Agent Negotiation

Content

- Negotiation principles
- Negotiation protocols
  - Voting
  - MCP
- Negotiation domains
- Negotiation protocols continue
  - Auctions
  - Contract Net Protocol (CNP)
- Example of CNP
- Example of Production sequencing as negotiation

References

- Curriculum: Wooldridge: ”Introduction to MAS”
  - Chapters 11, 12, 14, 15

- Additioonal reading (not curriculum)
  - Y.Shoham, K. Leyton-Brown. Multi-Agent Systems (Algorithmic, Game-Theoretic and Logical Foundations). Cambridge Press, 2009. Incorrected manuscript available at http://www.masfoundations.org/mas.pdf
  - Tim Baarslag et al. When Will Negotiation Agents Be Able to Represent Us? The Challenges and Opportunities for Autonomous Negotiators, 2017 https://www.ijcai.org/proceedings/2017/0653.pdf
  - S. Fatima, S.Kraus, M. Wooldridge. Principles of Automated Negotiations. Cambridge University Press, 2015
  - Chen Zhaoxsuan et al. Negotiation Based Collision Avoidance Scheme for Autonomous Mobile Robots. 2017, https://link.springer.com/chapter/10.1007/978-3-319-78139-6_34

What is negotiation?

Two students decide to work together on their exercises. They have to decide upon a time. One prefers to work on Thursday afternoons after the lecture while the other prefers to work on Friday morning. How do they decide upon a time to do the work?

Negotiation Principles

Definition 1 (adopted from Davis & Smith)

Negotiation is a process of improving agreement (reducing inconsistency and uncertainty) on common viewpoints or plans through the exchange of relevant information

- two-way exchange information
- each party evaluates the information from its own perspective
- final agreement is achieved by mutual selection

Negotiation Principles

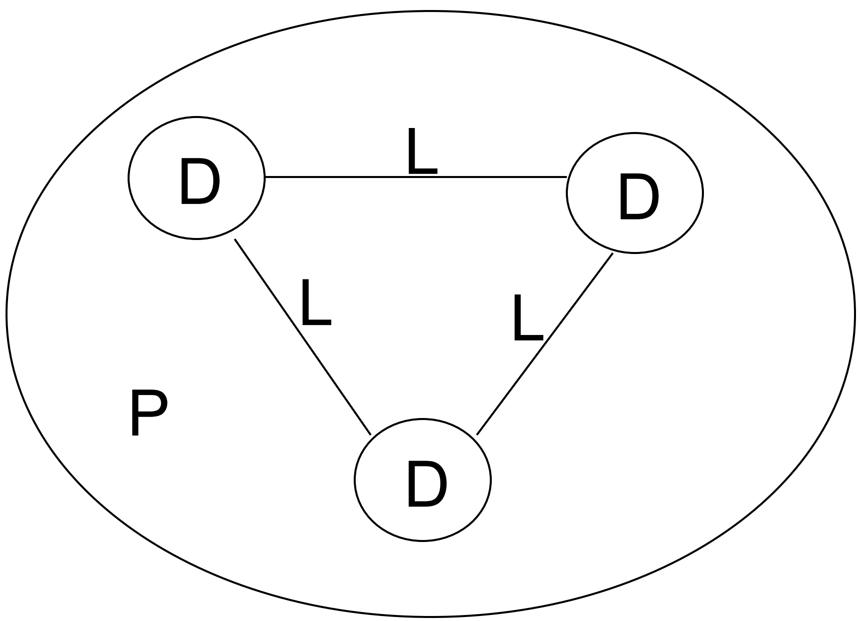
Definition 2 (Pruitt) - psychosocialogical

Negotiation is a process by which a joint decision is made by two or more parties. The parties first verbalize contradictory demands and then move towards agreement by a process of concession or search for new alternatives

Mutual conflict!

Basic negotiation categories

- Negotiation language category
- Negotiation decision category
- Negotiation process category



Language category

- language primitives
- object structure
- internal protocol
- semantics

Decision category

Which internal strategy to choose and which primitives to choose inside of protocol

- preferences
- utility functions
- comparing and matching functions
- negotiation strategies

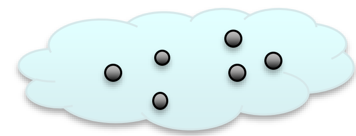
Negotiation process category

- Procedural negotiation model
- System behavior and analysis

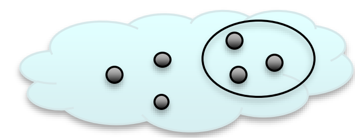
Negotiation Components

Any negotiation setting will have 4 components:

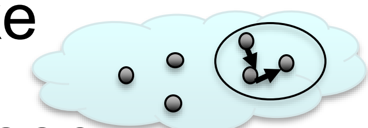
1. Negotiation set represents the space of possible proposals that agents can make



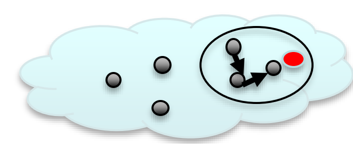
2. Protocol: defines the legal proposals that agents can make



3. Collection of strategies: (one for each agent) determines what proposals the agent will make

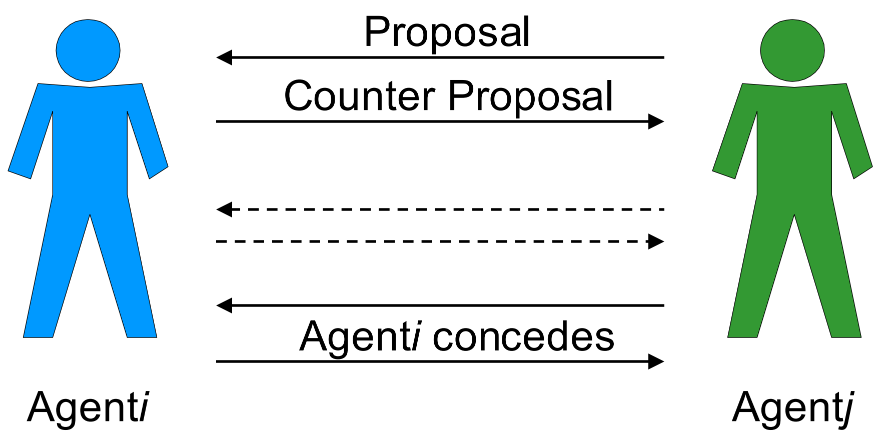


4. Rule: to determine when an agreement has been reached



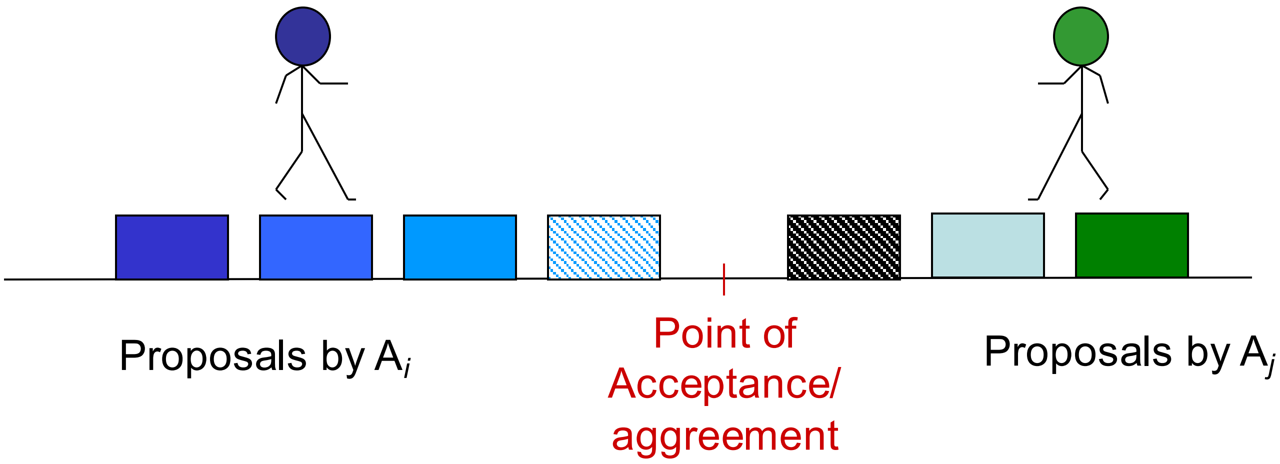
Negotiation Process 1

- Negotiation usually proceeds in series of rounds, with every agent making a proposal at every round.
- Communication during negotiation:



Negotiation Process 2

- Another way of looking at the negotiation process is:



Complex Negotiations

- Some attributes that make the negotiation process complex are:
  - Multiple attributes:
    - Single attribute (price) – symmetric scenario.
    - Multiple attributes – several inter-related attributes, e.g. buying a car.
  - The number of agents and the way they interact:
    - One-to-one, e.g. single buyer and single seller .
    - Many-to-one, e.g. multiple buyers and a single seller, auctions.
    - Many-to-many, e.g. multiple buyers and multiple sellers.

Decision category
Utility

- Preferences - expectations about outcome of negotiation actions

- Utility can be defined as a difference between the worth of achieving a goal and the price paid in achieving it.

Utilities and Preferences

- Assume we have 2 agents: $Ag = \{i,j\}$.
- Assume $\Omega = \{\omega_1, \omega_2, \ldots\}$ is the set of "outcomes" that agents have preferences over.
- We capture preferences by utility functions:
  - $u_i : \Omega \rightarrow \Re$
  - $u_j : \Omega \rightarrow \Re$
- Utility functions lead to preference orderings over outcomes:
  - $\omega \geq_i \omega'$ means $u_i(\omega) \geq u_i(\omega')$
  - $\omega >_i \omega'$ means $u_i(\omega) > u_i(\omega')$

Multi-agent Encounters 1

- To express agents preferences we need a model of the environment in which the agents will act:
  - Agents simultaneously choose an action and, as a result, an outcome in Ω will result.
  - Actual outcome depends on a combination of actions.
- Environment behaviour given by state transformer function τ :

$$\tau : \underbrace{Ai}_{\text{Agent } i\text{'s action}} \times \underbrace{Aj}_{\text{Agent } j\text{'s action}} \rightarrow \Omega$$

Multi-agent Encounters 2

- Assume that each agent (*i* and *j*) has two possible actions:

1. C: cooperate
2. D: defect
   - Let $A = \{C,D\}$

State Transformer Functions

- $\tau\ (D,D) = \omega_1 \qquad \tau\ (D,C) = \omega_2 \quad \tau\ (C,D) = \omega_3 \quad \tau\ (C,C) = \omega_4$

  environment sensitive to actions of both agents

- $\tau\ (D,D) = \omega_1 \qquad \tau\ (D,C) = \omega_1 \quad \tau\ (C,D) = \omega_1 \quad \tau\ (C,C) = \omega_1$

  environment where neither agent has any influence

- $\tau\ (D,D) = \omega_1 \qquad \tau\ (D,C) = \omega_2 \quad \tau\ (C,D) = \omega_1 \quad \tau\ (C,C) = \omega_2$

  environment controlled by *j*

Agent's Preference

- Consider the case where both agents influence the outcome and they have the following utility functions:

$u_i(\omega_1)=2 \quad u_i(\omega_2)=2 \quad u_i(\omega_3)=3 \quad u_i(\omega_4)=3$

$u_j(\omega_1)=2 \quad u_j(\omega_2)=3 \quad u_j(\omega_3)=2 \quad u_j(\omega_4)=3$

$u_i(D,D)=2 \quad u_i(D,C)=2 \quad u_i(C,D)=3 \quad u_i(C,C)=3$

$u_j(D,D)=2 \quad u_j(D,C)=3 \quad u_j(C,D)=2 \quad u_j(C,C)=3$

- Then, agent$_i$'s preferences are:

$$C,C \geq_i C,D \quad >_i \quad D,C \geq_i D,D$$

- Agent$_i$ preferes all outcomes that arise through *C* over all outcomes that arise through *D*

Payoff Matrices

We can characterise the previous scenario in a

payoff matrix

e.g. Top right cell:

i cooperates, *j* defects

		i Defect	*i* Coop
j	Defect	*i*: 2, *j*: 2	*i*: 3, *j*: 2
j	Coop	*i*: 2, *j*: 3	*i*: 4, *j*: 4

•Agent *i* is the column player

(payoff received by *i* shown in top right of each cell)

•Agent *j* is the row player

Mechanisms, Protocols, Strategies

- Negotiation is governed by a mechanism or a protocol:
  - the public rules by which the agents will come to agreements.
- How can we design a fair protocol?
- Given a particular protocol, how can a particular strategy be designed that individual agents can use?

Mechanisms Design

"Leo Baekeland sold the rights to his invention, Velox photographic printing paper, to Eastman Kodak in 1899. It was the first commercially successful photographic paper and he sold it to Eastman Kodak for $1 million. Baekeland had planned to ask $50000 and to go down to $25000 if necessary, but fortunately for him, Eastman spoke first."

(Asimov, 1982)

Desirable properties of mechanisms

- **Convergence/guaranteed success** - if it ensures that eventually agreement will be reached
- **Maximising social welfare** – maximizing the sum of all agent's payoffs or utilities in a given solution.
    - it measures the global good of the agents
    - it can be used as a criterion for comparing alternative mechanisms

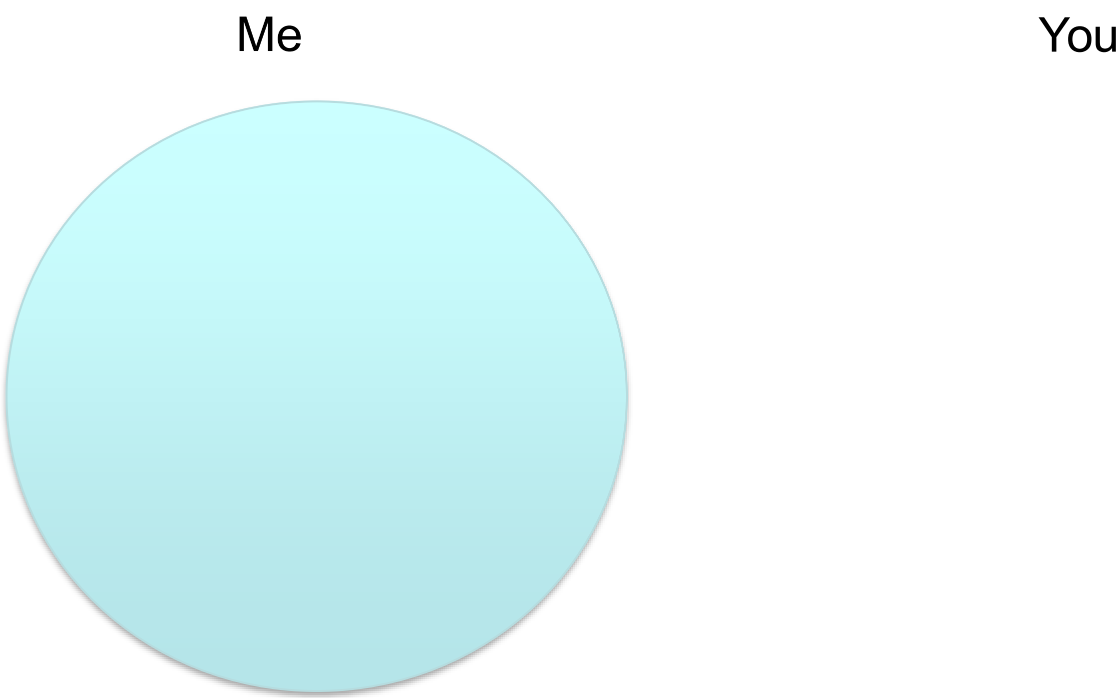
Desirable properties of mechanisms (Pareto efficiency)

- An outcome x is a Pareto Optimal (Efficient) if there is no other outcome x' such that at least one agent is better off in x' than in x and no agent is worse off in x' than in x.

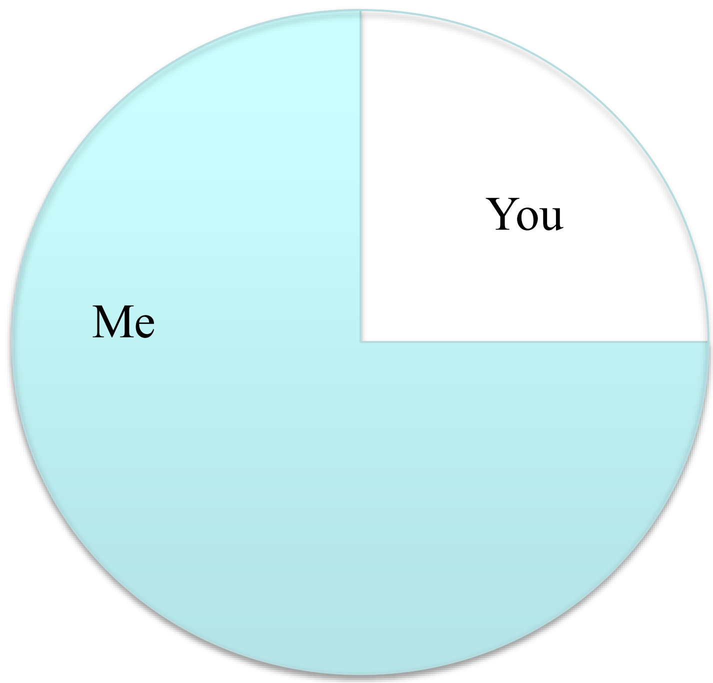
- Once the sum of payoffs is maximized, an agent's payoff can increase only if another agent's payoff decreases.

- Pareto efficiency measures global good and Social welfare maximizing solutions are subset of Pareto efficient ones.

