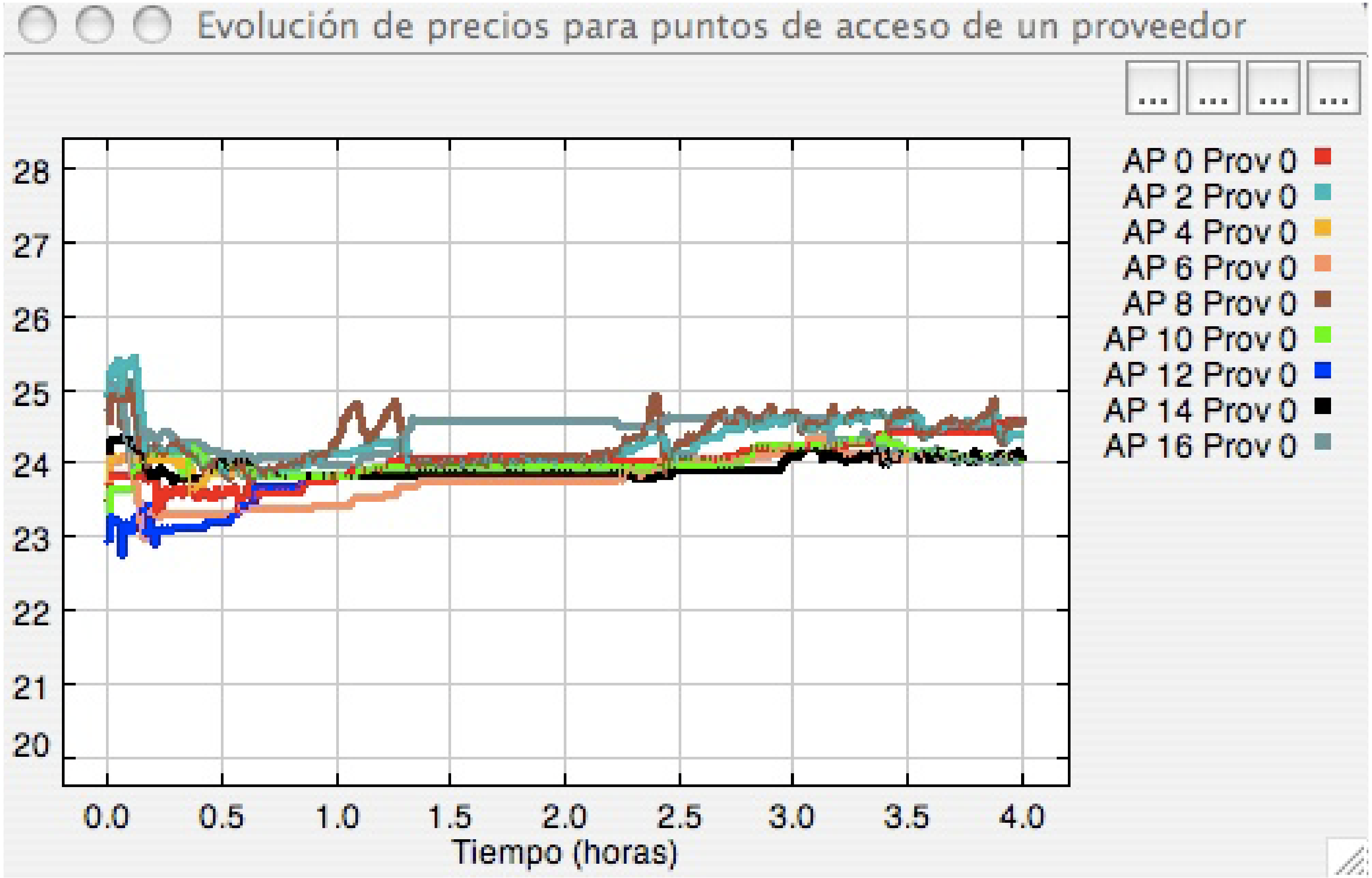
Results

Prices of Competitor APs



