Agent-Based Negotiation

Dr Enrico Gerding

eg@ecs.soton.ac.uk

COMP6203 Intelligent Agents

Part 1

Learning Outcomes/Aims

This lecture will cover:

- Approaches for conflict resolution between agents
- Negotiation protocols
- Negotiation behaviour
- Types of negotiation environments: e.g. single-issue (zero-sum), multi-issue, etc

This lecture will introduce the concepts informally whereas Part 2 will introduce formal notation to discuss these aspects.

Conflict of interest between agents

- Conflict arises when agents have different preferences or aims
- Conflicts of interest are a daily occurrence for people
  - Households: Who does the washing up? Where do we go out for dinner? What temperature should the house be? Who gets to use the car? Where and when do we meet up? Etc
  - Commerce, Business (contract negotiation) and Politics (e.g. trade agreements)
- Similarly, in a multi-agent system conflicts arise when agents are *self-interested* (e.g. maximising their own benefit) and the agents represent *different stakeholders* (i.e. different sets of interests/aims)
- For example, a *buyer* wants to pay as little as possible, whereas a *seller* wants to maximise profits, resulting in a *conflict*

Conflict *resolution*

- Conflict resolution is possible when there is a mutual benefit to reach an agreement, i.e. there is a conflict of interest, but agents are better off by reaching an agreement than not reaching one
- Approaches for conflict resolution:
  - Auctions
  - Voting
  - Negotiation
- Different approaches suitable for different types of settings

Auctions

- Primarily used to allocate *scarce resources* or *tasks*, e.g.:
    - Items to buyers
    - Advertising space to advertisers (sponsored search and display advertising)
    - Cloud computing
    - Tasks to robots
    - Stocks and shares (financial exchanges)
    - …
- Characterised by:
    - Clearly defined protocol (aka rules or *mechanism), e.g. English auction, Dutch auction, Vickrey auction*
    - Typically requires a trusted third party, i.e. auctioneer
    - Often involves a continuous resource such as *money*
    - Exploits competition between agents (works better with more agents)

Voting

- Used for group based decisions, so called *social choice, e.g.*
  - Where to go out to dinner
  - Where to build new bridge/services/housing
  - What temperature the house should be
  - (In politics) Choosing party/representative
- Characterised by
  - A single decision from a (typically finite) number of options
  - Each agent can have different preferences each option, which is given by a preference *order* (a.k.a. ordinal utility function)
  - Clearly defined protocol

Negotiation (a.k.a. Bargaining)

- Negotiation is governed by a **protocol** which defines the “rules of encounter” between the agents, including:
  - Type of communication allowed
  - Type of proposals that are allowed
  - Who can make what proposal at what time
- More flexible compared to other approaches:
  - Protocol typically involves exchanging *offers*, but can also include other information such as *arguments* (i.e. reasons why)
  - Allows for less structured protocols
  - Often bilateral (between two agents), but some protocols support multi-party negotiation (between more than two agents)
- Enables more complex types of agreements (e.g. multi-issue negotiation)
- Often decentralised (can involve mediator – agent-mediated negotiation)