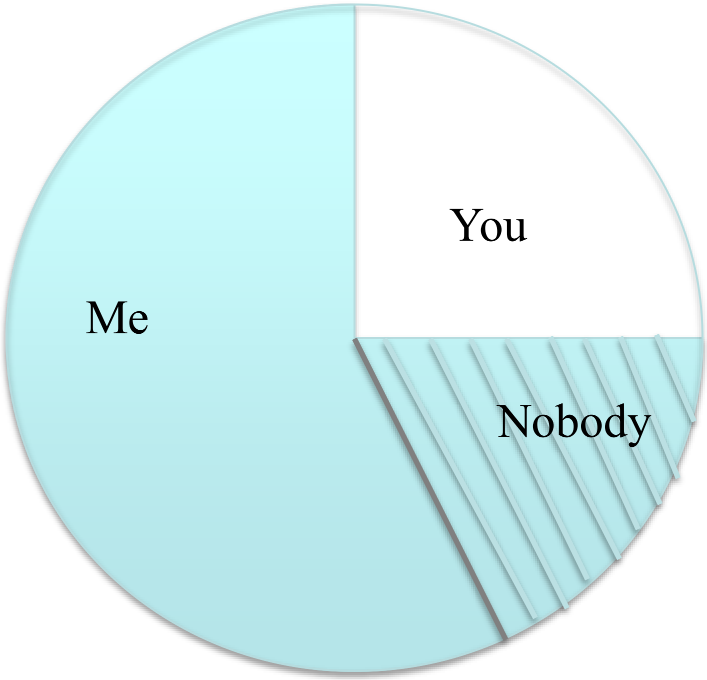
Pareto Optimality



Pareto Optimality



Pareto Optimality



Desirable properties of mechanisms (Individual rationality)

- participation in a negotiation is individually rational to an agent if the agent's payoff in the negotiated solution is no less than payoff that the agent would get by not participating in the negotiation

- a mechanism is individually rational if participation is individually rational for all agents

- if the negotiated solution is not individually rational for some agent then self-interested agent would not participate in that negotiation

Desirable properties of mechanisms

- **Computational efficiency**
  - as little computations is needed as possible
  - trade off between the cost of the process and the solution quality
- **Distribution**
  - all else being equal, distributed protocols should be preferred to avoid a single point of failure and a performance bottleneck
  - may conflict with minimizing the amount of communication that is required

Desirable properties of mechanisms

- **Stability** Among self-interested agents, mechanism should be designed to be stable (non-manipulable) - it should motivate each agent to behave in the desired manner

Voting protocol

- In a voting (social choice) all agents give input to a mechanism, and the outcome is chosen from given inputs.
- The outcome is obligatory to all participating agents

Truthful voters

Outcome is based on truthful individual's rankings of those outcomes

Let ***A*** is a set of agents, ***Ω*** is the set of feasible outcomes for the society and let each agent ***i*** from ***A*** has an asymmetric and transitive preference relation $>_i$ on ***Ω***.

A social welfare rule takes as input the agent's preference relations ($>_1$, ... , $>_n$) and produces as output the social preferences denoted by a $>^*$

Simple majority voting

Submitted rankings:

42% : $w_S > w_M > w_{SD}$

14% : $w_M > w_S > w_{SD}$

44% : $w_{SD} > w_M > w_S$

SD is the winner but 56% ranked it as the least desirable outcome

Condorcet´s paradox

33% $w_1 >_i w_2 >_i w_3$

33% $w_3 >_j w_1 >_j w_2$

33% $w_2 >_k w_3 >_k w_1$

- Whatever candidate we choose 2/3 of voter´s prefer another candidate

Truthful voters (Desirable properties of a social rule)

1. a social ordering >* should exist for all possible inputs. >* should be defined for every pair $\boldsymbol{\omega}, \boldsymbol{\omega}'$ from $\boldsymbol{\Omega}$
2. >* should be asymmetric and transitive over Ω
   - if $\omega_1 R \omega_2$ then ***not*** $\omega_2 R \omega_1$
   - if $\omega_1 R \omega_2$ and $\omega_2 R \omega_3$ then $\omega_1 R \omega_3$
3. the outcome should be Pareto efficient
4. the scheme should be independent of irrelevant alternatives: introducing new $\boldsymbol{\omega}$ with new relations should not change the social ranking
5. no agent should be a dictator that for some voter *i* we have social function $f(\omega_1,\ldots, \omega_n) = \omega_i$

Theorem: *No social choice rule satisfies all these five conditions*

Truthful voters

To design social rules, the desired properties should be relaxed:

- possible candidate is – transitivity
  - ~~if a>b and b>c then a~~>c

Truthful voters

- binary protocol - an alternatives are voted on pairwise and the winner stays to challenge further while the looser is eliminated

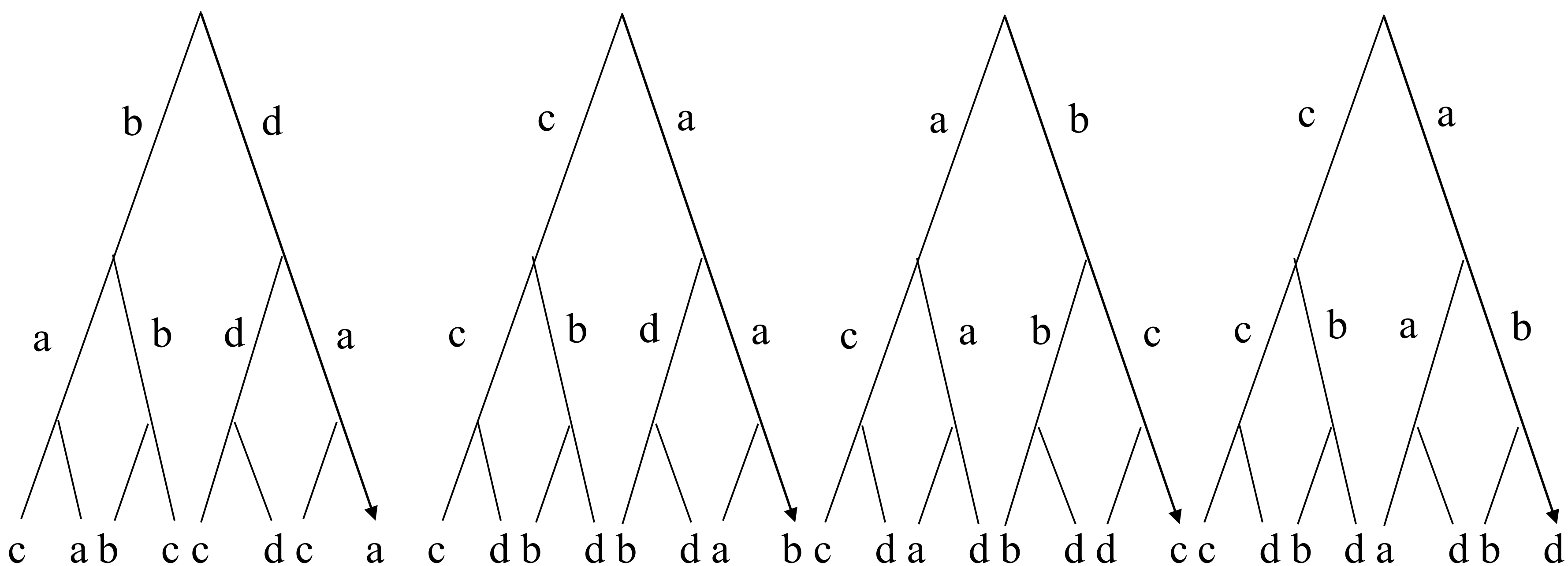
- Borda protocol- all alternatives are compared simultaneously and the one with the highest number of votes wins

Truthful voters (binary protocol)

33% of agents have preferences c > d > b > a

33% of agents have preferences b > a > c > d

33% of agents have preferences a > c > d > b



Truthful voters

If number of alternative outcomes is large, pair wise voting may be slow

Borda protocol

It assigns | Ω | points Ω whenever it highest in some agent's preference list, | Ω | - 1 whenever it is second and so on

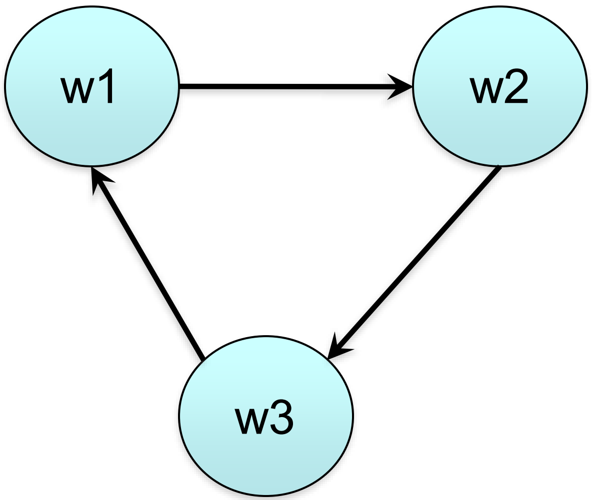
Agent	Preferences
1	a > b > c > d
2	b > c > d > a
3	c > d > a > b
4	a > b > c > d
5	b > c > d > a
6	c > d > a > b
7	a > b > c > d

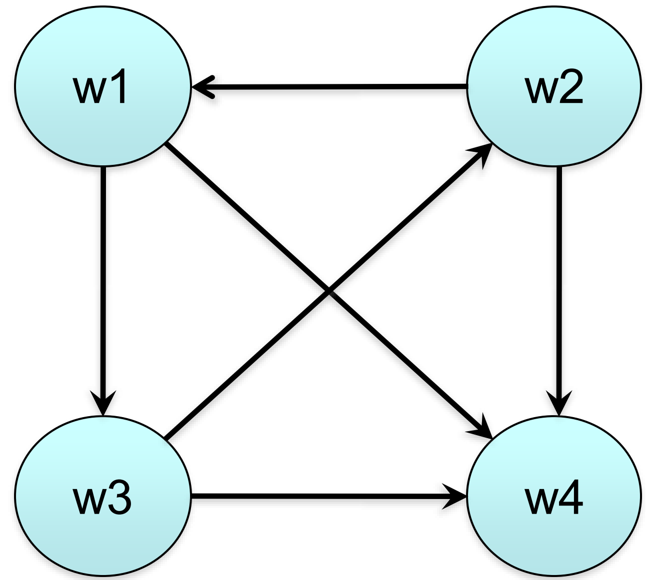
N	a	b	c	d
1	4	3	2	1
2	1	4	3	2
3	2	1	4	3
4	4	3	2	1
5	1	4	3	2
6	2	1	4	3
7	4	3	2	1

Borda count: c wins with 20, b has 19, a has 18, d loses with 13

Borda count with d removed: a wins with 15, b has 14, c loses with 13

Majority graph



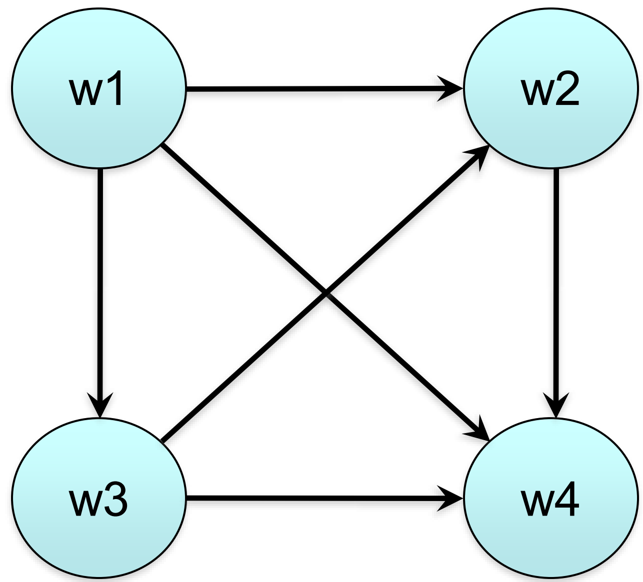


$w_1 >_i w_2 >_i w_3$

$w_3 >_j w_1 >_j w_2$

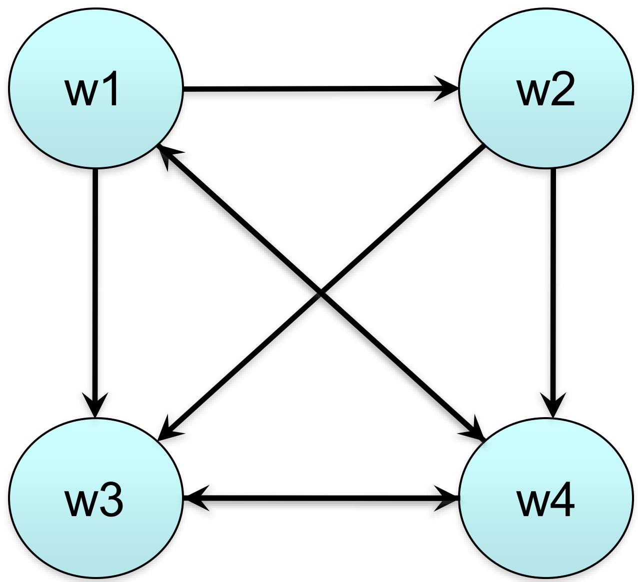
$w_2 >_k w_3 >_k w_1$

Condorcet winner



- w1 >* w3 >* w2 >* w4

Slater ranking



- w1 >* w2 >* w3 >* w4
- w1 >* w2 >* w4>* w3

Arrow´s theorem

- If $|\Omega|>2$ then if we satisfy Pareto efficiency and independence on irrelevant alternatives then the only voting procedure that satisfies is dictatorships

Strategic (Insincere) voters

- in reality it is seldom that all agent's preferences are known - usually agents have to reveal their preferences
- assuming knowledge of the preferences is equivalent to assuming that the agents reveal their preferences truthfully
- if an agent can benefit from insincerely declaring his preferences, he will do so
- the goal is to generate protocols such that when agents use them according to some stability solution concept (dominant strategy equilibrium) then desirable social outcomes follow
- the strategies are not externally imposed on the agents, but instead each agent uses the strategy that is best for itself

Strategic (insincere) voters

Theorem: Gibbard-Satterthwaite impossibility theorem

Let each agent's ordering $\boldsymbol{\theta_i}$ consists of a preference order $>_i$ on $\boldsymbol{\Omega}$. Let there be no restrictions on $>_i$, i.e. each agent may rank the outcomes $\boldsymbol{\Omega}$ in any order. Let $|\ \boldsymbol{\Omega}\ | > 2$. Now, if the social choice function f(•) is truthfully implementable in a dominant strategy equilibrium (or is not manipulable), then f(•) is dictatorial, i.e. there is some agent ***i*** who gets (one of) its most preferred outcomes chosen no matter what types the other reveal

Strategic (insincere) voters

Circumventing the impossibility theorem

The individual preferences may happen to belong to some restricted domain - thus invalidating the conditions of the impossibility theorem - and it is known that there are islands in the space of agent's preferences for which non-manipulable non-dictatorial protocols can be constructed.

Strategic (insincere) voters

Example (Sandholm)

- Let the outcomes be of the form $\boldsymbol{\omega=(g, \pi_1, \ldots, \pi_n)}$, where $\boldsymbol{\pi_i}$ is the amount of some divisible numeraire (e.g. money) that agent $\boldsymbol{i}$ receives in the outcome, and $\boldsymbol{g}$ encodes the other features of the outcome
- The agent's preferences are called quasilinear if they can be represented by utility functions of the form $\boldsymbol{u_i(\omega) = v_i(g) + \pi_i}$
- For example, in voting whether to build a joint pool, say $\boldsymbol{g = 1}$ if the pool is built and $\boldsymbol{g = 0}$ if not.
  - each agent's gross benefit from the pool $\boldsymbol{v_{grossi}(1)}$,
  - cost $\boldsymbol{P}$ of the pool will be divided equally among the agents, i.e. $\boldsymbol{\pi_i = -P/|A|}$.
  - an agent net benefit is $\boldsymbol{u_i(1) = v_{grossi}(1) - P/|A|}$

Strategic (insincere) voters

Quasilinearity of environment would require:

- no agent should care how others divide payoffs among themselves
- an agent's valuation $v_{grossi}(g)$ of the pool should not depend on the amount of money the agent will have
- when voting whether to build the pool or not, the agents who vote for the pool impose an externality on the others because the others have to pay as well

Strategic (insincere) voters

The solution is to make agents precisely internalize the externality by imposing a tax on those **agents whose vote changes the outcome**. The size of tax is exactly how much his/her vote lowers the other's utility

The Clarke tax algorithm

- Every agent i from A reveals his valuation $v^*_i(g)$ for every possible g (which may be non-truthful)
- The social choice is $g^* = \arg\max_g \Sigma\, v^*_i(g)$
- Every agent is levied a tax :

  $tax_i = \Sigma_{j \neq i}\, v^*_j(\arg\max_g \Sigma_{k \neq i}\, v^*_k(g)) - \Sigma_{j \neq i}\, v^*_j\,(g)$

Strategic (insincere) voters

Theorem. If each agent has quasilinear preferences, then, under the Clarke tax algorithm, each agent's dominant strategy is to reveal his true preferences, i.e. $v^*_i(g) = v_i(g)$ for all g.

In the example, if the pool is built then the utility for each agent i becomes

$$u_i(\omega) = v_i(1) - P/|A| - tax_i$$

and if not, then

$$u_i(\omega) = v_i(0)$$

Another example

- Truthful ranking
- Valuation $v_i(g)$

	8TB	16TB	64TB
CS	1	2	3
CD	3	1	2
ME	3.5	1	2
Sum	7.5	4	7

$$tax_i = \Sigma_{j \neq i}\, v^*_j(\arg\max_g \Sigma_{k \neq i}\, v^*_k(g)) - \Sigma_{j \neq i}\, v^*_j(g)$$

Tax for CS = 6.5-6.5 =0

Utility for CS = 1

Adopted from M. A. Goodrich

Example continue

- Non-truthful ranking

	8TB	16TB	64TB
CS	1	2	4
CD	3	1	2
ME	3.5	1	2
Sum	7.5	4	8

$$tax_i = \Sigma_{j \neq i}\, v^*_j(\arg\max_g \Sigma_{k \neq i}\, v^*_k(g)) - \Sigma_{j \neq i}\, v^*_j(g)$$

Tax for CS =6.5-4= 2.5
Utility for CS = 3-2.5 =0.5

Adopted from M. A. Goodrich

Strategic (insincere) voters

The Clarke tax algorithm

+ the mechanism leads to the socially most preferred ***g*** to be chosen

+ because of truth telling, the agents need not waste effort in counter speculating each other preference declarations

+ participation in the mechanism may only increase an agent's utility

- the mechanism does not maintain budget balance: too much tax is collected

- it is not coalition proof

Strategic (insincere) voters

Other way to circumvent the Impossibility theorem

- some fairness can be achieved by choosing the dictator randomly in the protocol

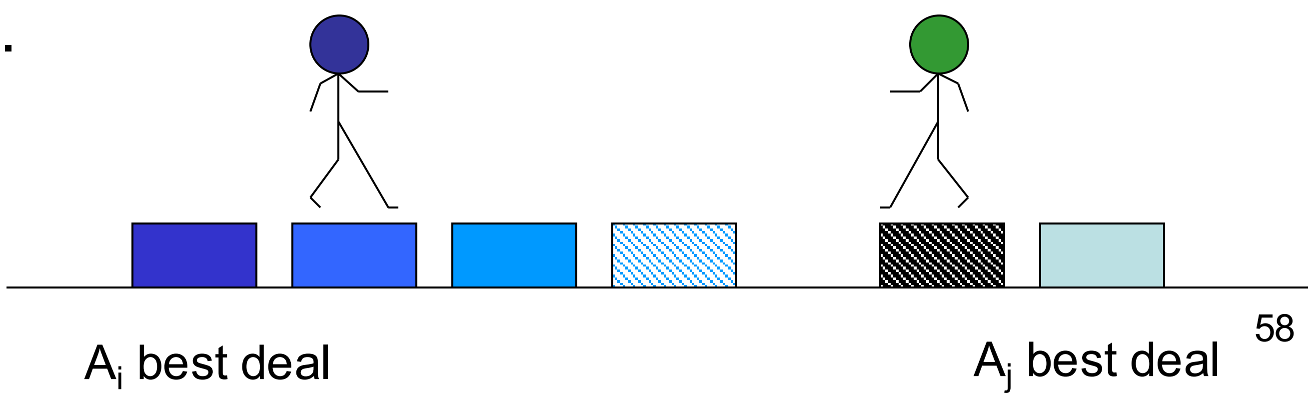
- to use a protocol for which computing an untruthful revelation (that is the better than the truthful one) is prohibitively costly computationally

The Monotonic Concession Protocol (MCP) 1

- Negotiation proceeds in rounds
- Space of possible deals is defined
- On round 1, agents simultaneously propose a deal from the negotiation set.
- Agreement is reached if one agent finds that the deal proposed by the other agent is at least as good or better than its proposal.