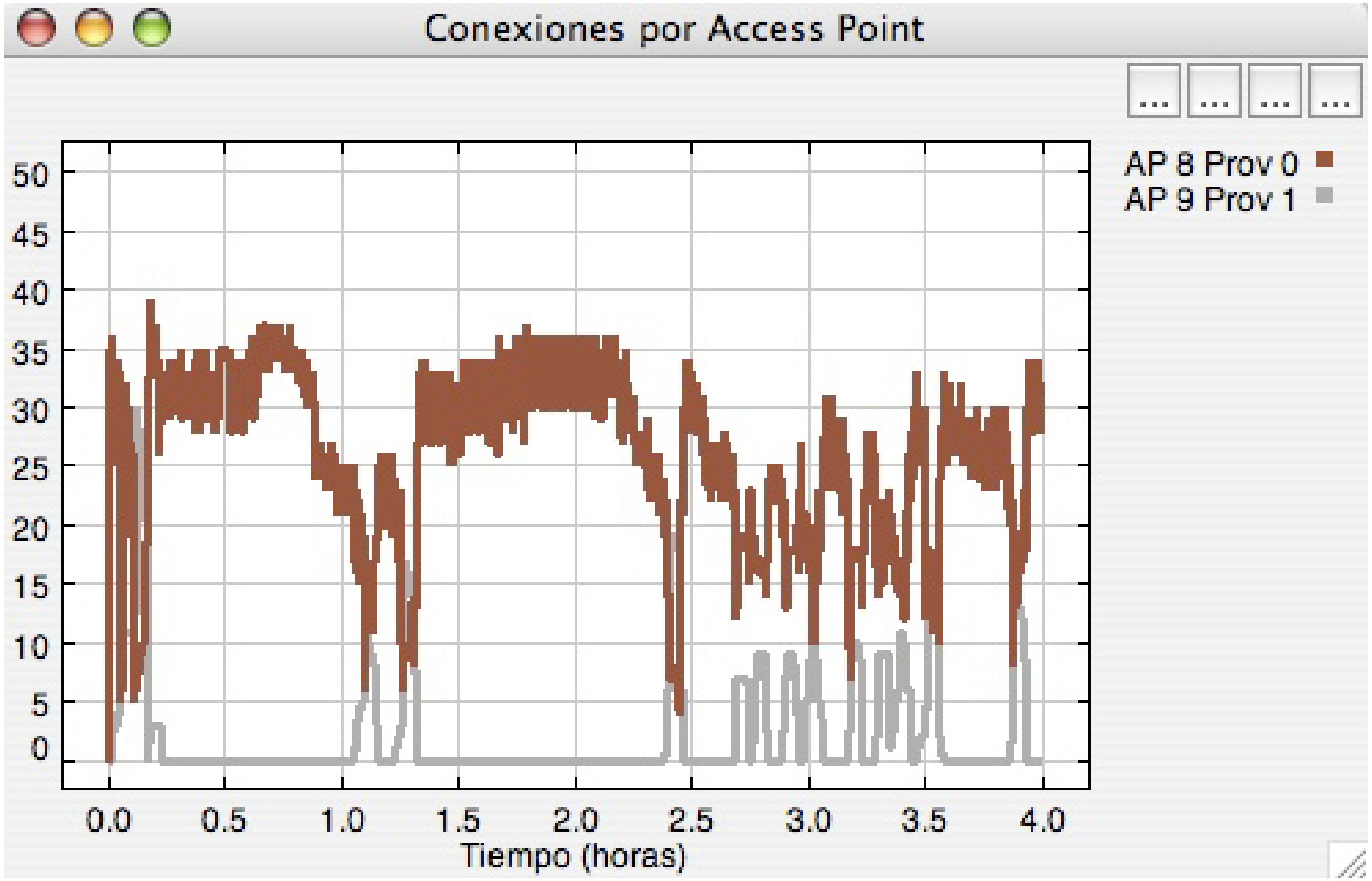
Results

Prices of a TFFST-based Provider