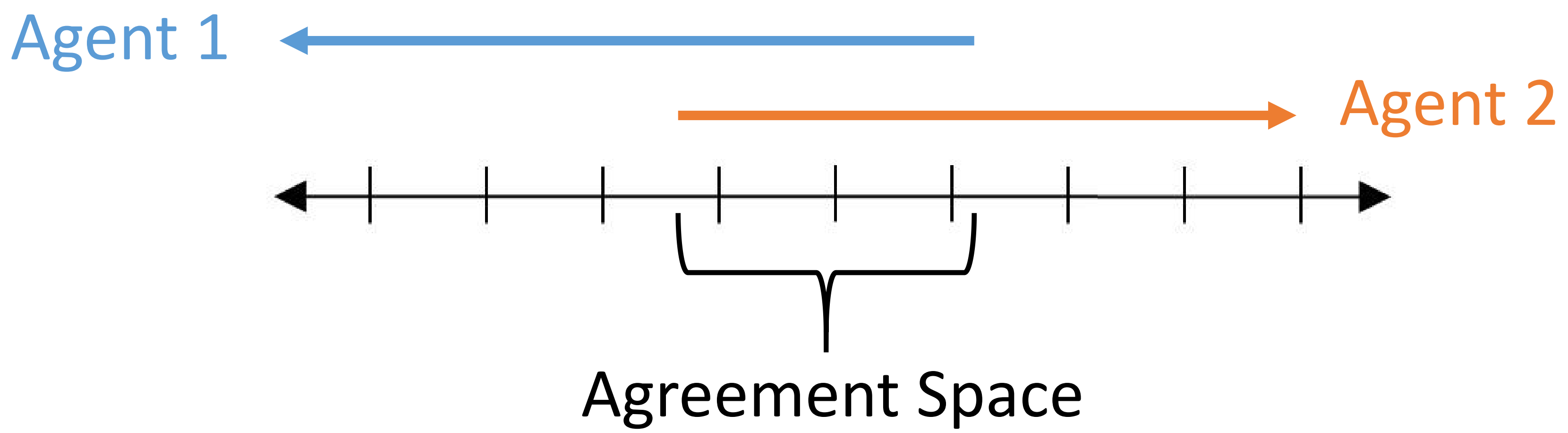
Characterising Agent Negotiations (cont.)

- Negotiation **environment**: set of possible outcomes/agreements
  - Single-issue negotiation (aka distributive bargaining): e.g. price. Win-lose, competitive setting
  - Multi-issue negotiation (integrative bargaining): include more issues, allows for mutual benefit and cooperation
- Agent **preferences**
  - The preferences over all possible agreements (and disagreement), typically specified using **utility functions**
- Agent negotiation **strategies**
  - Specifies the *behaviour* of the agents
  - Actions employed (e.g. the proposals made) at each possible decision point, given the information available to the agent