The Monotonic Concession Protocol 2

- If no agreement is reached, then negotiation proceeds to another round of simultaneous proposals.
- In round *u*+1, no agent is allowed to make a proposal that is less preferred by the other agent than the deal proposed at time *u*.
- If neither agent concedes, then negotiation terminates with a conflict deal.



The Monotonic Concession Protocol 3

- Advantages:
  - Symmetrically distributed (no agent plays a special role)
  - Ensures convergence
  - It will not go on indefinitely
- Disadvantages:
  - Agents can run into conflicts
  - Inefficient – no guarantee that an agreement will be reached quickly
  - Assumption of rationality (maximization of utility)

Key Questions

3 key questions to be answered:

1. What should an agent's first proposal be?
2. On any given round, who should concede?
3. If an agent concedes, then *how much* should it *concede*?

Negotiation strategies

A function from the history of the negotiation to the current offer that is consistent with the negotiation protocol.

It specifies how an agent will continue given a specific protocol and the negotiation up to this point

- concede unilaterally
- competitive: stand firm and employ pressure tactics
- cooperative: search for a mutually acceptable solution
- inaction
- breaking

The Prisoner's Dilemma 1

2 men are collectively charged with a crime and held in separate cells. They have no way of communicating with each other or making an agreement. They are told:

- if one confesses and the other does not, confessor will be freed and the other jailed for 3 years.
- if both confess, then each will be jailed for 2 years.
- If neither confess, then each will be jailed for 1 year.

•Confessing => defecting (*D*)

•Not confessing => cooperating (*C*)

If you were one of the prisoners, what would you do?

The Prisoner’s Dilemma 2

Payoff matrix for Prisoner’s Dilemma:

- Top left: If both defect,
  punishment for mutual defection.
- Top right: if *i* cooperates and *j* defects,
  i gets payoff of 0 while *j* gets 5.
- Bottom left: if *j* cooperates and i defects,
  j gets payoff of 0 while *i* gets 5.
- Bottom right:
  Reward for mutual cooperation.

		i	
		Defect	Coop
j	Defect	2 2	0 5
	Coop	5 0	3 3

✱Numbers in the payoff matrix reflect how good an outcome is for the agent. (not the number of year too be spent in jail) e.g.

$u_i(D,D)=2$ $u_i(D,C)=5$ $u_i(C,D)=0$ $u_i(C,C)=3$

$u_j(D,D)=2$ $u_j(D,C)=0$ $u_j(C,D)=5$ $u_j(C,C)=3$

The Prisoner's Dilemma 3

- The *individual rational agent* will *defect*!
  - This guarantees a payoff of no worse than 2
  - Cooperating guarantees a payoff of no worth than 0
- Defection is the best response to all possible strategies
  - Both agents defect and get a payoff = 2.
- If both agents cooperate, they will each get payoff = 3.

How can we apply the Prisoner's Dilemma to real situations?

- e.g. Arms races – nuclear weapons compliance treaty between two countries.
- Can you think of other situations?

Prisoner's dilemma

- Game theory notion of rational agents is wrong we not always maximize our utility but may think about maximizing social welfare
- We also may take a risk when risk is not too much or does not hurt us a lot
- Somehow the dilemma is being formulated wrongly

• Can we recover cooperation?
  – The Iterated Prisoner's Dilemma

Zero-sum interactions

- One agent can only get a more preferred outcome at the expense of the other agent
  - ➢ strictly competitive: $\omega >_i \omega'$ iff $\omega' >_j \omega$
- Zero-sum encounters
  - $u_i(\omega) + u_j(\omega) = 0$, for all $\omega \in \Omega$.
  - e.g. chess or checkers.

Dominance

- We say that s_1 dominates s_2 if the set of outcomes possible by i playing s_1 is preferred over the set of outcomes possible by i playing s_2.
  - w4>w3>w2>w1 s1=[w4, w3] s2=[w1, w2]
- sometimes it is possible to design mechanisms with dominant strategies - an agent is best off by using a specific strategy no matter what strategies the other agents use
- however, often an agent's best strategy depends on what strategies other agents choose and dominant strategy does not exist
- another stability criteria should be used - Nash equilibrium

Nash Equilibrium

- 2 strategies s_1 and s_2 are in Nash Equilibrium if:
  - Under the assumption that agent *i* plays s_1, agent *j* can do no better than play s_2;
  - Under the assumption that agent *j* plays s_2, agent *i* can do no better than play s_1;
- Neither agent has any incentive to deviate from a Nash Equilibrium.

		i Defect	*i* Coop
j	Defect	2 (i), 2 (j)	0 (i), 5 (j)
j	Coop	5 (i), 0 (j)	3 (i), 3 (j)

Desirable properties of mechanisms (Nash equilibrium)

- the strategy profile $S = \langle S_1, S_2, \ldots, S_n \rangle$ among agents A is in Nash equilibrium if for each agent i, S_i is the agent's best strategy (best response) given that the other agents choose strategies $S = \langle S_1, S_2, \ldots, S_{i-1}, S_{i+1}, \ldots, S_n \rangle$.

- in Nash equilibrium each agent chooses a strategy that is a best response to the other agent's strategies

Nash Equilibrium (Matching penny)

		i	
		heads	tails
j	heads	1 -1	-1 1
	tails	-1 1	1 -1

Nash Equilibrium

		i		
		rock	paper	scissors
j	rock	0 ; 0	1 ; -1	-1 ; 1
	paper	-1 ; 1	0 ; 0	1 ; -1
	scissors	1 ; -1	-1 ; 1	0 ; 0

Mixed strategy:

- Play s_1 with probability p_1
- Play s_2 with probability p_2
- …
- Play s_n with probability p_n

Nash´s theorem

Every game where player has a finite set of possible strategies has Nash equilibrium in mixed strategy

Nash Equilibrium - Example

The Battle of the Sexes

- Conflict between a man and a woman, where the man wants to go to a *Mission Impossible Fight* and the woman wants to go to a *Ballet*
- They are deeply in love. So, they would make a sacrifice to be with each other.

- 2 Nash Equilibria
  - Strategy combination (*Mission Impossible*, *Mission Impossible)*
  - Strategy combination (*Ballet*, *Ballet)*

Woman

Man		Mission	Ballet
	Mission	1 2	0 0
	Ballet	0 0	2 1

Ref: "Games and Information, E. Rasmussen, 2001

Nash equiliblium

Basic problems

- multiple Nash equilibrium may exist and it is not obvious which one the agents should actually apply

- it guarantees stability only in the beginning of the game

- Nash equilibrium is often too weak because subgroups of agents can deviate in a coordinated manner

MCP - Risk evaluation

Which agent should concede?

- One way to think about which agent should concede is to consider how much each has to loose by running into conflict at that point.

Risk evaluation

- if after step t Agent1 decides not to make a concession, then it takes a risk that Agent2 will also not make a concession, and that they will run into a conflict.

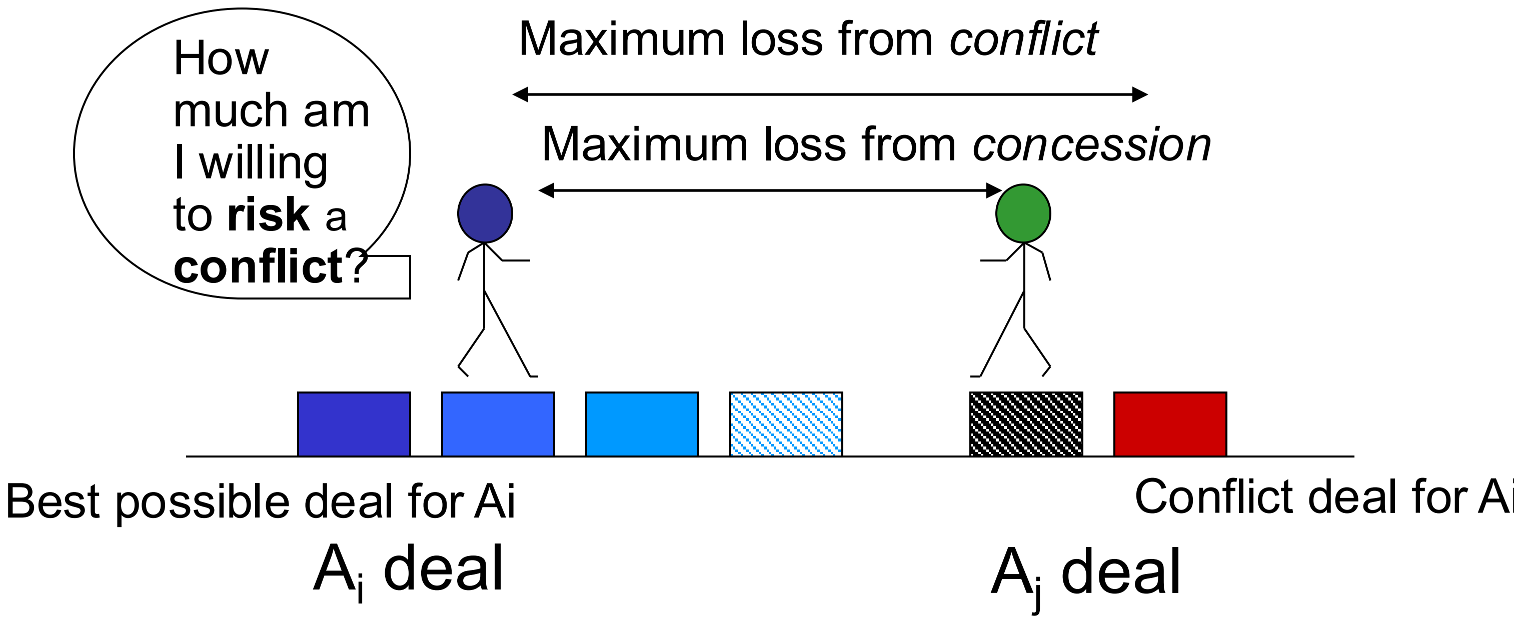
The Zeuthen Strategy

- Uses the risk evaluation strategy

- An agent will be *more willing* to risk conflict if the difference in utility between its current proposal and the conflict deal is *low*.

The Risk Factor

One way to think about which agent should concede is to consider how much each has to loose by running into conflict at that point.



How calculate risk?

Degree of willingness to risk a conflict can be defined as:

$$Risk_i^t = \frac{\text{utility } i \text{ loses by conceding and accepting } j\text{'s offer}}{\text{utility } i \text{ loses by not conceding and causing conflict}}$$

$$Risk_i = \begin{cases} 1 & \text{if } u_i(d_i) = 0 \\ \dfrac{u_i(d_i) - u_i(d_j)}{u_i(d_i)} \end{cases}$$

Key Questions (MCP)

1. What should an agent's first proposal be?

   It's most preferred deal.

2. On any given round, who should concede?

   The agent least willing to risk conflict.

3. If an agent concedes, then *how much* should it *concede*?

   Just enough to change the balance of risk.

MCP (continue)

- on each step calculate own risk and risk of opponent
- if risk is smaller to opponent then make an offer which involves minimal sufficient concession (if risk is equal to opponent then flip a coin who should conside)
  - sufficient - change the balance of risk between you and opponent (after this his risk is smaller)
  - minimal sufficient - gives opponent the least utility
- otherwise, offer the same deal that you offered previously

About MCP and Zeuthen Strategies

- **Advantages**:
  - Simple and reflects the way human negotiations work.
  - Stability – in Nash equilibrium – if one agent is using the strategy, then the other can do no better than using it him/herself.
  - The protocol doesn't guarantee success but guarantees termination
  - It doesn‘t guarantee to maximize social welfare but if agreement is reached then it is Pareto Optimal
- **Disadvantages**:
  - Assumes that both agents are rational and work on maximizing their utilities

Let's play a little game…..

Guess half the average

- Choose a number between 0 and 100.
  Your aim is to choose a number that is closest to half the average of the numbers chosen by all the students.
- What is your number?

Negotiation Domains: Task-oriented

- ”Domains in which an agent’s activity can be defined in terms of a set of tasks that it has to achieve”, *(Rosenschein & Zlotkin, 1994)*
    - An agent can carry out the tasks without interference from other agents
    - All resources are available to the agent
    - Tasks redistributed for the benefit of all agents

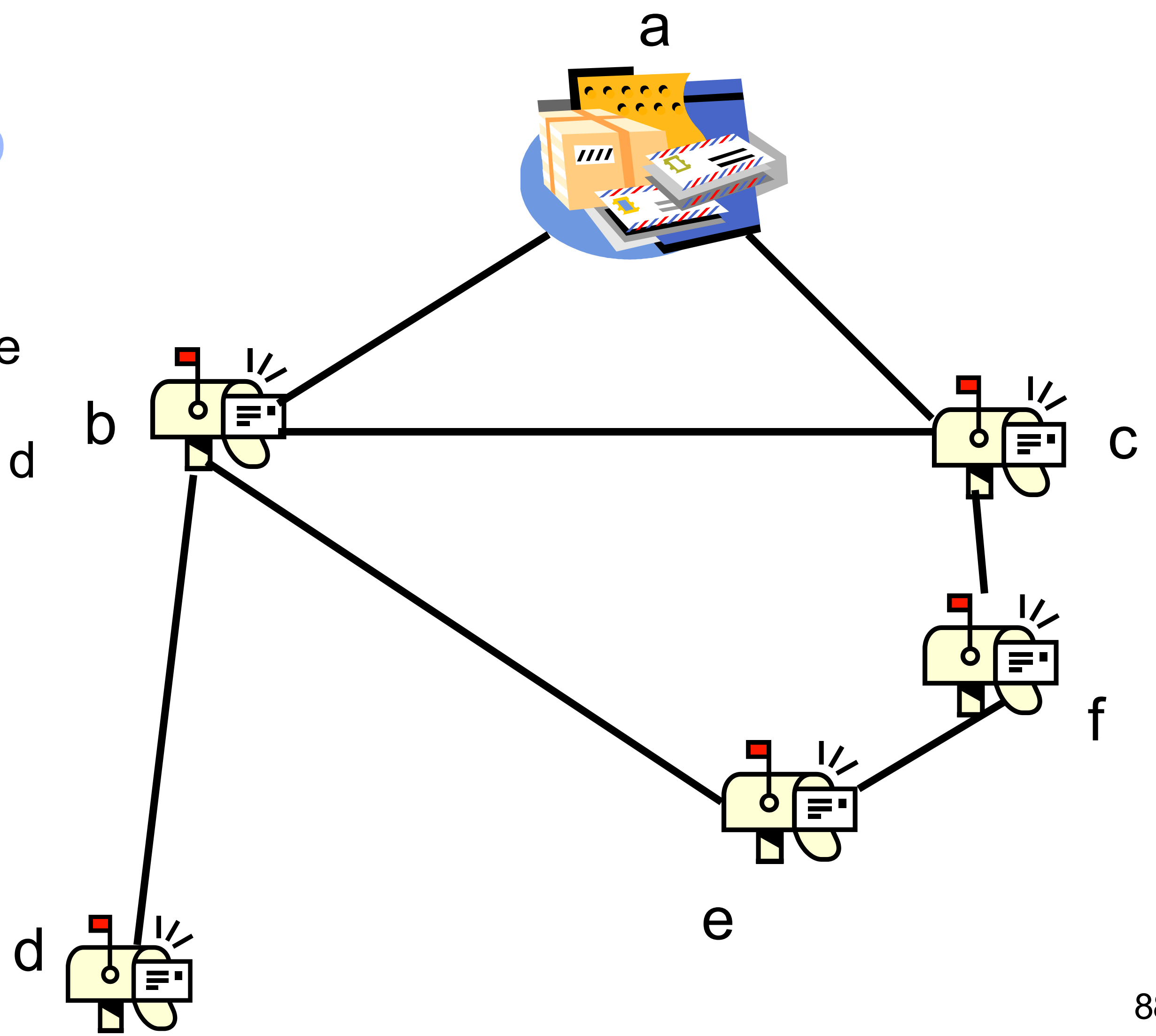
Task-oriented Domain: Example

Two postman arrive at the post-office early in the morning; they receive sacks of letters that they have to deliver around the city, which can be represented by a graph. At each node there is a little mailbox. The cost of carrying out a delivery is only in the travel distance. The is no limit to the number of letters they deliver and the number of letters they can put into a mailbox. If they have the same addresses for mail delivery they can negotiate about division of tasks