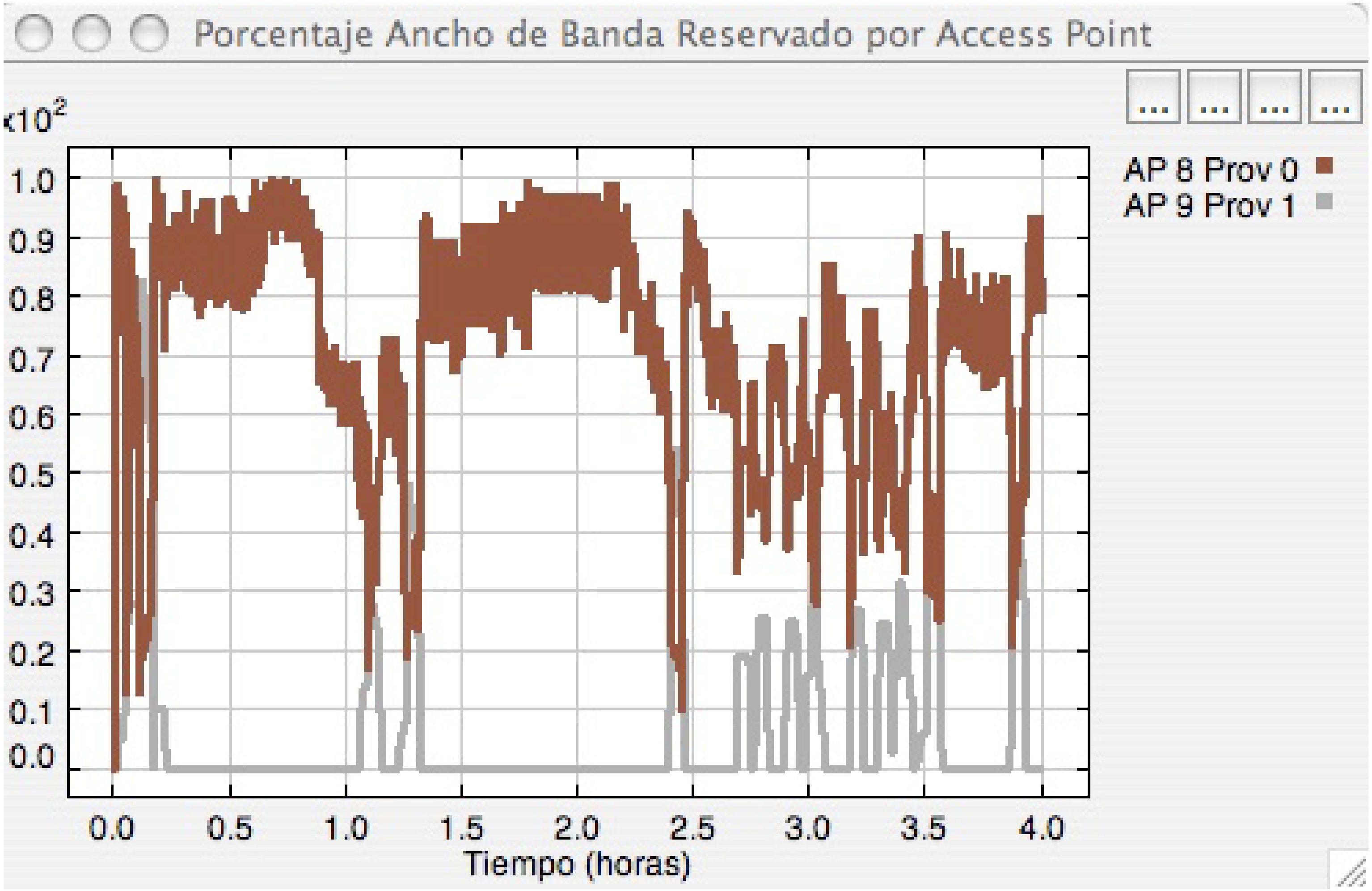
Results

Connections got from the TFFST-based Provider