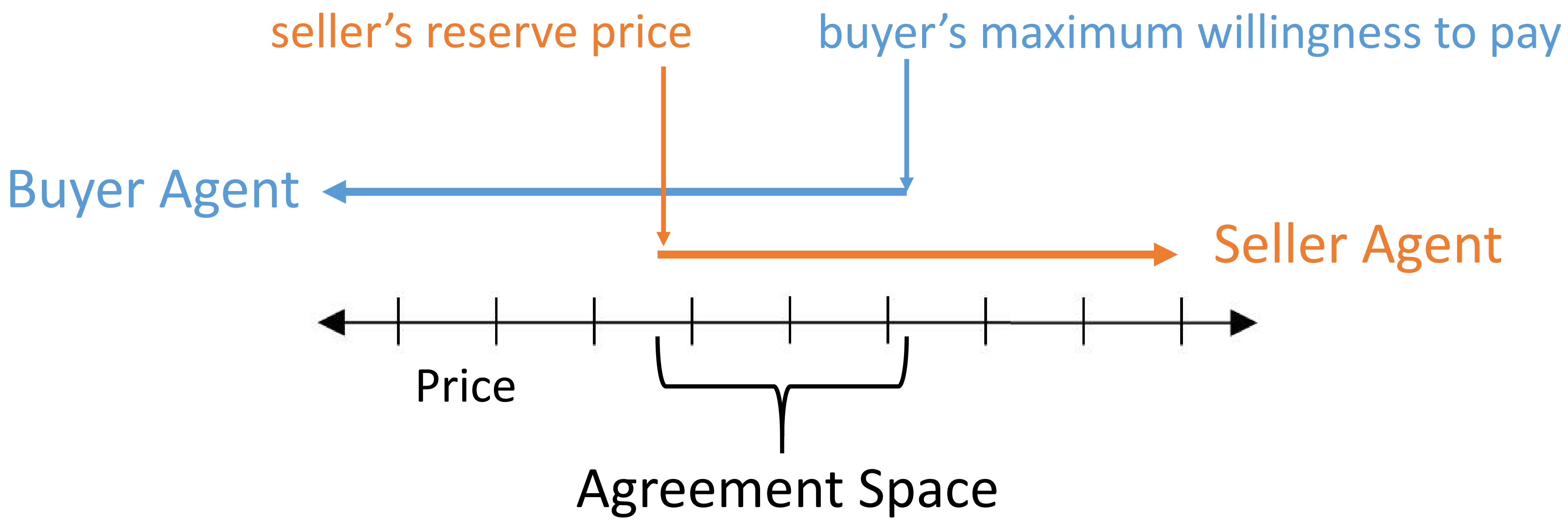
Negotiation Protocols

- Ultimatum Game
- Alternating Offers
- Monotonic Concession Protocol
- Divide and Choose

Single-Issue Negotiation



Single-Issue Negotiation: Price

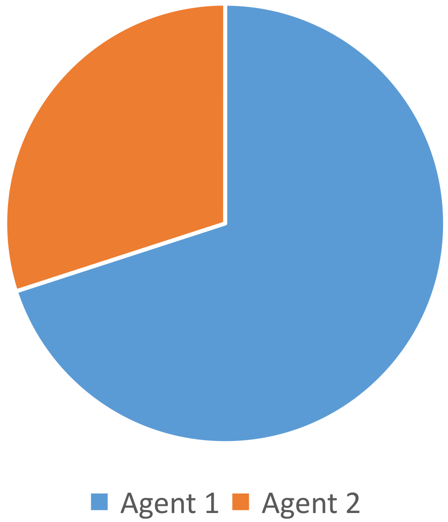


Agreement Space as a Pie/Cake

Negotiation Problem becomes: How to divide a (uniform) cake between two agents?

Bilateral Negotiation: The Ultimatum Game

- Agent 1 suggests a division of the cake



- Agent 2 can only choose to accept or reject the cake
- If agent 2 rejects, no-one gets any cake (the cake gets thrown away)
- Also known as “take it or leave it” game

Reflection Exercise 1

Before moving to next slide, pause the video and have a think about the following points:

1. What would good strategies be of the agents if they are completely selfish (and rational)
2. Is this a good negotiation protocol? What are the advantages and disadvantages?
3. What could be alternative protocols?

Reflection Exercise 1 Explained: (Rational) Behaviour in the Ultimatum Game

- Suppose agent 1 offers agent 2 half the cake. Will agent 2 accept if she/he is rational?
- What about less than half the cake?
- What about a tiny sliver of cake?
- So what's agent's 1 best strategy, given agent 2's behaviour?