Postman domain

Agent1 b, c, f, e

Agent2 b, c, e, d



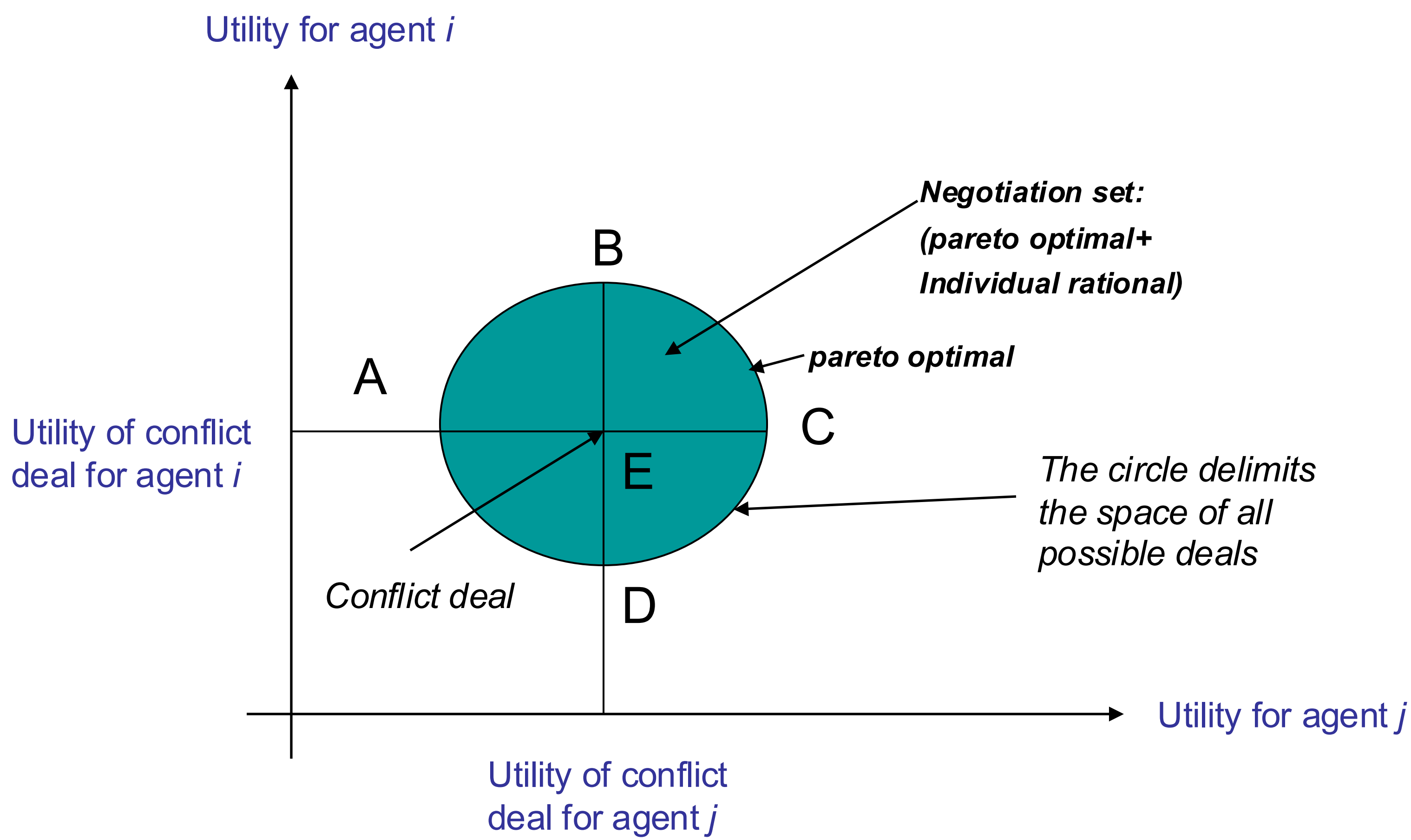
Task-oriented Domain: Definition

- Can be defined as a triple $\langle T, Ag, c \rangle$
  - *T*: finite set of all possible tasks
  - *Ag*: set of negotiating agents
  - *C*: cost of executing each subset of tasks
- How can an agent evaluate the utility of a specific deal?
  - Utility represents how much an agent has to gain from the deal.
  - Since an agent can achieve the goal on its own, it can compare the cost of achieving the goal on its own to the cost of its part of the deal.
    - If $utility_{alone} < utility_{deal}$, it is better to make deal.

Deals in Task-oriented Domains

- Conflict deal: if agents fail to reach an agreement:
  - where no agent has to execute tasks other than its own.
- A deal dominates if it is at least as good for every agent as another deal and is better for some agent than another deal. A deal is weakly dominating if at least the first condition holds.
- A deal that is not dominated by any other deal is pareto optimal.
- A deal is individually rational if it weakly dominates the conflict deal.

Negotiation Set in Task-Oriented Domains

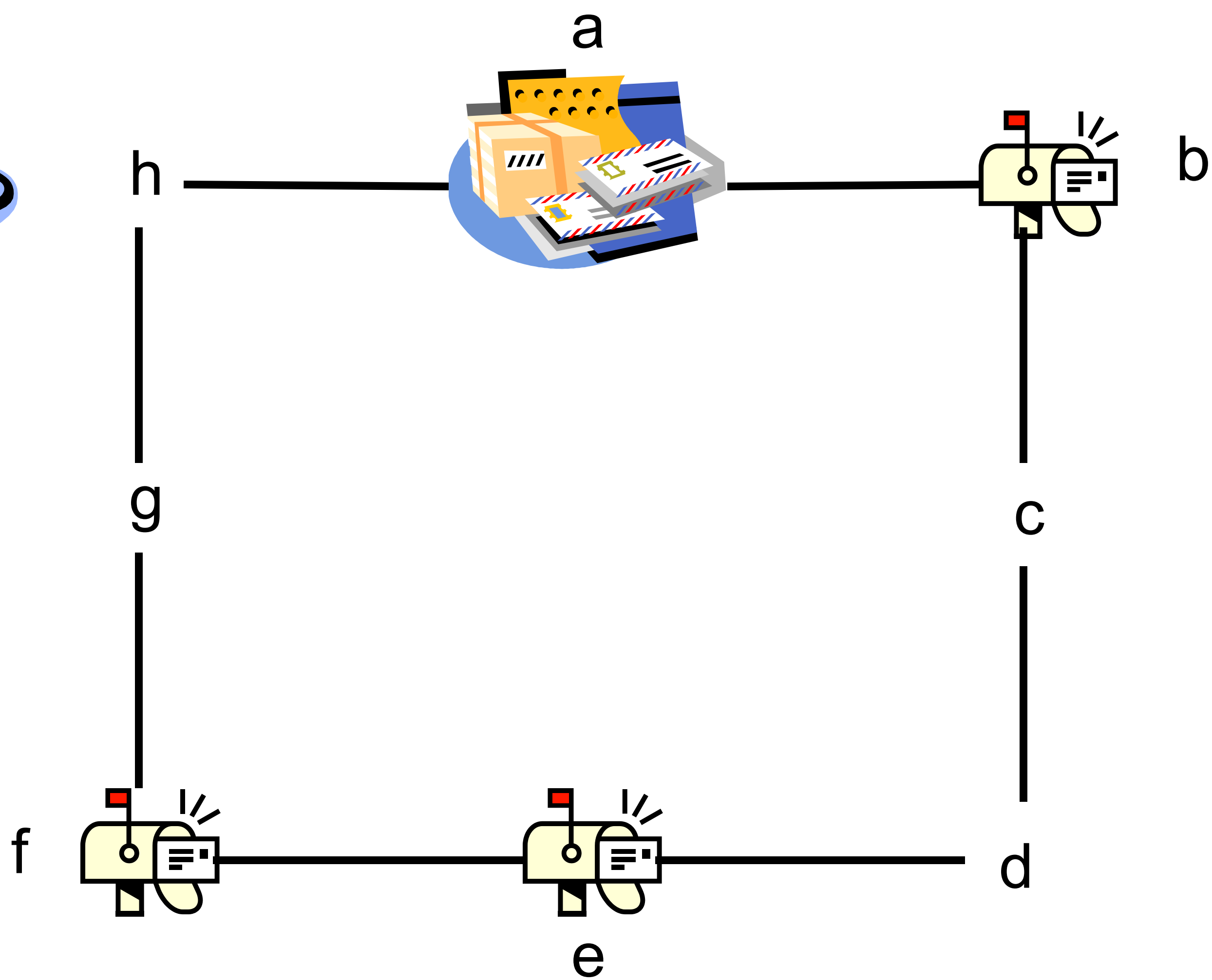


Adopted from M. Wooldriidge

Postman domain (broadcast task)

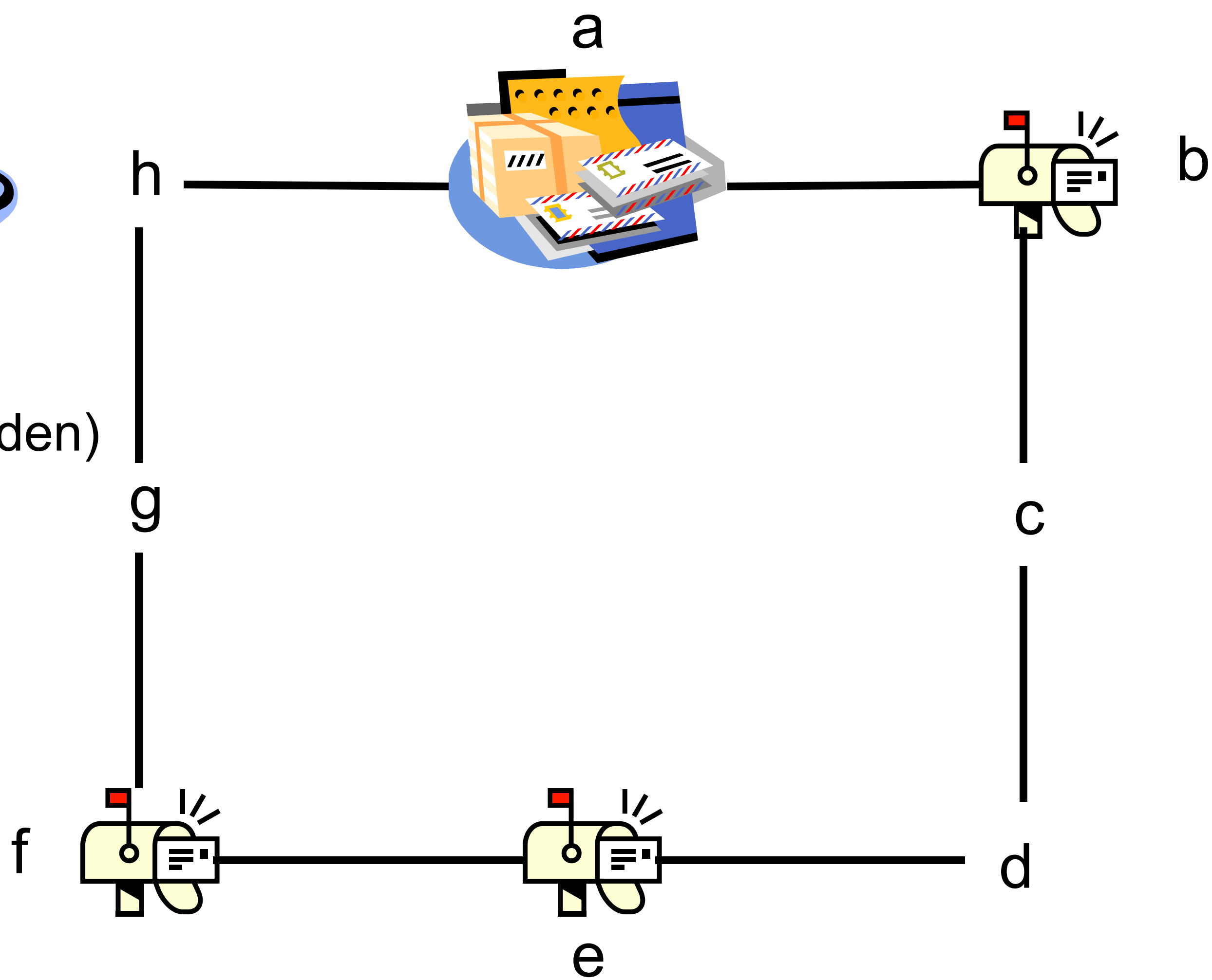
Agent1 b, f

Agent2 e



Agents will flip a coin to decide who delivers the letters

Postman domain (hidden letters)



Agent1 f (b hidden)

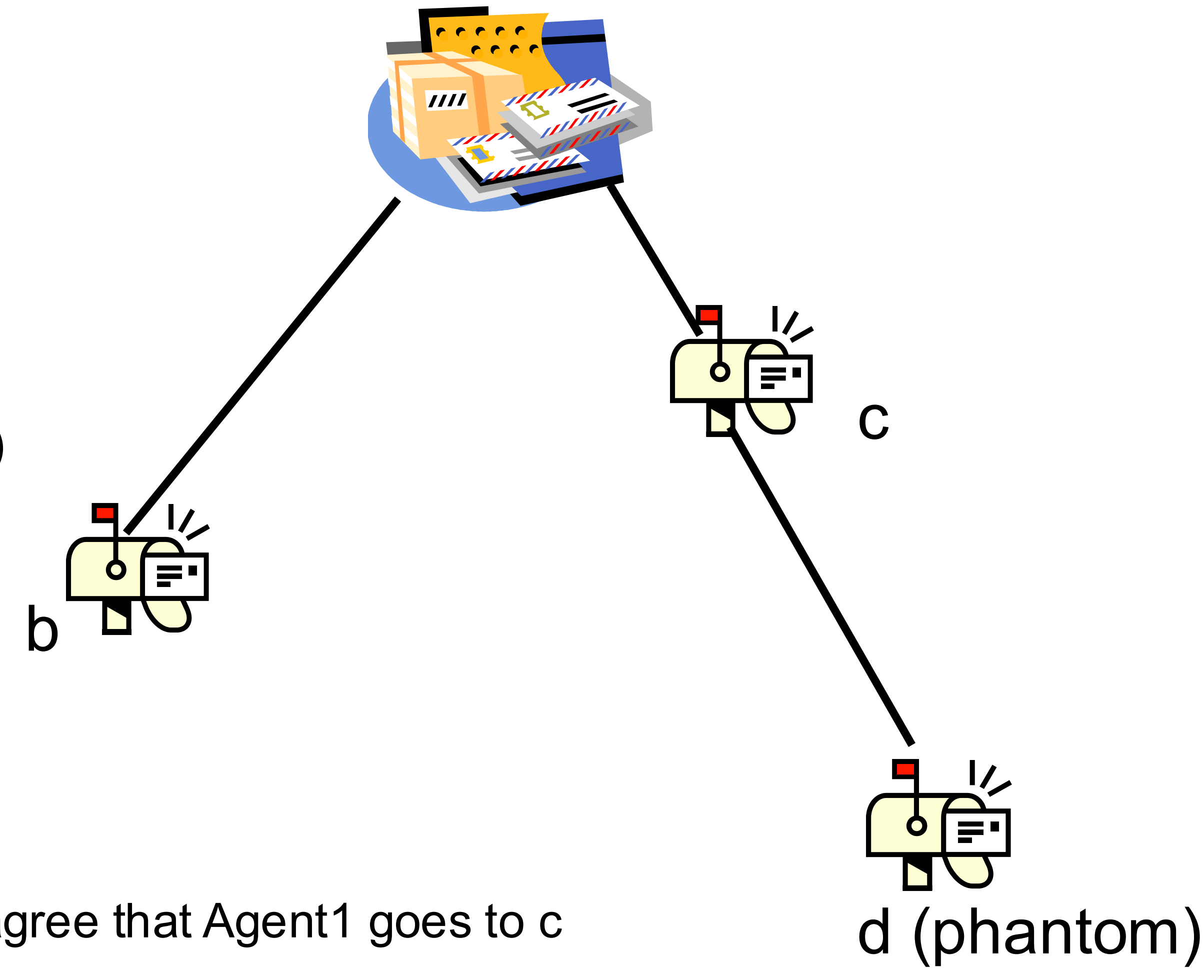
Agent2 e

Agents then agree that Agent2 delivers to f and e

Postman domain (Phantom letters)

Agent1 b, c
(d phantom)

Agent2 b, c

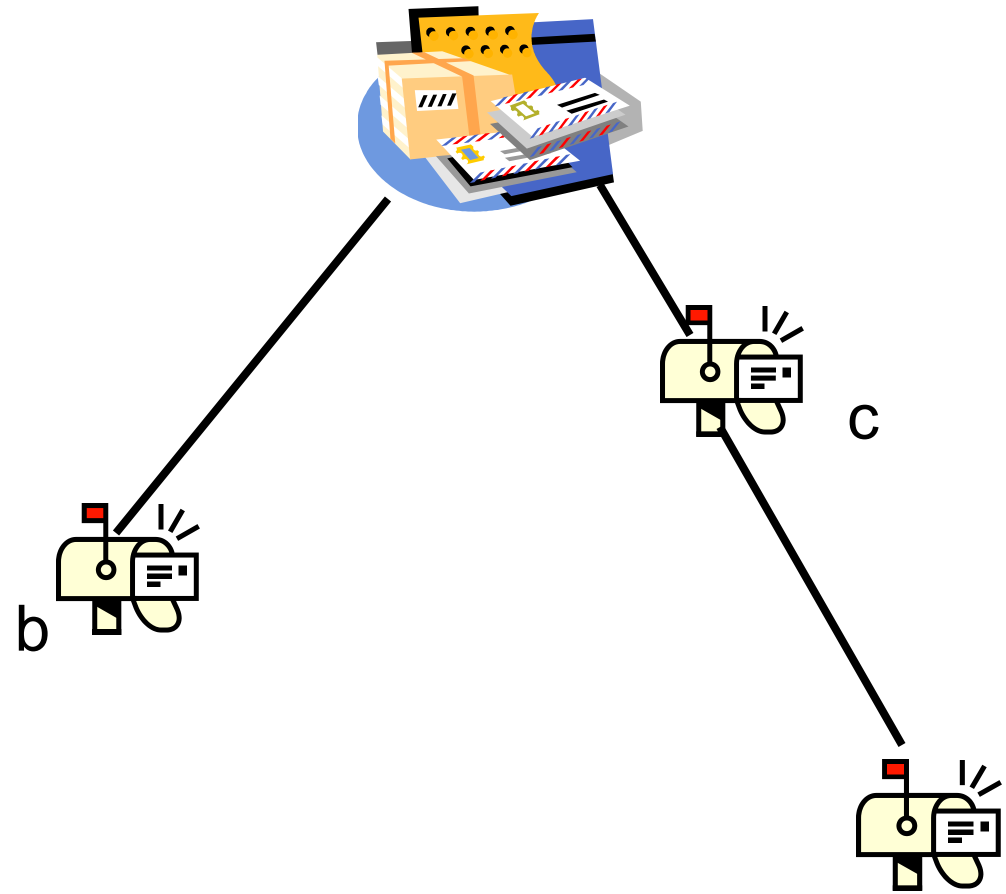


Agents then agree that Agent1 goes to c

Postman domain (decoy)

Agent1 b, c
(d decoy)

Agent2 b, c



Task d can be generated on demand if necessary

Postman domain (how to allocate deals)