vs the Competitor in the same area

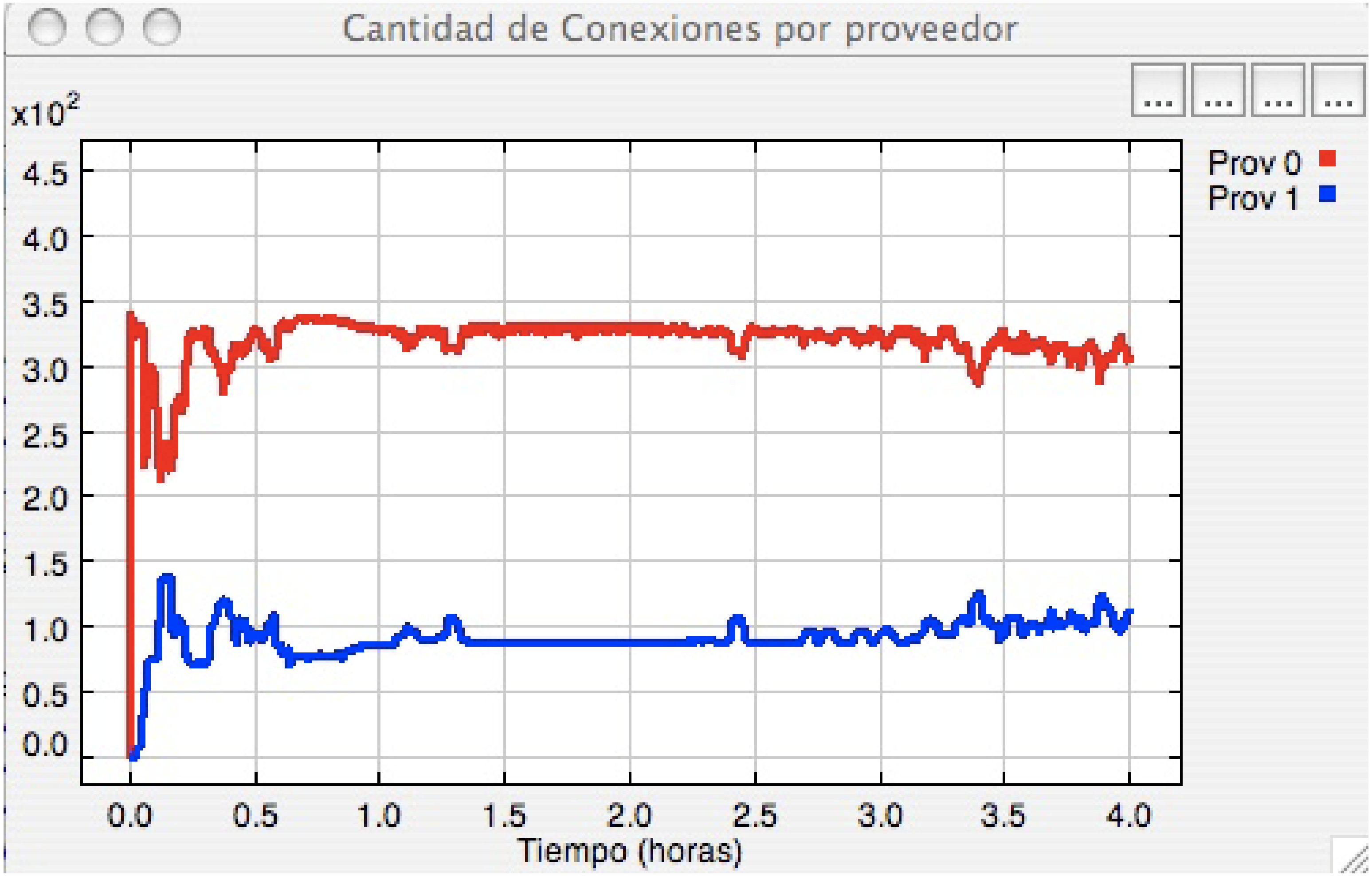


Results