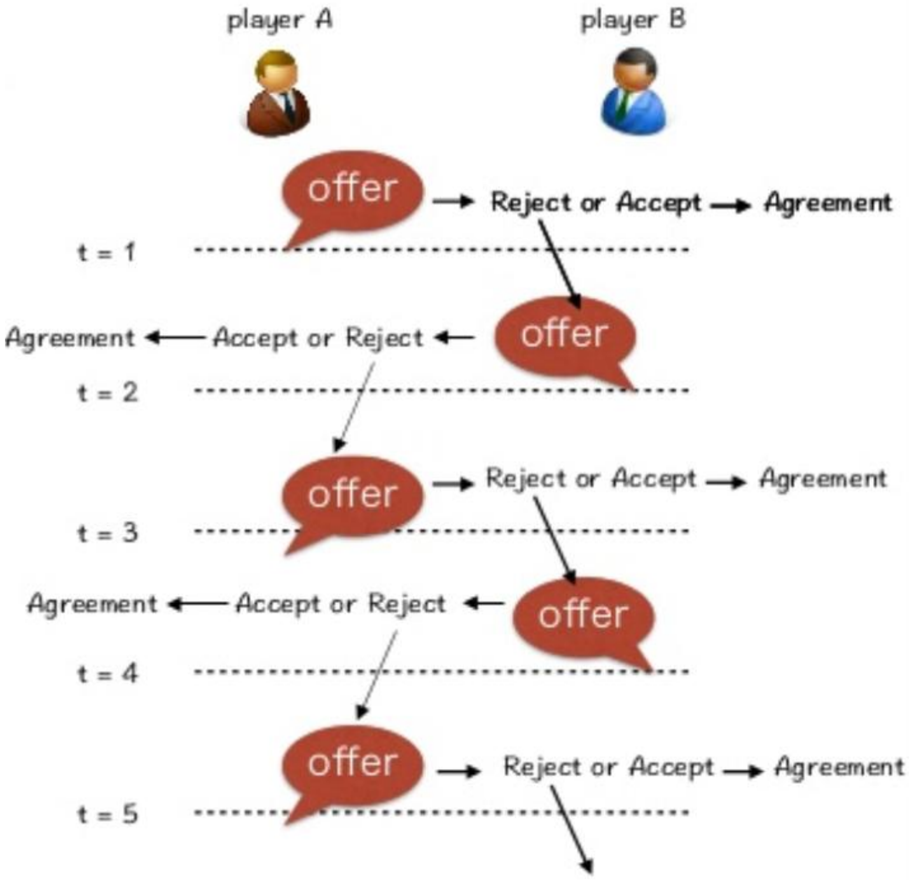
This type of thinking is formalised in *game theory,* which analyses such rational, strategic behaviour between self-interested agents. We will cover the topic of game theory extensively in these lectures.

Reflection Exercise 1 Explained: Advantages/Disadvantages

- Agent 1 has a clear advantage, the so-called first-mover advantage
    - Results in outcomes that can be seen as *unfair*
- Assumes that agreement space (a.k.a. the size of the pie) is “known”, i.e. the seller’s reserve and buyer’s willingness to pay are shared knowledge
- No ability to explore the negotiation space, which is especially important when there are multiple issues (as we will discuss later)

Alternating Offers Protocol

- Negotiation consists of a number of rounds
- Agents exchange offers in an alternating fashion
- First player starts with proposing an offer. Second player can either accept or reject and counter offer, or reject and break off negotiation.
- Negotiation ends after set number of rounds (i.e. deadline) or if either players breaks off negotiations



Reflection Exercise 2

- What are good strategies for the alternating offer game?
- How does the protocol compare to the ultimatum game?

Reflection Exercise 2 Explained

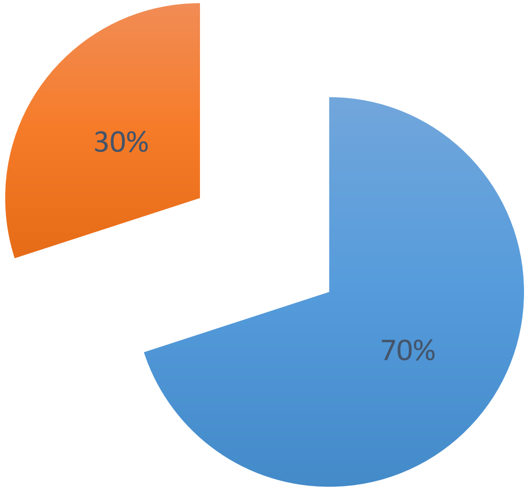
- Although the protocol offers exchange of more rounds, is there any incentive for any of the agents to concede?
- Note that there is a final round which is crucial here
- The last round is identical to the ultimatum game
- So depending on who gets to make the last offer, they can again propose a take-it-or-leave-it offer, which a rational agent would accept
- So we need some way to incentivise a concession.
- This can be due to some form of time pressure (discussed in Part 2), or by changing the protocol (discussed next)

Monotonic Concession Protocol

- Negotiations proceed in rounds
- In each round, the agents simultaneously propose offers (without seeing the other offer)
- If both offers “match” then one of them is chosen
- If offers don’t match, then negotiation proceeds to the next round
- In the next round, at least one of the agents need to concede.
- If neither of the agent concedes, the negotiation ends without a deal.