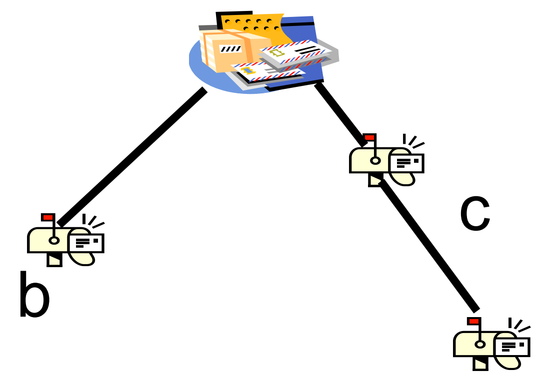
- **pure deals** - agents are deterministically allocated exhaustive disjoint task set
- **mixed deals** - specify a probability distribution over partitions
- **all-or-nothing deals** - mixed deals where the alternatives only include partitions where one agent handles the tasks of all agents

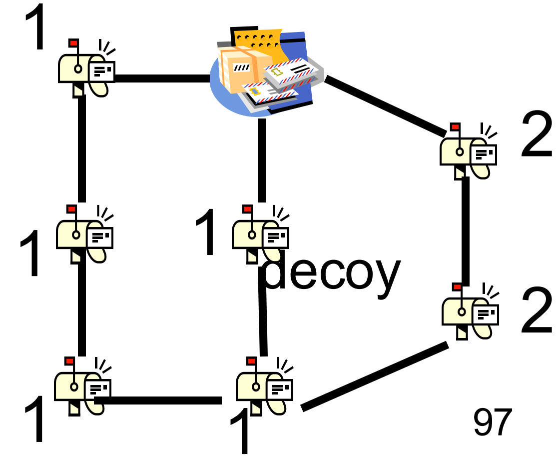
Postman domain

Using probabilities and all-or-nothing deal agreement helps to get rid of deceptions

- probability is assigned to each Agent task
- all-or-nothing deal is applied
- weighted coin (with probabilities) is used
- in case of phantom and decoy letters Agent1 can have a bigger probability to deliver all letters (and there is no point for lying)

- however, it is possible that decoy letter doesn't increase probability to deliver all letters in some topology





Postman domain

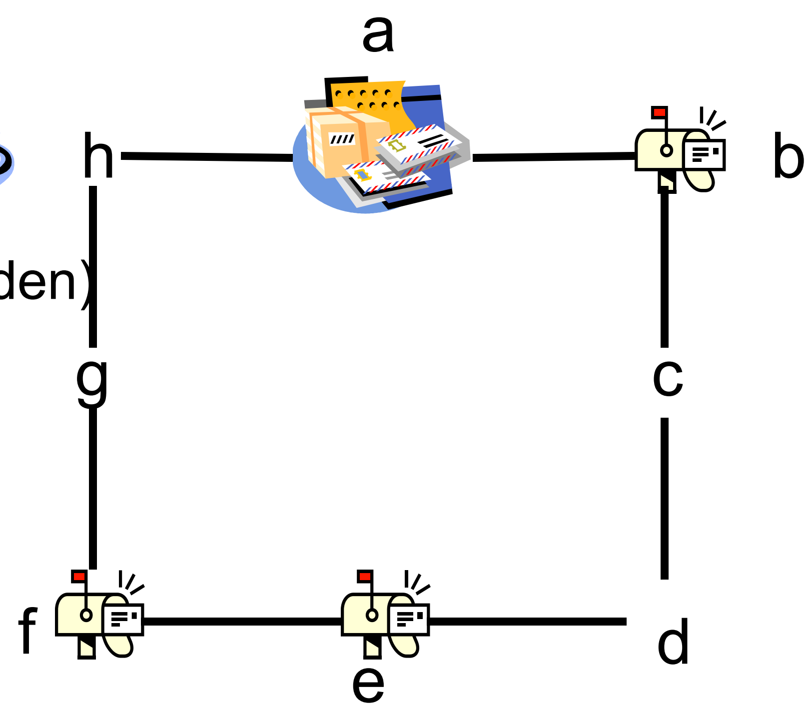
- in case of hidden letter Agent1 still has probability to deliver all letters (in any case it still has a guaranteed trip to b)

expected cost =
  (3/8)*8 + (5/8)*2 = 4.25

utility = 8 - 4.25 = 3.75

Agent1 f (b hidden)

Agent2 e



in case of non-hiding letter
  expected cost = 0.5*8 = 4

and utility = 8 - 4 = 4

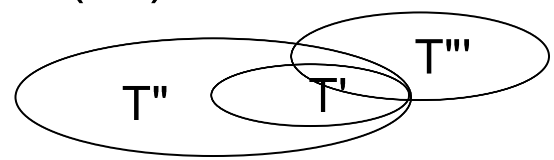
Task-Oriented domain (subtypes)

- Task Oriented Domains (TODs) - assume that agents have symmetric cost functions $ci(T') = cj(T')$ and that every agent is capable of handling tasks of all agents
- Subadditive TODs (STODs) are TODs, where

$$ci(T' \cup T'') \le ci(T') + ci(T'')$$

- Concave TODs (CTODs) are Subadditive TODs, where

$$ci(T' \cup T''') - ci(T') \ge ci(T'' \cup T''') - ci(T'') \qquad \text{and } T' \subset T''$$



- Modular TODs (MTODs) are Concave TODs where

$$ci(T' \cup T'') = ci(T') + ci(T'') - ci(T' \cap T'')$$

	TOD			STOD			CTOD			MTOD		
	Hid	**Pha**	**Dec**	**Hid**	**Pha**	**Dec**	**Hid**	**Pha**	**Dec**	**Hid**	**Pha**	**Dec**
Pure	**L**	**L**	**L**	**L**	**L**	**L**	**L**	**L**	**L**	**L**		
Mixed	**L**		**L**	**L**		**L**	**L**			**L**		
All-or-Nothing	-	-	-			**L**						

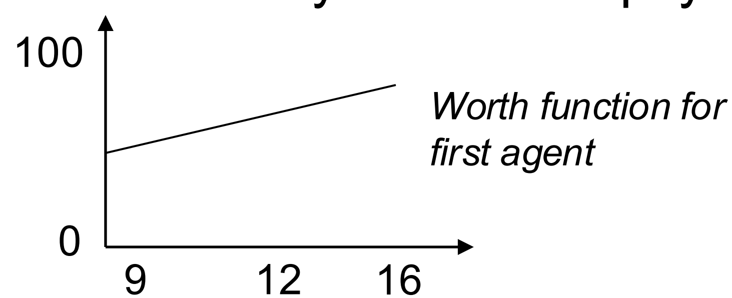
Negotiation Domains: Worth-Oriented Domains

- ”Domains where agents assign a worth to each potential state (of the environment), which captures its desirability for the agent”, *(Rosenschein & Zlotkin, 1994)*
  - agent’s goal is to bring about the state of the environment with highest value
  - we assume that the desirable environment state can be achieved only by a joint plan executed by several different agents

Worth-Oriented Domain: Example

2 agents are trying to set up a meeting. The first agent wishes to meet later in the day while the second wishes to meet earlier in the day. Both prefer today to tomorrow. While the first agent assigns highest worth to a meeting at 16:00hrs, s/he also assigns progressively smaller worths to a meeting at 15:00hrs, 14:00hrs….

By showing flexibility and accepting a sub-optimal time, an agent can accept a lower worth which may have other payoffs, (e.g. reduced travel costs).



Ref: Rosenschein & Zlotkin, 1994

Worth-Oriented Domain: Definition

- Can be defined as a tuple:
  - $\langle E, Ag, J, c \rangle$
    - E: set of possible environment states
    - Ag: set of possible agents
    - J: set of possible joint plans
    - $c(j,i)$: cost for agent i of executing the plan j
- $W(e,i)$ – represents the value, or worth, to agent i in state e

Worth-Oriented Domains

- Unlike Task-Oriented Domains, agents negotiating over Worth-Oriented Domains are not negotiating about a single issue:
  - they are negotiating over both the state that they wish to bring about (which has a different value for different agent), and
  - over the means by which they will reach the state.

Worth-Oriented Domains

- Multiple set of attributes
  - If you want to pay for some software, then you might consider several attributes of the software such as the price, quality and support.
  - You may be willing to pay more if the quality is above a given limit, i.e. you can't get it cheaper without compromising on quality.

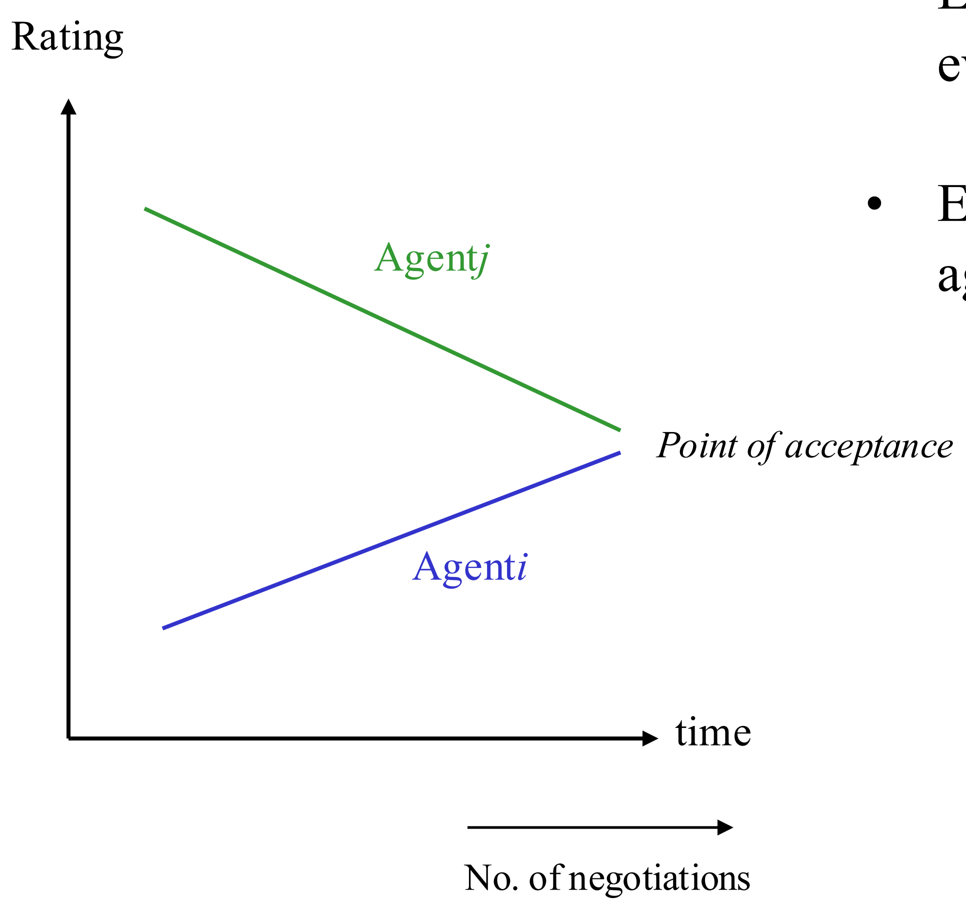
➢ Pareto Optimal – Need to find the price for acceptable quality and support (without compromising on some attributes).

How can we calculate Utility?

- Weighting each attribute
  - Utility = {Price*60 + quality*15 + support*25}

- Rating each attribute value
  - Price :(max 70); 300000$: 0 ; 310000$: 45; 320000$: 70

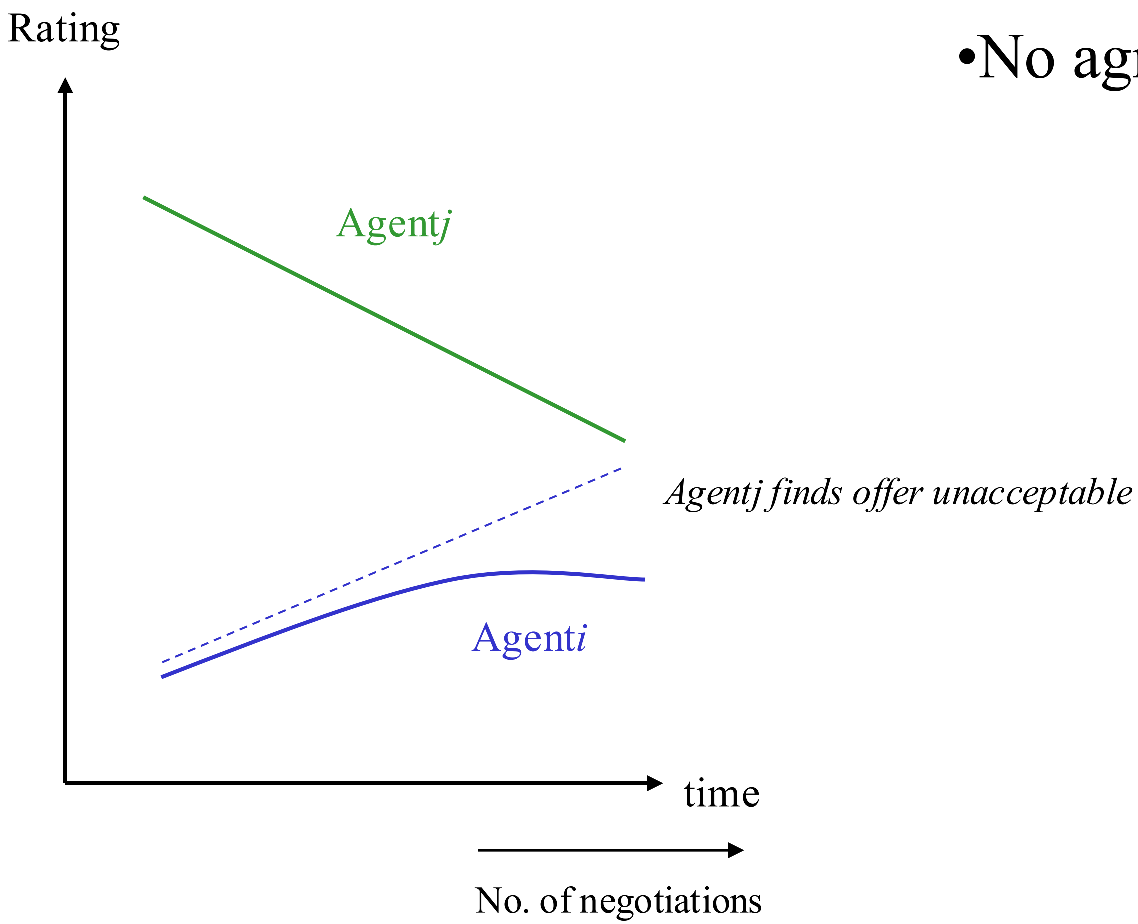
Utility Graphs 1



- Each agent concedes in every round of negotiation
- Eventually reach an agreement

Utility Graphs 2

- No agreement



Example (Negotiations between Misty and Smiley)

- Both Misty and Smiley have carefully read the information about their respective organizations to understand the problem and its issues. Since each negotiator can access the system and make a proposal independently, we will follow Misty's side of the negotiation.
- There are only two *issues* in this simple negotiation: the *price* of the aircraft and the terms of the *warranty*. It has been established that the normal price of this aircraft is in the range of \$300 000 to \$320 000. The sensible increase is of \$10 000. Thus, the price *options* are \$300 000, \$310 000, and \$320 000. In this industry there are four types of warranty typically available. The options are: no warranty, a 6 months, one year, and a 2 years warranty.
- Both negotiators analyze the two issues and their associated options in terms of their relevance to their respective organizations and move to the pre-negotiation phase.

From http://www.interneg.org/inspire/doc/demo.html

Example (Preparation)

Issue rating

- To prepare for the negotiations Misty and Smiley each rated the two issues. Note, that the pre-negotiation steps are conducted independently; one negotiator can *never* see the information (ratings) that the other negotiator enters.
- Misty feels that price is far more important than warranty. Therefore, she assigns 70 points to price and 30 to warranty. Although Misty does not know it, Smiley feels that each issue is equally important and so Smiley assigns 50 points to each.

Misty's Issue Ratings

Negotiation	**Issue Rating**
Price	70
Warranty	30

Smiley's Issue Ratings (Misty does not see this)

Negotiation	**Issue Rating**
Price	50
Warranty	50

From http://www.interneg.org/inspire/doc/demo.html

Example (Preparation)

Option rating

- Each issue has one or more options, for example, price has three options: 300 000 $, 310000 $, 320 000 $. After rating the issues, the options in each issue must also be rated similarly. For each issue at least one option must be assigned the maximum rating for the issue and at least one option must be assigned a rating of zero.
- Misty considers the three options for the price of the aircraft and assigns the maximum rating (that is, 70) to the price of $320 000 because Misty represents Rosa which wants to sell the aircraft. The lowest possible price is assigned a rating of zero. Misty considers the price of $ 310 000 as somewhat acceptable and assigns a rating of 45.
- Misty assigns ratings to each warranty option in a similar way. Note that for Misty, "no warranty" has the same maximum rating (30) as the 6 months warranty perhaps because the organization does not think it is possible that this plane will fail during the first 6 months of operation.

Price	**Rating (Max = 70)**	**Warranty**	**Rating (Max = 30)**
300 000 $	**0**	No warranty	30
310 000 $	45	6 months	30
320 000 $	70	One year	10
		Two years	0

From http://www.interneg.org/inspire/doc/demo.html

Example (Preparation)

Package evaluation

- Given user's ratings for each issue and each option, system calculates ratings for complete packages that are the subject of negotiations. A package consists of price option and warranty option, for example, "320 000 $ and No warranty" is one complete package. The system presents few packages and their ratings so that the user can assess if the system's results accurately describe the user's preferences.

Misty's evaluations of four packages are below:

Price	**Warranty**	**Rating**
320 000 $	No warranty	100
320 000 $	One year	80
310 000 $	6 months	75
320 000 $	Two years	70

From http://www.interneg.org/inspire/doc/demo.html

Example (Preparation)

The ratings indicate how good the packages are given Misty's ratings of the issues and options. Comparing the packages Misty decided to change two ratings, for the second and fourth package. The second package while worse than the first (best) package is still quite good, so Misty increases its rating from 80 to 82. The fourth package is not nearly as good as the third package and Misty downgrades its rating to 69.

Price	Warranty	Rating
320 000 $	No warranty	100
320 000 $	One year	82
310 000 $	6 months	75
320 000 $	Two years	69

The system uses now the issue, option and package ratings and it determines the utility function which closely reflects Misty's preferences. These function will be used to provide a rating for every considered package.