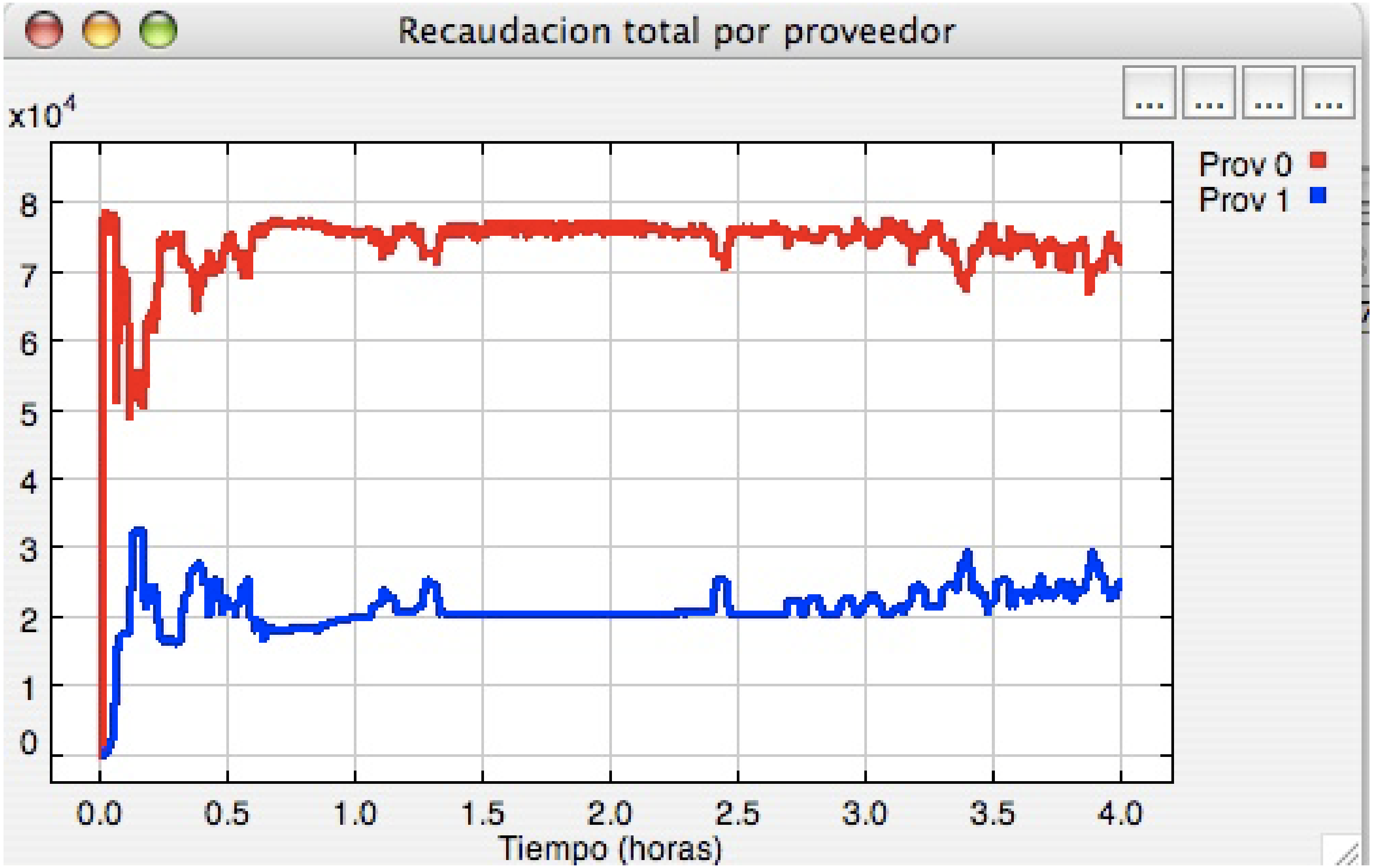
Percent of used bandwidth per access point