Divide and Choose

- In case the “cake” denotes a continuous resource that needs to be divided between multiple agents (e.g. inheritance money or land) then the “divide and choose” protocol can be used. Note it does not make sense for other types of negotiations such as price negotiation.
- The protocol consists of 2 steps:
  - First, Agent 1 divides the cake into 2 portions
  - First, Agent 2 chooses one of the portions



Reflection Exercise 3

- What is the optimal strategy of the agents?
- Is the outcome fair?

Reflection Exercise Explained+Additional Notes

- Since agent 2 is going to choose the best portion for them, agent 1 has an incentive to divide the cake equally.
- So each agent ends up with half
- Remember that this protocol only works for so-called resource allocation problems, where there is some (continuous) resource that needs to be divided between multiple agents
- This protocol also works for non-homogenous resources, e.g. when parts of the cake are more attractive (e.g. in the case of land division)
- There are equivalent protocols for more than 2 agents, but this gets more complicated

Desirable properties of a negotiation

- A deal should be better than no deal for all agents (individual rationality)
- There should be “no money” (or cake) left on the table (Pareto efficiency)
- The agreement should be “fair”

We will formalise these terms in Part 2

Part 2

Learning Outcomes/Aims

This lecture will cover:

- Concept of utility function to capture agent *preferences*
- Formalising different types of negotiation *environments*: time pressure, single-issue (zero-sum), multi-issue
- Desirable properties of negotiation *outcomes,* including Pareto efficiency and fairness

Agent Preferences: Formal

We model an agent's preferences over outcomes using a **utility function,** $\boldsymbol{U(o)}$, where $\boldsymbol{o} \in \boldsymbol{O}$ is the offer (or *outcome* or *proposal* or *agreement*) and $\boldsymbol{O}$ is the set of possible offers.

There are 2 types of utility functions:

- **Ordinal preferences:** a preference *order* over outcomes is specified but there is no numerical utility

  e.g. $\boldsymbol{U(o1) \succ U(o2)}$ means that *o1* is preferred over *o2*

The agents specify what they prefer but not by how much

- **Cardinal preferences:** each outcome has a *numerical utility value*

  e.g. $\boldsymbol{U(o1) = 0.78}$, $\boldsymbol{U(o2) = 0.5}$

Note that we can always infer ordinal preferences from cardinal ones

Example: Buyer/Seller Price Negotiation

- Set of possible outcomes ***O*** are: the *price,* denoted by *p* or a *disagreement*.
- The cardinal utility functions are given by:

$$U_{seller}(p) = p - r$$
$$U_{seller}(disagreement) = 0$$
$$U_{buyer}(p) = v - p$$
$$U_{buyer}(disagreement) = 0$$

where:

r is the seller’s **opportunity cost/reserve value** (=minimum price)

v is the buyer’s **value** for the good (=maximum willingness to pay)