From http://www.interneg.org/inspire/doc/demo.html

Example (Negotiation)

Exchange of offers and messages

- Misty thought a while on how to begin negotiations. In the meantime Smiley prepared and sent the following offer together with a short message:

Smiley's opening offer and message

Price	300 000 $
Warranty	Two years
Misty's rating:	0

From http://www.interneg.org/inspire/doc/demo.html

Example (Negotiation)

Misty considers Smiley's offer and -- contrary to Smiley's expectations -- finds it unacceptable. Note that the offer's rating is 0 (Misty's rating: 0). Misty prepares a counter-offer which better reflects Rose Inc. requirements. A message is attached to Misty's offer.

Misty's response

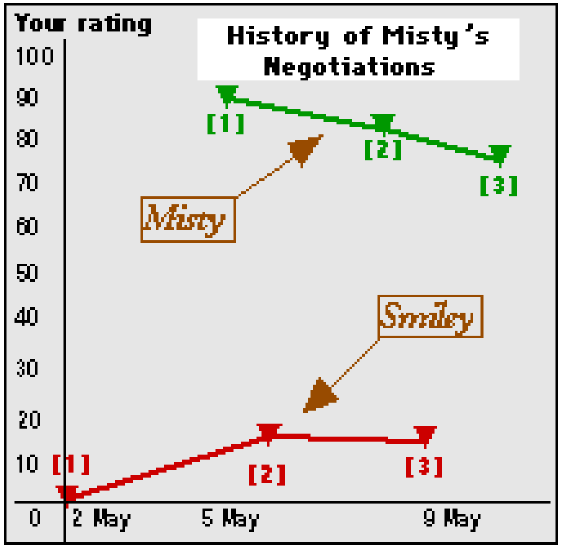
Price	320 000 $
Warranty	6 months
Misty's rating:	95

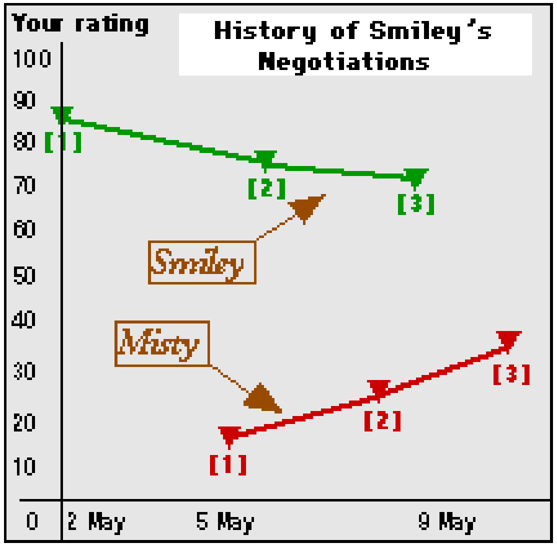
Note that the rating of the above offer is 95. From Misty's ratings of Price and Warranty it would appear that Mysty's offer "320 000 $ and 6 months" should be rated 100. However, the adjustments that Mysty made in the Package evaluation step caused a drop in rating to 95. Although these adjustments were made only to two packages they affect ratings of all possible packages because they modify Misty's utility function.

From http://www.interneg.org/inspire/doc/demo.html

Example (Negotiation)

At anytime during the negotiation, a graphical overview of the history of offers can be viewed. The graph plots the users' ratings of offers sent (in green) and received (in red) and the time that the offers were sent. The graph can be viewed from the "View offer and message history" link.





From http://www.interneg.org/inspire/doc/demo.html

Example (Negotiation)

The trade off

Misty and Smiley exchange the offers and messages until, in the fourth round, Misty presented the following offer:

Price	310 000 $
Warranty	One year
Misty's rating:	**55**

From http://www.interneg.org/inspire/doc/demo.html

Example (Negotiation)

Smiley reviews Misty's last offer which is shown below. (Note that the offer's rating now reflects Smiley's preferences).

Price	310 000 $
Warranty	One year
Smiley's rating:	53

After short consideration Smiley accepts Misty's last offer. Smiley does it by selecting, on the system page, the option:

Yes, I accept my counterpart's most recent offer as listed above.

From http://www.interneg.org/inspire/doc/demo.html

Example (Post-settlement)

Pareto Efficient packages

In some negotiations it may happen that the parties reach an agreement but there is one or more packages which are better than the accepted offer for *both* sides. Note, that *better* is measured with the parties utility functions. Thus, there may be a package for which the two ratings are higher than the package that has been accepted.

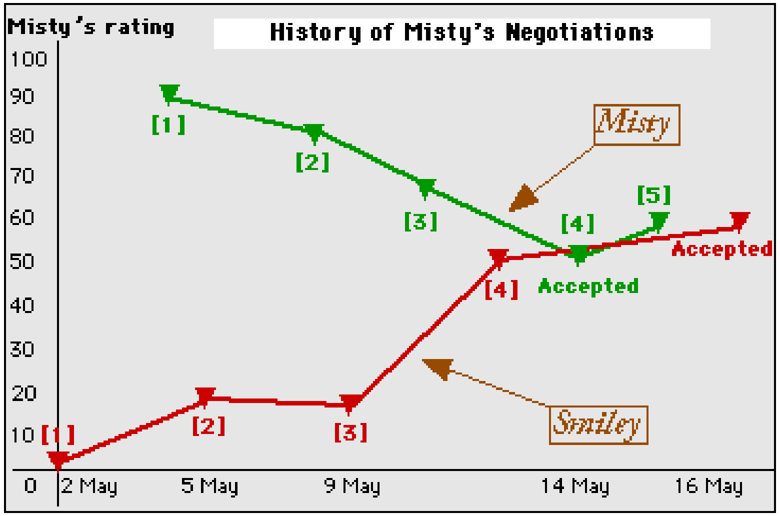
System may have a post-settlement stage, during which it uses the preference information provided by each user to determine whether it is possible to construct packages that are better for the two parties. In this negotiation system determined that Misty's and Smiley's settlement could be improved. In this simple negotiation there is only one such package:

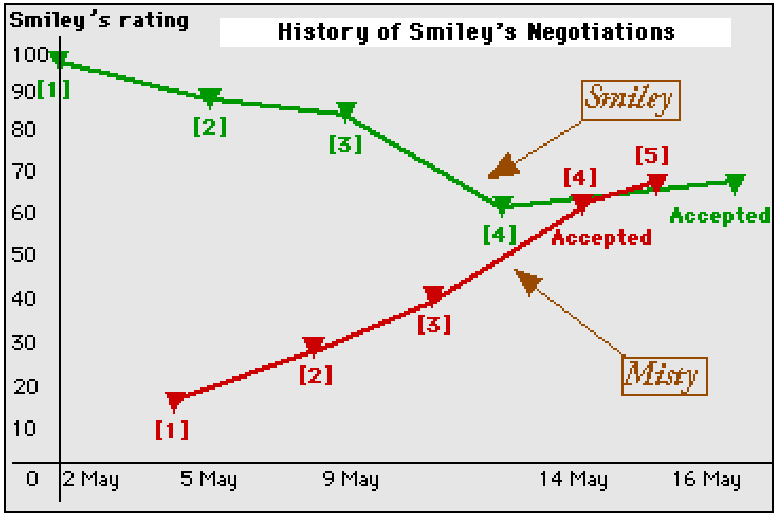
Price	320 000 $
Warranty	Two years

Misty asked Smiley about this new offer and after a short exchange they both agreed that this last package is superior to the compromise package. Thus the negotiation is completed.

From http://www.interneg.org/inspire/doc/demo.html

Negotiation graph





From http://www.interneg.org/inspire/doc/demo.html

Practical examples of negotiations (protocols and algorithms)

- Auctions
- Contract Net Protocol
- Application examples

Auctions

Several millions of $ paid for art at auction houses such as Sotheby's.


Sunflowers 1888
Courtesy of Van Gogh Museum
Vincent van Gogh Foundation

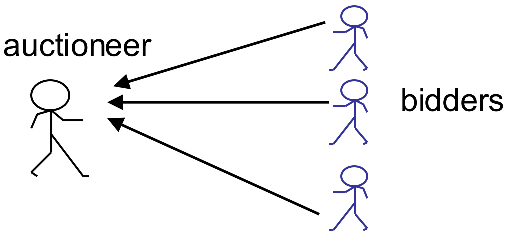
Auctions

Online Auctions

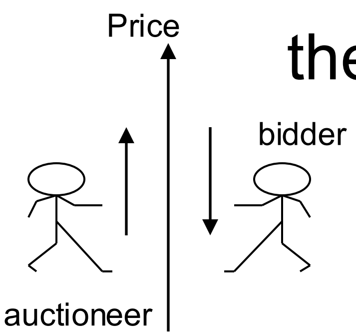
The World's Online Marketplace™

You want to buy some exciting video games. You see that there are some available on eBay. You register at eBay and offer a bid for some of these games.

Auctions



- An Auction takes place between an auctioneer and a collection of bidders.
- Goal is for the auctioneer to allocate the goods to one of the bidders.
- In most settings, the auctioneer desires to maximise the price; bidders desire to minimise the price.



Auctions

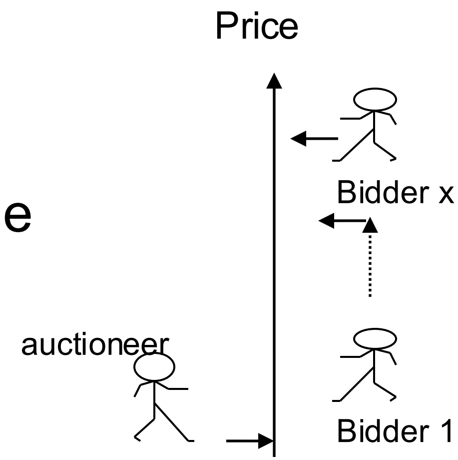
- private value auctions - the value of the good depends only on the agent's own preferences (no resale)
- common value auctions - an agent's value of an item depends entirely on other agent's values of it (resale)
- correlated value auctions - an agent's value depends partly on its own preferences and partly on other's values

Auction Parameters

Value of goods	Private, public/common, Correlated
Winner determination	First price, second price
Bids may be	Open cry, Sealed
Bidding may be	One shot, ascending, descending

English Auctions

- English auctions are:
  - First price
  - Open cry
  - Ascending
- The auctioner begins with the lowest acceptable price (sometimes without the lowest price)
- Bidders successively raise bids for item until single bidder remains
- Bidder may re-assess evaluation during auction
- Outcome
  - The item is ”knoked down” (sold to the highestbidder)



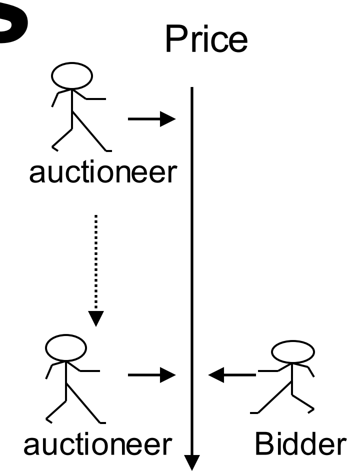
English Auctions

- Bid strategy: successively bid a small amount more than the highest current bid until it reaches your valuation, then withdraw.
- Susceptible to Winners curse
  - Winner is the one who overvalues the goods on offer and may end up paying more than its worth.

Japanese Auction

- Start price
- Each agent must choose whether or not to be in
- Auctioneer successively increases price
- After each increase an agent must say if he stays or not. When agent drops out it is irrevocable
- The auction ends when exactly one agent left
- The winner must buy the good for the current price

Dutch Auctions

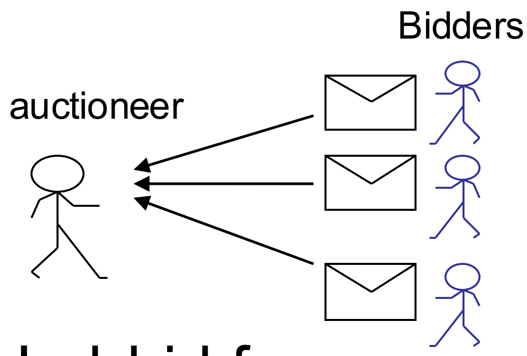


- Dutch auctions are:
  - Open cry
  - Descending
- Auctioneer starts at an artificially high price. Then continually lowers the offer price until an agent makes a bid which is equal to the current offer price.
- Outcome
  - The winner pays his/her price
  - When multiple units are auctioned the first winner takes his price and later winner pays less

Dutch Auctions

- Bid strategy: shade bid a bit below true willingness to pay
- Susceptible to Winners curse

First-price Sealed-bid Auctions



- One shot auction
- Single round, where bidders submit a sealed-bid for the good.
- Good is awarded to agent that made the highest bid.
- Winner pays price of highest bid. With more than one unit for sale, bids are sorted from high to low and items awarded at highest bid price until the supply is exhausted
- Best strategy: bid a bit less than true value.

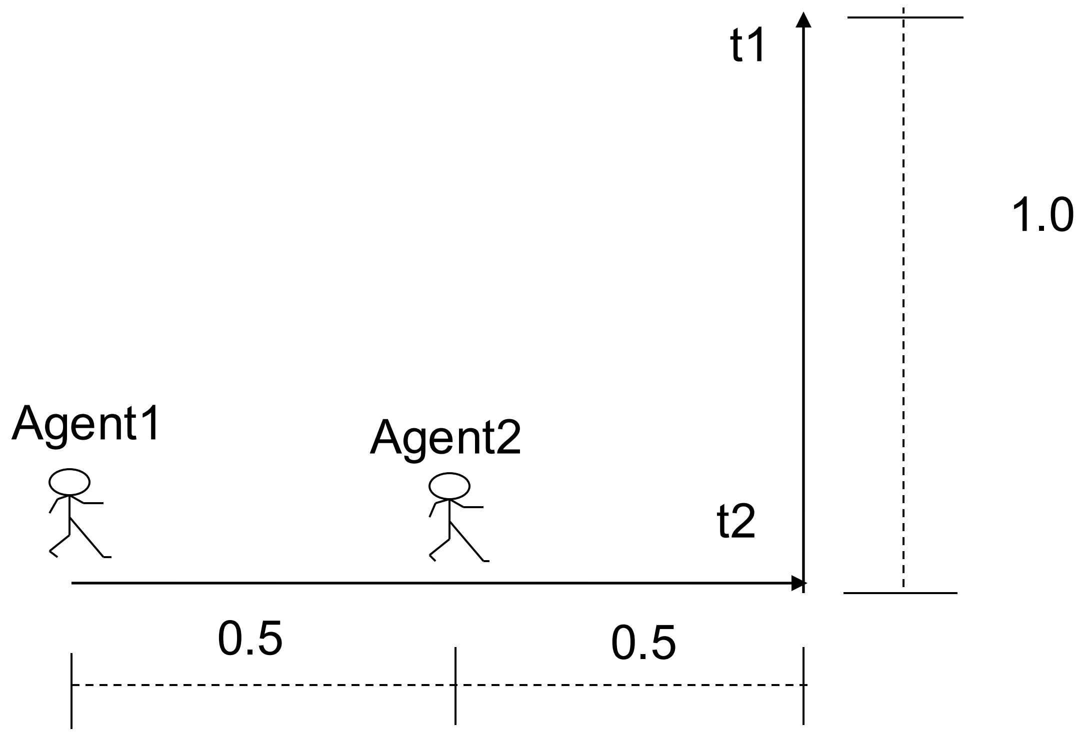
Vickrey Auctions

- Vickrey auctions are:
  - second-price
  - sealed-bids
- Good is awarded to agent that made the highest bid.
- Winner pays price of second highest bid.

Vickrey Auctions

- Best strategy:
  - bid the true value.
- Susceptible to anti-social behaviour

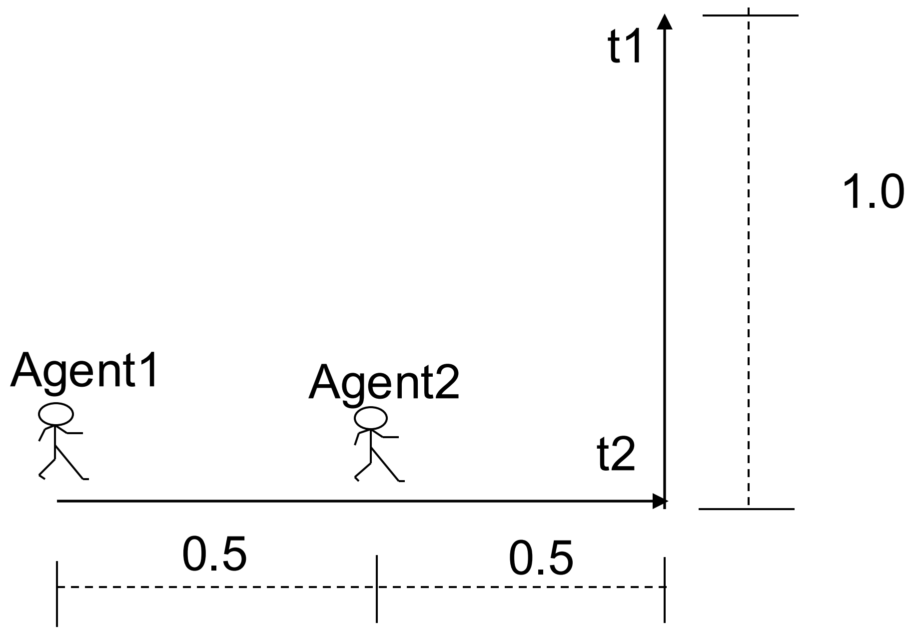
Interrelated auctions



The costs for handling tasks:

c1(t1)=2, c1(t2)=1, c1(t1,t2)=2,
c2(t1)=1.5, c2(t2)=1.5, c2(t1,t2)=2.5

Interrelated auctions



Truthful negotiation:

- Task 1: c1(t1)=2, c2(t1)=1.5
- Task 2: c1(t2)=1, c2(t2)=1.5

If tasks are considered independently then t1 will be allocated to Agent2 and t2 to Agent1

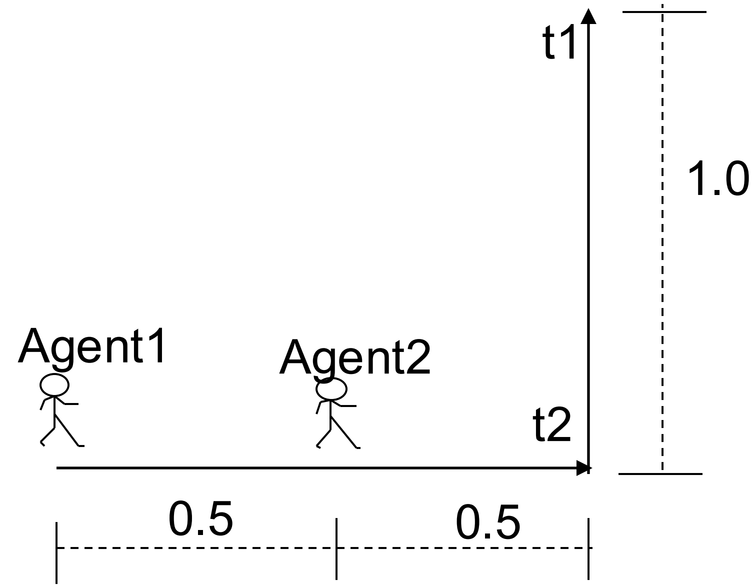
Interrelated auctions

Incorporation of full look ahead:

If Agent 1 has t1 it will bid
  c1(t1,t2) - c1(t1) = 2 – 2 = 0 and
  c1(t2)=1 otherwise

So if Agent 1 has t1 it will win t2 at the
  price 0 and gets payoff c1(t2) – 0 = 1

So Agent's 1 dominant strategy could be
  accommodating payoff into the first
  bid: c1(t1)-1=2-1=1 and it wins t1



The costs for handling tasks:

c1(t1)=2, c1(t2)=1, c1(t1,t2)=2,
c2(t1)=1.5, c2(t2)=1.5, c2(t1,t2)=2.5

Lies and Collusions

- All auction types can be manipulated
- Seller's perspective
  - revealing information removes uncertainty
- Lies:
  - By the bidders (e.g. in Vickrey auction)
  - By the auctioneer (shills, in sealed-bid auctions)

Collusion of bidders (Rings)

Coalition of bidders where they agree beforehand to put forward artificially low bids for the good on offer. When the good is obtained, the bidders can then get the true value of the good and share the profits.