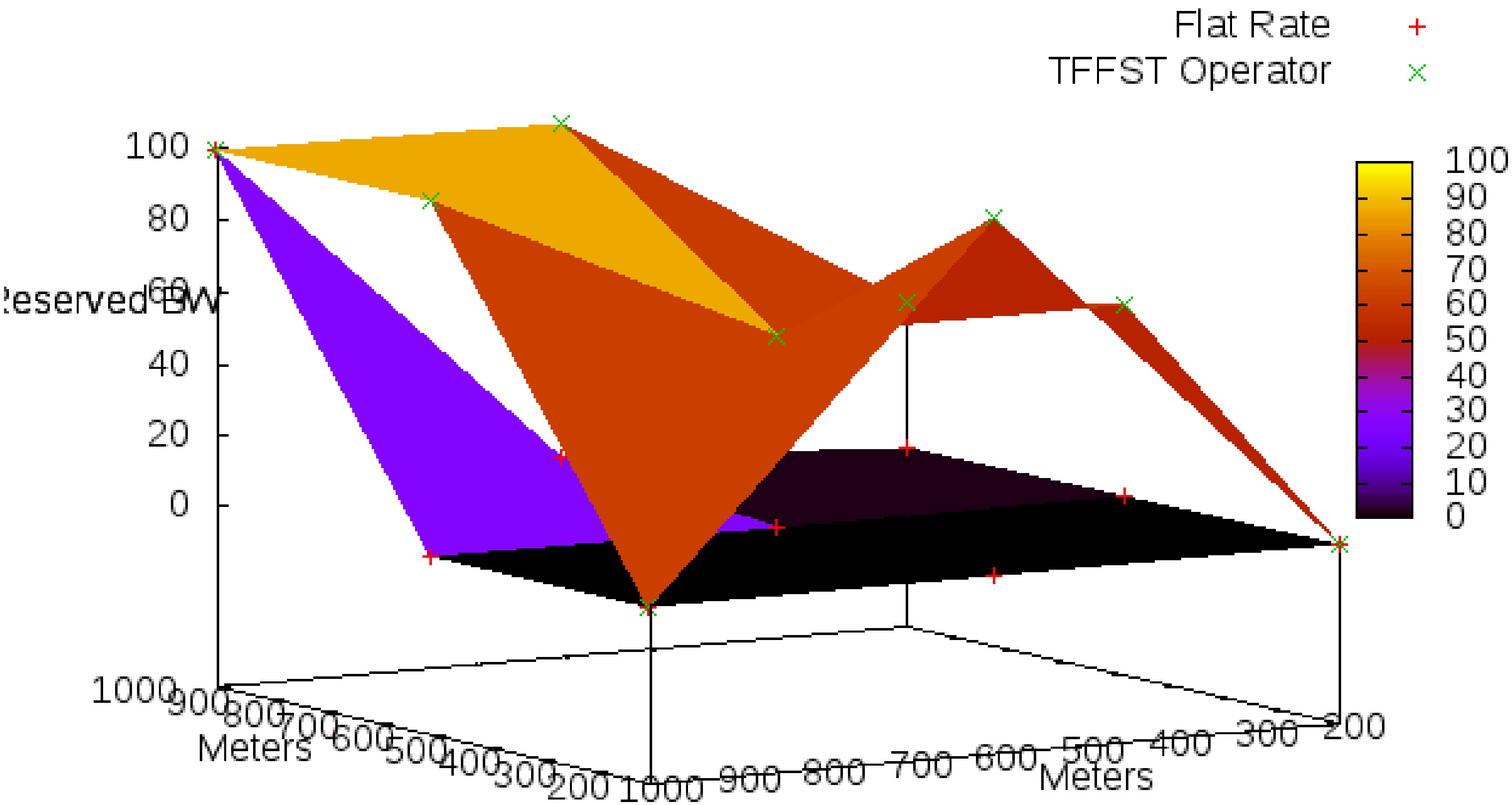
Results
Total number of connections got by each provider



Results
Total profits per provider



Load distribution per provider



Next Steps

- Model user demand functions
  - Users adopt decisions based on QoS (signal strength, bandwidth,…). This requires modeling of demand functions
  - Users have a given and different maximum price to accept
- Carry out simulation with ns3 (radio propagation)
- Substitute the RNAP protocol or similar making use of the APs beacon to distribute price information
  - Adopt some strategy for the zones where an AP can’t see the beacon of a competitor AP
- Shaping the Tautness functions to avoid instabilities
  - Making use of learning techniques or other mechanisms
- Comparison with other options

Acknowledgement

- This work is being done in collaboration with the following people
  - Dr. Javier Baliosian (Universidad de la República, Uruguay)
  - Dr. Javier Rubio (CINVESTAV, México)
  - Prof. José L. Melús (Universitat Politècnica de Catalunya)

- This work is funded by Ministerio de Ciencia e Innovación as project TEC2009-14598-C02-02