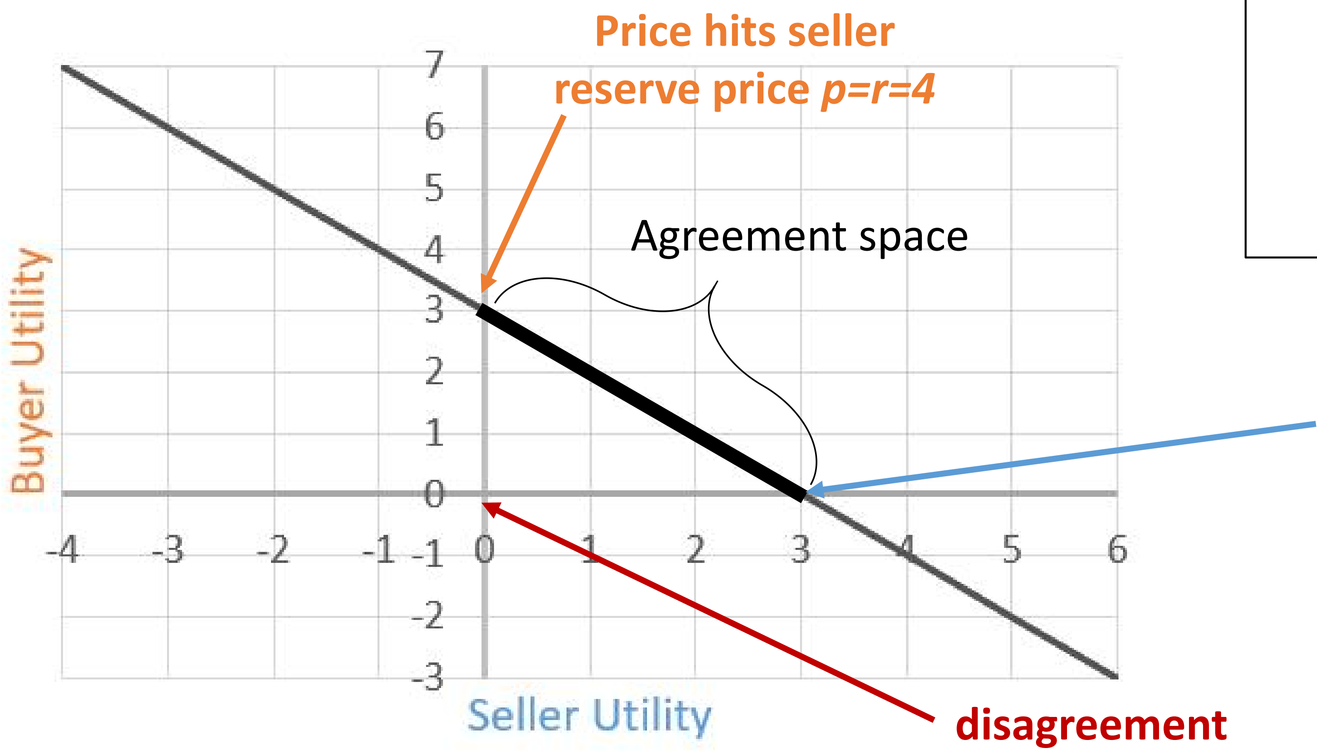
Note that the buyer prefer to pay less, and the seller prefers to pay more.

Price Negotiation: Utility Space

- We can visualise the outcomes using the *utility space,* which shows the utility of the two agents on respective axis for all possible outcomes (i.e. prices)
- In this example:
  - Seller's reservation value is set to *r=4*
  - Buyer's valuation/willingness to pay is set to *v=7*

$$U_{seller}(p) = p - 4$$
$$U_{seller}(disagreement) = 0$$
$$U_{buyer}(p) = 7 - p$$
$$U_{buyer}(disagreement) = 0$$

Agreement space: $p \in [4,7]$



Time pressure

- Negotiation cannot go on for infinity
- Several reasons for time pressure and different ways to model them:
  - Deadlines
    - Imposed by protocol (e.g. Ultimatum game is 1 round)
    - Or determined by individual constraints – in which case deadlines of agents can differ
  - Break-off probability
    - Imposed by protocol (e.g. like the monotonic concession protocol)
    - Or agents can decide to break off negotiations themselves
  - Bargaining costs
    - Fixed bargaining costs per round
    - Discount factors

