CmpE 593
Multiagent Systems

Pınar Yolum
pinar.yolum@boun.edu.tr

Department of
Computer Engineering
Boğaziçi University

Service Selection and Trust

Service Selection

- Finding the right service provider from a set of providers
- Yellow-pages
  - Lookup based on service criteria
  - May not always exist
  - May return many results
- Economic service selection
- Semantic service selection

Economic Service Selection

- Market-oriented programming
  - Design an environment with mechanisms for buying and selling
  - Little interaction between agents; mostly for exchanging goods at different prices
  - Preferences or abilities of agents are not explicitly considered
- Consumer and producer agents
  - Self-interested
  - Maximize their utility

Prices

- The computational state is described completely by current prices for the various goods
- Communications are between each participant and the market, and only in terms of prices
- Participants reason about others and choose strategies entirely in terms of prices being bid

Functions of a Market

- Provides this information to participants
- Takes requests (buy, sell bids) from participants, enforcing rules such as bid increments and time limits
- Decides outcome based on messages from participants, considering rules such as reserve prices.

Auctions

- Market where prices are determined dynamically
- Online auctions where agents participate
  - Must be fair and secure
  - Preserve privacy
- Auction types vary
  - Ascending (English) vs. Descending (Dutch)
  - Silent (auctioneer names a price; bids are silent) vs. outcry (bids name prices; auctioneer listens)
  - Hidden identity or not.
  - Combinatorial: involve *bundles* or sets of goods

English and Dutch Auctions

- English
  - Prices start low and increase
  - Highest bidder gets the object at price bid
  - Variations:
    - Minimum bid increment
    - Reserve price (no sale if too low)
    - Limited time
- Dutch
  - Prices start high and decrease
  - First to interrupt wins

Sealed-Bid Auctions

- One-shot bidding without knowing what other bids are being placed.
- Used by governments and large companies to give out certain large contracts (lowest price quote for stated task or procurement).
  - All bids are gathered.
  - Auctioneer decides outcomes based on given rules (e.g., highest bidder wins and pays the price it bid).
- Vickrey Auction:
  - Second price sealed bid auction
  - Highest bidder wins, but pays the *second* highest price

Continuous Double Auction

As in stock markets.

- Multiple sellers and buyers, potentially with multiple sell and buy bids each.
- Buy bids are like upper bounds
- Sell bids are like lower bounds
- *Clears* continually:
  - The moment a buyer and seller agree on a price, the deal is done and the matching bids are taken out of the market
  - Possible, a moment later a better price may come along, but it will be too late then.

Auction Management (1)

- Bidding rules to govern, e.g.,
  - Whose turn it is
  - What the minimum acceptable bid is, e.g., increments
- Information disclosure
  - What information is revealed to participants?
    - Bid value
    - Bidder
    - Winning bid
    - Winner
    - How often

Auction Management (2)

- Bids are cleared when they are executed and taken out of the market.
  - How are bids matched?
  - Who?
  - What prices?
  - How often?
  - Until when?

Problems

- More applicable for services that differ only on price
- Services differ on quality?
- Negotiation of service descriptions
- Suggestions of service providers
- Semantic service selection
  - Requires deciding on which service provider will do the job best for the user
  - Take into account provider's reputation or customer's trust in the provider

Reputation

- Consider a society of principals, potentially each having opinions about the others.
  - The opinions are applied implicitly in whether and how different parties do business with each other
- Someone's reputation is a *general opinion* about that party
  - Sometimes partially probed by asking others

Reputation

The central agency is the authority that

- Authenticates users
- Records, aggregates, and reveals ratings
- Examples: E-bay, Slashdot.org, Amazon.com
- Provides the conceptual schema for
  - How to capture ratings (typically a number and text)
  - How to aggregate them
  - How to decay them over time

Why a Decentralized Approach?

Problems with explicit aggregation

- *Context and understanding:* The contexts of usage may not be in agreement.
- *Privacy:* The parties providing their ratings are stating publicly (or to the reputation agency) what they may only wish to reveal in private.
- *Trust:* The parties using the ratings don't necessarily know where the ratings come from.

Confidence vs. trust

Luhmann's distinction:

- Hope: Wish it will come true (no basis).
- Confidence: Think it will come true (based on evidence).
- Trust: Commit to action with partly uncertain consequences.
- Risk (vs. Danger): Unexpected results may be a consequence of our decisions (not just fate).

Varieties of Commitment

Szompka relates trust to commitments:

- *Anticipatory trust:* Expecting the other party to do its work normally (choosing an airline based on past reliability).
- *Responsive trust:* Giving up control on an object and giving it to someone else (leaving a child with a babysitter).
- *Evocative trust:* Expecting to initiate reciprocal trust.

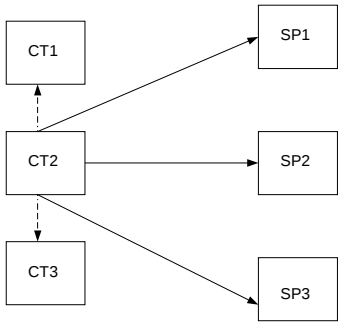
Circumstances of Commitment

- Range of consequences (employment vs. trip)
- Expected duration (marriage vs. date)
- Possibility of withdrawal (pre-commitment)
- Amount of risk relative to its probability (flight vs. surgery)
- Presence of insurance (bank with state guarantees)
- Value of object to be trusted (lending a car vs. a book)

Computational Trust

- Institutional Trust
  - Organizations monitor members' actions
  - Ensure a quality of service
  - Realized by digital certificates
- Local Trust
  - Based on personal evidence
  - Prior interactions
- Social Trust
  - Based on evidence from others
  - Information sources should be trustworthy

Computational Trust (General Structure)



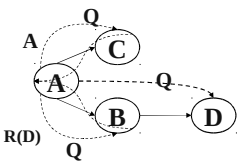
- Send request to SPx
- Receive a service
- Evaluate the service
- Update the model of SP1 based on evidence
- Look at SP models to choose an SP

Example Systems

- Beta Reputation System
  - Consumers send [good, bad] counts for SP1
  - Trustor adds the information
  - Uses a beta distribution to predict the reputation
  - Assume most ratings are fair
- TRAVOS, Referral Networks, POYRAZ

Social Trust: Referrals

- An agent represents a principal offering or searching for services.

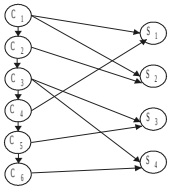


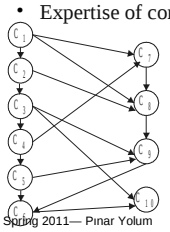
- An agent generates a query for a service and sends it to its *neighbors* (a small subset of its acquaintances). Each neighbor may provide the service or refer to other agents (based on its *referral policies*).
- Each agent models the *expertise* (quality of a service) and *sociability* (quality of the referrals) of its acquaintances.
- Based on these models, each agent can change its set of neighbors (using its *neighbor selection policy*).
- *Referal network:* as induced by the neighborhood relation.

Application Domains

Commerce:

- Distinct service producers and consumers.
- Producers have expertise, consumers have sociability.
- Answers are easy to evaluate.
- Expertise of consumers does not increase.





Knowledge Management:

- All agents can be producers and consumers.
- Answers are harder to evaluate.
- Expertise of consumers may increase (expertise of the producers can be cached by others).

Directions

- Most approaches attempt to find the most trustworthy SP
  - Cost, constraints not taken into account
- Generally, assumed to have access to large data on previous experience
  - Some services are only required once
- You need a group of Sps.