- Ring with 5 persons appoint one to big for 50$ on a common auction
- Then then have internal auction with final price 100$
- 50$ return to the appointed person who pays 50$ on a common auction
- (100-50)/5 = 10$ pays to everybody in the Ring

A possible ranking auctions from most prone to collusions to least as follows: English, sealed-bids, Dutch

Limitations of Auctions

- Only concerned with the allocation of goods
- Not adequate for settling agreements that concerns matters of mutual interest

Auctions: Summary

- An Auction takes place between an auctioneer and a collection of bidders.
- In most settings, the auctioneer desires to maximise the price; bidders desire to minimise the price.
- Types of Auctions:
  - English auction
  - Japanese auction
  - Dutch auction
  - First-price sealed bids
  - Vickrey (Second-price sealed bids)
- Useful for allocating goods. But too simple for many other settings.

Basic idea of CNP (Contract Net Protocol)

The basic assumption of the CNP is that, if an agent cannot solve an assigned problem using local resources/expertise, it will decompose the problem into subproblems and try to find other willing agents with the necessary resource/expertise to solve these subproblems

Contract Net Protocol

Cooperating experts metaphor

- work independently
- exchange results
- ask help when enable to solve individually

Contract Net Protocol

- Problem is too large or outside of expertise.
  - If large then partition it and find other experts who can solve the tasks
  - If outside of expertise then also find another expert
- If expert is known then contact them directly
- If not then describe a task to entire group
- If somebody from the group agrees then s/he notifies you
- If more then one agree then choose one

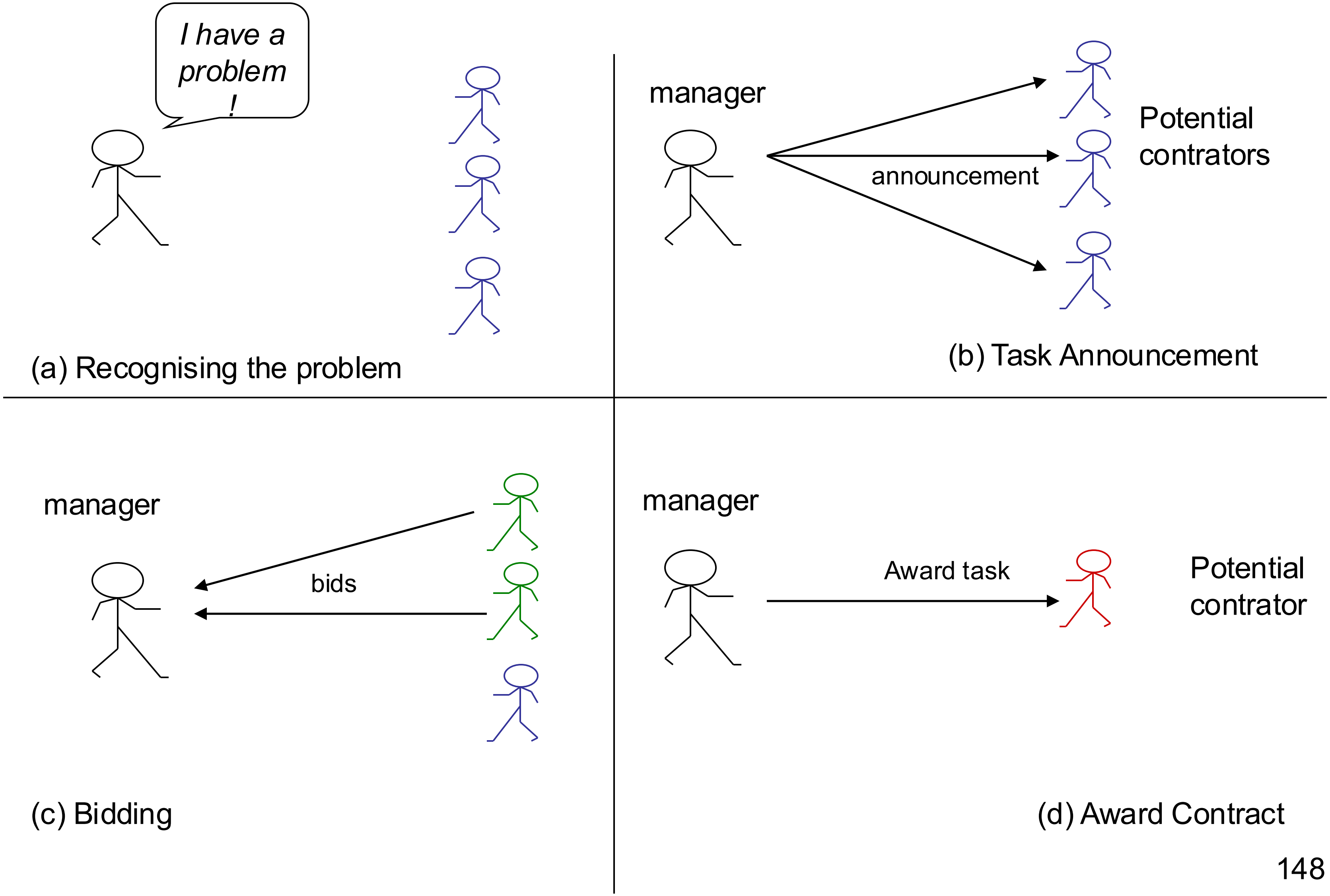
Basic notions of CNP

Each agent can take two roles:

- manager
- contractor

- contracting mechanism
  - contract announcement by the manager agent
  - submission of bids by contracting agents in response to the announcement
  - evaluation of submitted bids by the manager
  - awarding a subproblem contract to the contractor with the most appropriate bid(s)

The Contract Net Protocol



Marginal cost (Sandholm)

- Let us
  - τ_i^t is a set of tasks allocated to the agent *i* at time *t*
  - e_i are resources available to *i*
  - $\tau(ts)$ is announced task
- Marginal cost
  - $\mu_i(\tau(ts)|\tau_i^t) = c_i(\tau(ts)\ U\ \tau_i^t) - c_i(\tau_i^t)$
- Decision to bid:
  - $\mu_i(\tau(ts)|\tau_i^t) < (e(ts) + e_i)$

  where $e(ts)$ is reward for doing the new task

Disadvantages of CNP

- doesn't detect conflicts
- assumes benevolent and non-antagonistic agents
- can be rather communication intensive

CNP example and extension

MACIV - a distributed system for resource managing in a building construction company (Foseca et al)

Tasks:

- excavation,
- land movement etc.

that executed by:

- isolated machine
- groups of machines

Very often an announcing agent can't wait indefinitely for the best solution

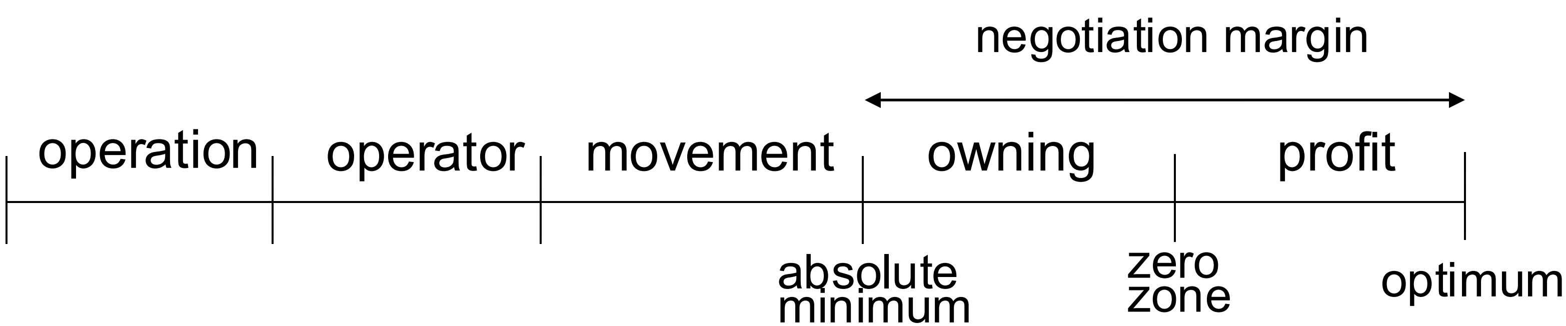
More details about the example

CNP is used - announcing, bidding and awarding

Each agent estimates the cost for execution based on the following factors:

- travel cost
- depreciation cost
- operation cost
- operator cost
- profits

Where does negotiation come?



Negotiation algorithm

1. CNP is used - announcing, bidding and awarding
2. announcing
3. task evaluation
4. selection phase
5. market manipulation
6. price adjusting

4 and 5 are repeated iteratively

Forming coalitions

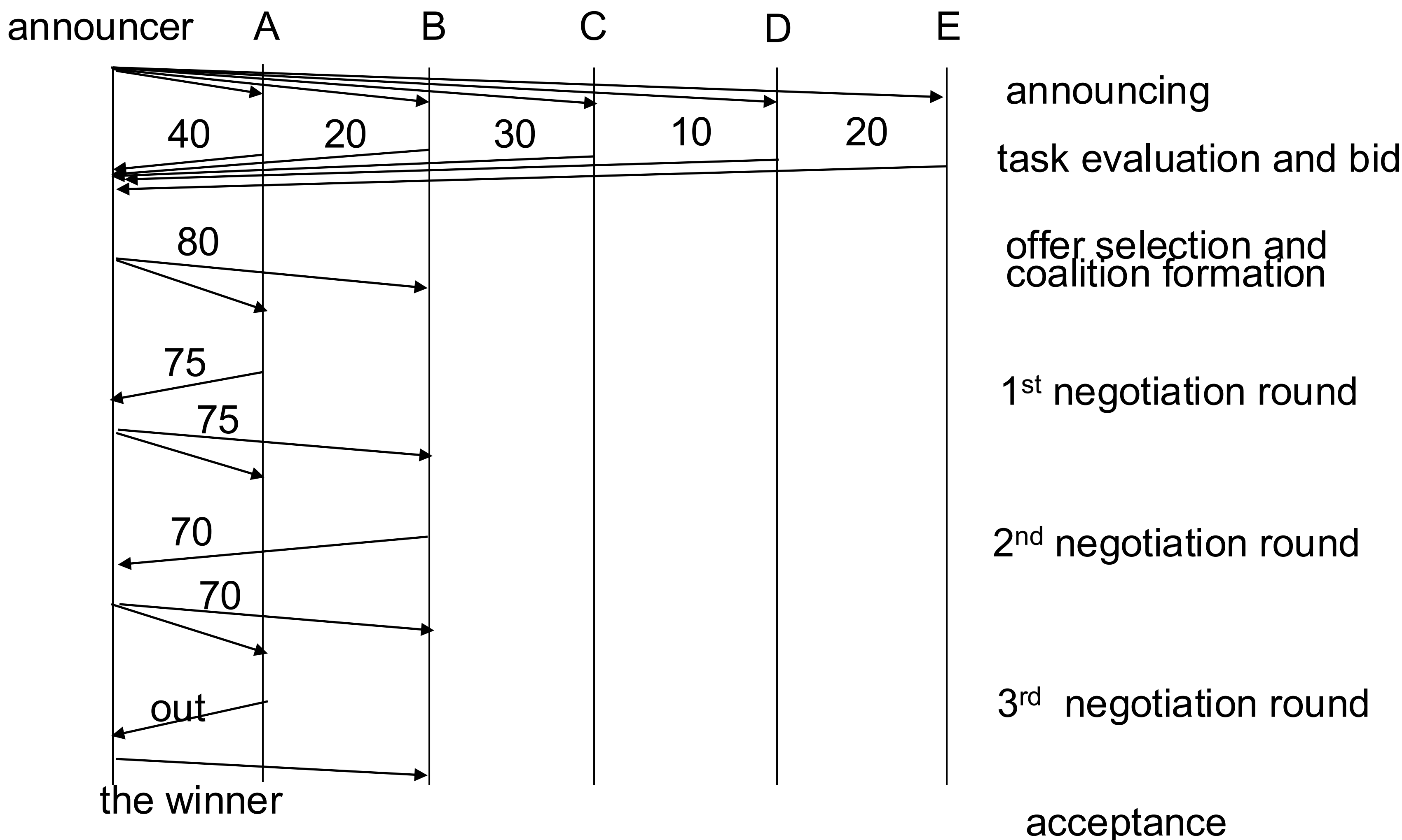
Steps 4-5 are rewritten as follows

3. selection phase - as previous but manager also calculates possible coalitions
4. market manipulation - manager sends to the coordinator of the coalition the best price achievable on this stage
5. price adjusting - coordinator negotiates with each partners and if possible to achieve lower price then sends it further otherwise retires
6. price selection - the announcer evaluates the best offer and communicates the new best to all coalitions

5 and 6 repeated until just one coalition remains

Example

Coalitions:
C1 = (**A**,B,C,D)
C2 = (**B**,C,D,E)

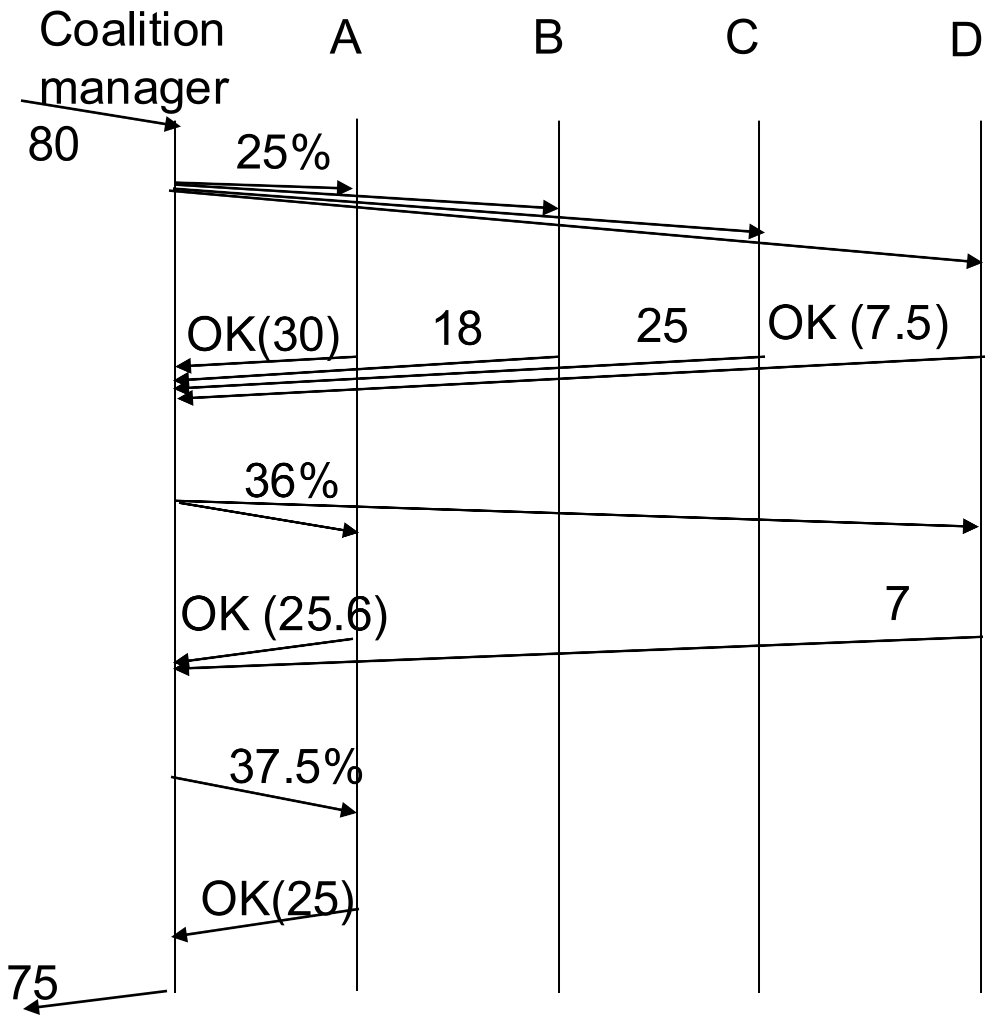


The intra-coalition negotiation

1. coordinator calculates the percentage of the initial cost that coalition must decrease
2. announcing percentage
3. agents respond by accept or reject (inform about lowest acceptable price)
4. if there are rejects then new percentage calculation and goto 2
5. coordinator informs the announcer about new coalition position

Example (cont.)

C1 = (**A**,B,C,D) 40 20 30 10



coordinator receives new lower price

coordinator announces decreasing percentage

A and D accept but B and C refuse

coordinator announces new percentage for A and D

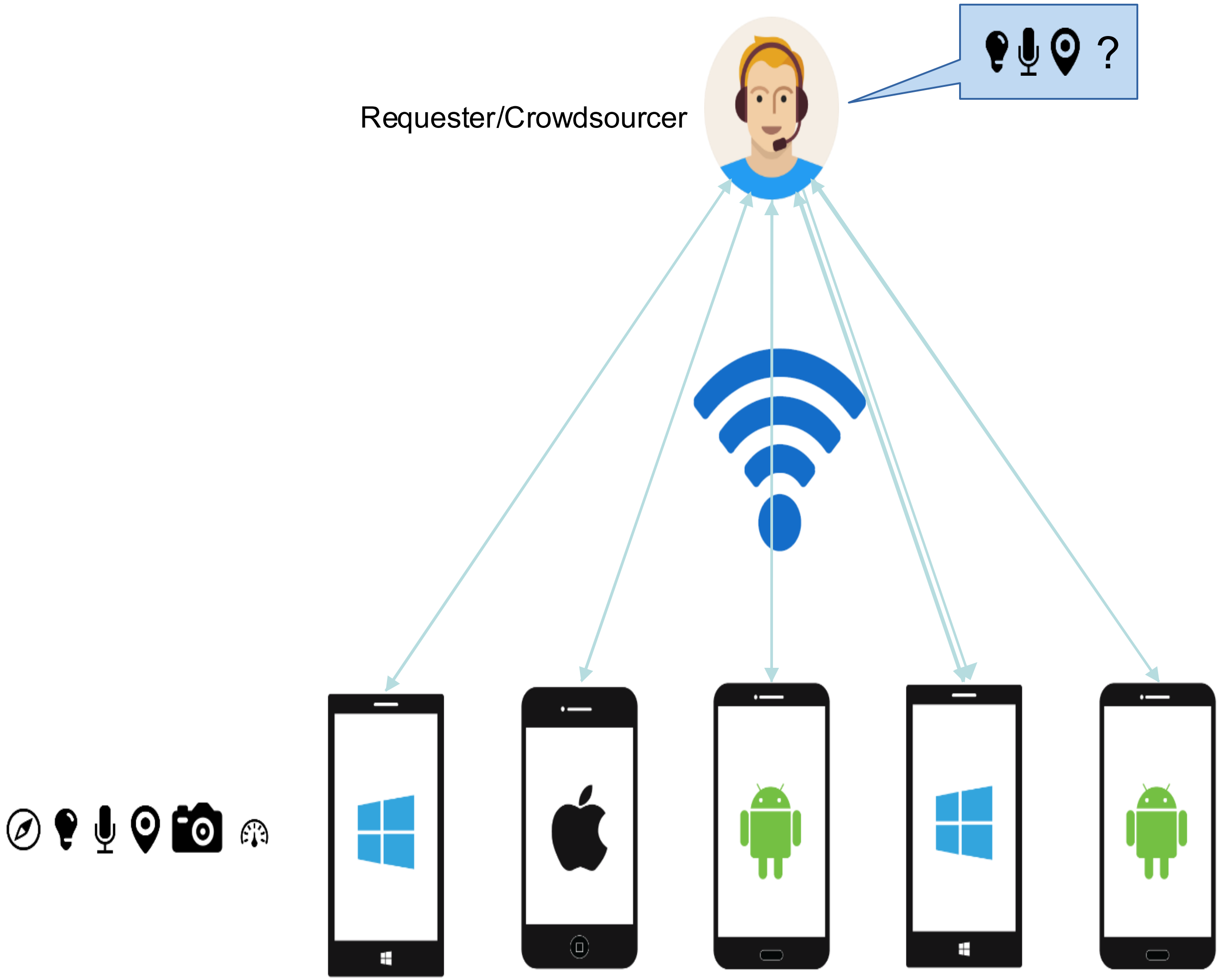
A accept but D refuse

coordinator announces new percentage for A

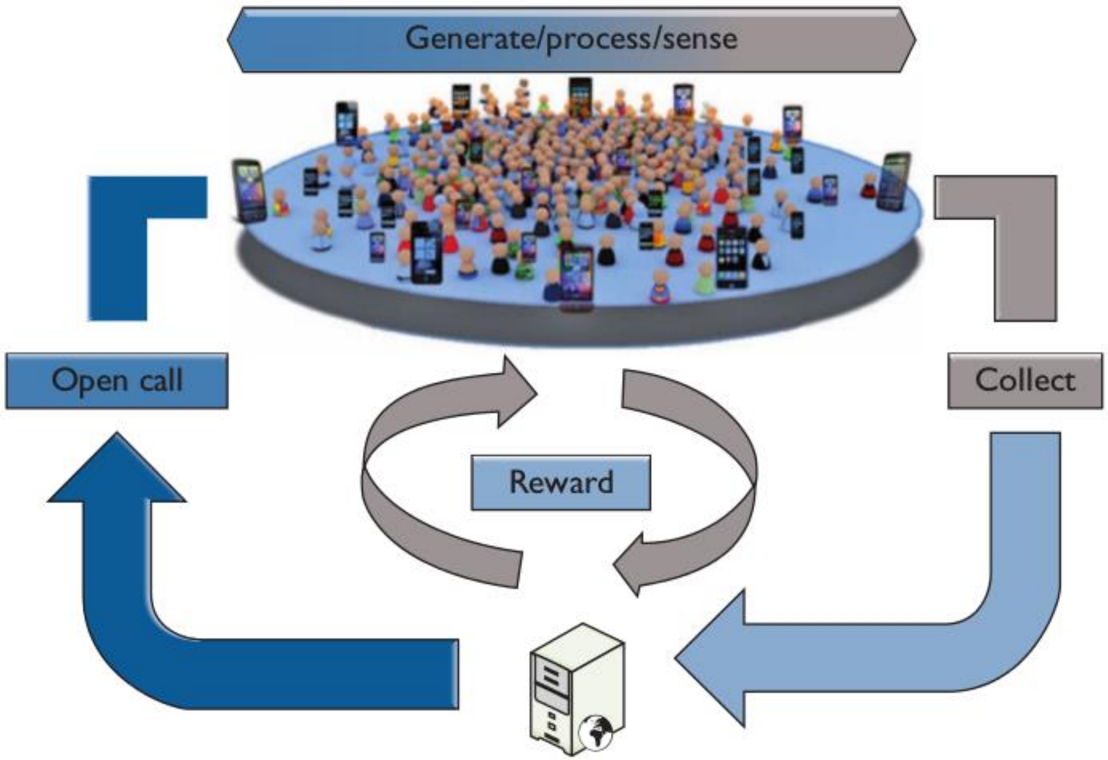
A accepts

coordinator sends new offer

Mobile Crowdsensing



Mobile Crowdsensing (MCS)



- Petar Mrazovic, Mihhail Matskin. MobiCS: Mobile Platform for Combining Crowdsourcing and Participatory Sensing. 2015 IEEE 39th Annual International Computers, Software & Applications Conference (COMPSAC2015), IEEE Computer Society, Taichung, Taiwan, July 1-5, 2015,553-562
- Ville Granfors, Johan Waller, Petar Mrazovic, and Mihhail Matskin. CrowdS: Crowdsourcing with Smart Devices. Int'l Conf. Internet Computing and Internet of Things | ICOMP'18, Las Vegas, NV, 2018
- Viktoriya Kutsarova, Mihhail Matskin:Combining Mobile Crowdsensing and Wearable Devices for Managing Alarming Situations. COMPSAC 2021: 538-543
- Justas Dautaras and Mihhail Matskin. Mobile Crowdsensing with Imagery Tasks. WI-IAT Conference, Melbourne, 14-17 Dec., 2021

Incentive mechanism

- Goal:
  - to develop an incentive mechanism for MCS systems which can motivate participants to reveal their valuations truthfully and insures fairness both for service providers and requesters
- Problem scope:
  - the incentive mechanisms based on the sealed-bid reverse auction with multiple winners
- Approach:
  - develop an incentive mechanism for MCS systems which can motivate participants to reveal their valuations truthful – Vickrey-Clarke-Groves (VCG) mechanism

Ozlem Zehra Ozyagci and Mihhail Matskin. "Truthful Incentive Mechanism for Mobile Crowdsensing with Smart Consumer Devices". In Proc: 4th IEEE International Workshop on Consumer Devices and Systems (CDS 2016), COMPSAC 2016 Workshops, Atlanta, USA, 2016

Vicrey-Clarke-Groves (VCG) mechanisms

- Agents simultaneously send their valuation values for the goods to be allocated
- Allocation happens according to the maximization of social welfare
- For every agent, a tax is calculated as the Clarke tax algorithm which is considered as a compensation to other agents for the amount of utility they lose if this agent wins.
- The winner pays the amount equal to its tax
- If there is only one good/task is allocated, then VCG mechanism is equal to the Vickrey auction

Handling service request

- **Step 1.** *The cloud finds providers (P) who are in the area of the service request and whose bids are lower than the cost limit*
- *If P is empty, then change the distance between the service request location and available providers – ask user update the request.*
    - *If the user updates the service request, goto* **Step 1**.
    - *If the user does not update the service request, then Exit.*

  – *If P is not empty, then the rVCG mechanism is applied:*
    - *If the maximum number of providers (n) specified in the service request is less than the number of providers in P:*
        – *The providers in P are sorted in ascending order according to the bids of their users. If some of these bids are the equal, then the providers with the equal bids are sorted according to their waiting time.*
        – *The first n providers are selected. The payment of each selected provider's user is determined to be equal to the bid of the $n+1^{st}$ smart device's user.*
    - *Otherwise, all providers in P are selected. The payment of each selected provider's user is equal to their bids.*
- **Step 2.** *The cloud notifies the selected smart devices and receives their sensor outputs. The online waiting times of the selected providers are reset to 0.*
- **Step 3.** *The cloud pays to the providers whose smart devices' sensor outputs have been received. The service price is equal to the sum of the payments. The user who requested the service is charged the service price.*
- **Step 4.** *The cloud sends received sensor outputs to the user who sent the service request.*

Production sequencing as negotiation

Production sequencing is treated as a multi-agent negotiation process (Jennings)

Problem overview

Typical manufacturing plant

- number of production cells (paint spraying, cleaning, assembly etc.)
- every period of time it is necessary to plan how to use them
- production sequence defines the order in which products will pass through cells

Some production sequences cost more than another

A factory has

- a spraying cell (any number of colors but it is too expensive to change color)
- manufacturing cell (petrol, diesel)

<green,green,red> is cheaper than <green,red,green>

and

<<red,petrol>, <green,petrol>, <red,diesel>>
can be cheaper than
<<red,petrol>,<red,diesel>,<green,petrol>>

Finding the cheapest production sequence with respect to the whole factory - production sequencing problem

Production sequence as negotiation

- real factory made up of a number of cells
- each cell has associated cost function
- the cost function implicitly defines preferences over production sequence
- we can consider cell as an agent
- agents negotiate a production cost for a sequence of products
- the goal is to improve agent's utility by reducing cost of products

Advantages of the approach

- modularity
- adequate modeling
- flexibility
- parallelism

More details about the example

Each product has 3 attributes:

- color (green, red)
- engine (petrol, diesel)
- drive side (right, left)

Factory has 3 production cells.

Cost functions for each cell:

c1. it costs cell 1 a total 5 units to process each different product, plus 5 units for every change of engine type, and additional 5 units for every change of drive side

c2. it costs cell 2 a total of 5 units to process every product, plus 5 units for every color change

c3. it costs cell 3 a total of 5 units to process every product, plus 5 units for every change of engine type

Production sequences and their costs

p1=<red, petrol, left>; p2=<red, diesel, right>; p3=<green, diesel, left>

For 2 x p1, 1 x p2 and 1 x p3 there are 12 production sequences that satisfy this order

		c1(Sn)	c2(Sn)	c3(Sn)	cf(Sn)
S1	<p1,p1,p2,p3>	30	25	25	80
S2	<p1,p1,p3,p2>	25	30	25	80
S3	<p1,p2,p1,p3>	40	25	35	100
S4	<p1,p2,p3,p1>	35	30	30	95
S5	<p1,p3,p1,p2>	30	30	35	95
S6	<p1,p3,p2,p1>	35	30	30	95
S7	<p2,p1,p1,p3>	35	25	30	90
S8	<p2,p1,p3,p1>	40	30	35	105
S9	<p2,p3,p1,p1>	30	30	25	85
S10	<p3,p1,p1,p2>	30	25	30	85
S11	<p3,p1,p2,p1>	35	25	35	95
S12	<p3,p2,p1,p1>	30	25	25	80
Max cost per cell.		**40**	**30**	**35**	

Properties of production sequences

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
utility1(Sn)	10	15	0	5	10	5	5	0	10	10	5	10
utility2(Sn)	5	0	5	0	0	0	5	0	0	5	5	5
utility3(Sn)	10	10	0	5	0	5	5	0	10	5	0	10
optimal?	x	x										x

Negotiation mechanism

Extended and adopted version of Monotonic Concession Protocol and Zeuthen strategy

- negotiation proceeds in rounds and on the first round every agent takes an active part by proposing some deal
- if the proposed deal makes everybody happy then negotiation ends
- otherwise, next round, where some subset of agents must concede
  - what should the first proposal be?
  - who should concede?
  - how much should concede?

What should be the first proposal?

no negotiation history

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
utility1(Sn)	10	15	0	5	10	5	5	0	10	10	5	10
utility2(Sn)	5	0	5	0	0	0	5	0	0	5	5	5
utility3(Sn)	10	10	0	5	0	5	5	0	10	5	0	10
optimal?	x	x										x

$b'p_i = \{S|S \in (NS - T_i) \ \& \ c_i(S) = min\{c_i(S')|S' \in (NS-T_i)\}\}$

where *NS* - all possible deals

S - next deal
T_i - previous deals made (empty at the beginning)

We can require that *i* only proposes deals in bp_i that minimizes factory costs

$bp_i = \{S|S \in b'p_i \ \& \ cf(S) = min\ \{cf(S')|S' \in b'p_i\}\}$

Who should concede?

Zeuthen Strategy

Degree of willingness to risk a conflict can be defined as:

$$\text{Risk}^t_i = \frac{\text{utility } i \text{ loses by conceding and accepting } j\text{'s offer}}{\text{utility } i \text{ loses by not conceding and causing conflict}}$$

$\text{Risk}_i = 1$ if $u_i(d_i) = 0$

$$\text{Risk}_i = \frac{u_i(d_i) - u_i(d_j)}{u_i(d_i)} \quad \text{otherwise}$$

How much should be conceded?

1. $S_i(t+1) \neq T_i$ - proposing a deal that you proposed earlier is pointless (illegal)
2. $S_i(t+1) >_{Ag-i} S_i(t)$ - the deal is at least as good for every other agent and better for at least one other agent
3. $cf(S_i(t+1)) < cf(S_i(t))$ improves a lot of the whole factory
4. concession is sufficient to change the balance of risk and next round somebody else concede
5. $ci(S_i(t+1)) = min\{ci(S) | S \in NS \;\&\; S$ satisfies (1)-(4)$\}$ - deal *i* offer the minimal concession that *i* could make

The Negotiation Algorithm

The algorithm will use the following variables:

- t is a counter
- $T_i \subseteq NS$ represents the set of deals proposed by agent i thus far, for all $i \in Ag$
- $Active \subseteq Ag$ represents the set of agents still active in negotiation

The Negotiation Algorithm (cont.)

1. Set t to 1.
2. For each agent $i \in Ag$, set T_i to $\emptyset$
3. Set *Active* to *Ag*.
4. For each agent $i \in Ag$ compute bp_i
5. For each $i \in Ag$, non-deterministically select a deal $S_i(1)$ from bp_i
6. For each agent $i \in Active$ set T_i to $T_i \cup \{Si(t)\}$
7. Check for agreement. This is done by computing the agreement set: $\{S | S \in \cup T_i$ and $\forall j \in Ag, \exists S'$ such that $utility_j(S) \geq utility_j(S')\}$ and if $S \neq \emptyset$ then exit with any element from S
8. For each agent $i \in Active$ compute $risk_i(t)$
9. Let g be the set of agents such that $\forall i \in g, risk_i(t) = min\{risk_j(t) \mid j \in Active\}$
10. For each agent $i \in g$:
    – if i can make a confession deal, that satisfies the properties listed earlier, then set $S_i(t+1)$ to be such a deal
    – if i cannot make a concession deal, then compute bp_i:
      - if $bp_i \neq \emptyset$ then set $S_i(t+1)$ to be any element of bp_i
      - if $bp_i = \emptyset$ then set *Active* to *Active* - $\{i\}$
11. For each agent $i \in Active - g$, set $S_i(t+1)$ to be $S_i(t)$
12. Set t to $t+1$
13. Goto 6

Theorem: After a finite number of steps, the agreement set will be non-empty, and thus the negotiation algorithm is guaranteed to terminate with agreement.

Example cont. (running algorithm)

- $t = 1$; $T_i = \varnothing$; *Active=Ag*
- First deal for agents:
  $bp_1 = \{S2\}$,
  $bp_2 = \{S1, S12\}$,
  $bp_3 = \{S1, S2, S12\}$

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
utility1(Sn)	10	15	0	5	10	5	5	0	10	10	5	10
utility2(Sn)	5	0	5	0	0	0	5	0	0	5	5	5
utility3(Sn)	10	10	0	5	0	5	5	0	10	5	0	10
optimal?	x	x										x

- non-deterministically: Agent1 chooses *S2*, Agent2 chooses *S1* and Agent3 chooses *S12*
- willingness to risk
  $risk_1(1) = 1/3$, $risk_2(1) = 1$, $risk_3(1) = 0$;

- Agent3 must concede and it attempts to compute {S1} or {S2}

Example cont. (running algorithm)

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12
utility1(Sn)	10	15	0	5	10	5	5	0	10	10	5	10
utility2(Sn)	5	0	5	0	0	0	5	0	0	5	5	5
utility3(Sn)	10	10	0	5	0	5	5	0	10	5	0	10
optimal?	x	x										x

- Round 1
  - Agent3 concedes, so recomputes bp_3 = *{S1,S2}*
- Round 2
  - Agent1 *{S2},* Agent2 *{S1},* Agent3 *{S1}* risk the same Agent3 concedes again and recomputes bp_3 = *{S2}*
- Round 3
  - Agent1 *{S2}*, Agent3 *{S2}* and Agent2 *{S1}* risk the same Agent3 concedes again and recomputes $bp_3 = \varnothing$ and retires, *Active={1,2}*
- Round 4
  - Agent1 *{S2},* Agent2 *{S1}* $risk_1(4) = 1/3$, $risk_2(4) = 1$

Agent1 must concede, recomputes bp_1 = *{S1,S12},* agreement will be reached in Round 5 independently of choice

Next Lecture: Communication in MAS

Related to:

Chapters 6,7

in

Wooldridge: ”Introduction to MultiAgent Systems”