Modelling bargaining costs

- **Fixed costs:** Let $c_i$denote the costs for agent i, and t is time or bargaining *round,* then the utility at time t is given by:

$$U_i^t = U_i - t \cdot c_i$$

- **Discount factors**: An analogy is that of a “Melting” ice cake – the longer you wait, the more the cake “shrinks”. Let $\delta_i < 1$ is the *discount factor* of agent i, then the utility is given by:

$$U_i^t = U_i \cdot \delta_i^t$$

Multi-Issue Negotiation

- Negotiations often involve multiple other issues, such as time, quality of service, delivery time, etc
- In that case an offer/outcome o is a *vector* consisting of a value, o_j for each issue *j*
- Often **weighted additive utility function** assumed:
$$U_i(o) = \sum_j w_{i,j} \cdot U_{i,j}(o_j)$$
where $U_{i,j}(o_j)$ is the utility for issue *j*

Additive Utility Function Example

- Suppose there are two cakes, 1 and 2, and offer o_j represents the share that agent 1 receives for cake $j \in \{1,2\}$ (so agent 2 gets (1-o_j)). So we have:

$$U_{1,j}(o_j) = o_j \text{ and } U_{2,j}(o_j) = (1 - o_j) \text{ for } j \in \{1,2\}$$

- Agent 1 prefers cake 1, and agent 2 prefers cake 2. In particular, agent 1 and agent 2 have weights 7, 3 and 3,7 respectively.
- This results in utility functions:

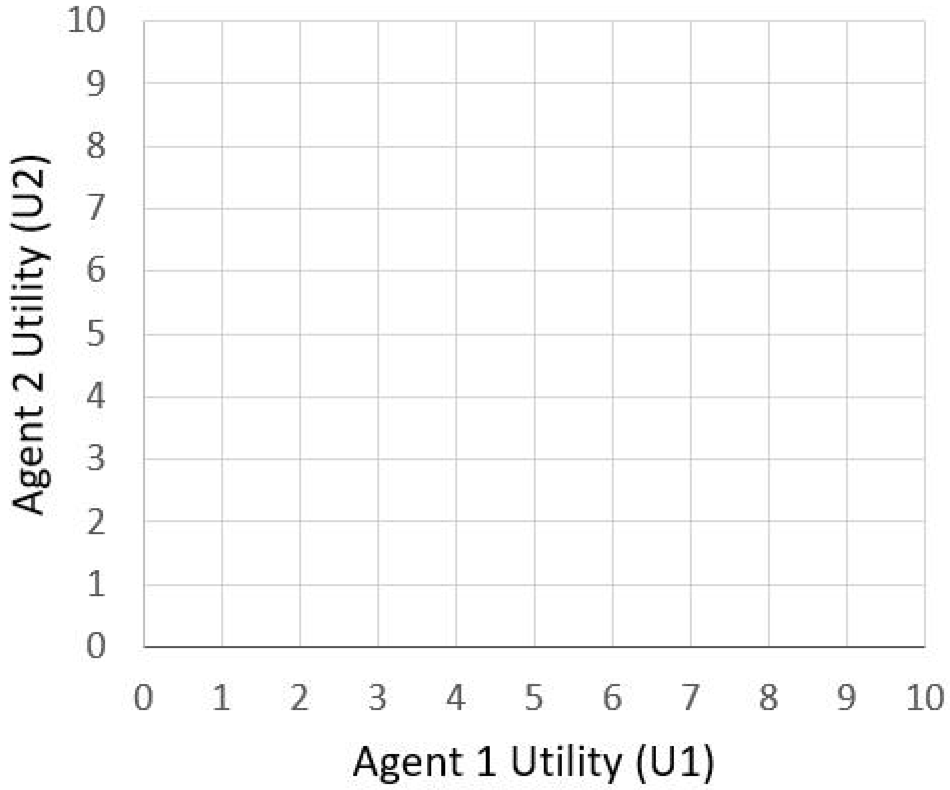
$$U_1(o) = 7 * o_1 + 3 * o_2$$

$$U_2(o) = 3 * (1 - o_1) + 7 * (1 - o_2)$$

Utility Space: Exercise

$$U_1(o) = 7 * o_1 + 3 * o_2$$
$$U_2(o) = 3 * (1 - o_1) + 7 * (1 - o_2)$$

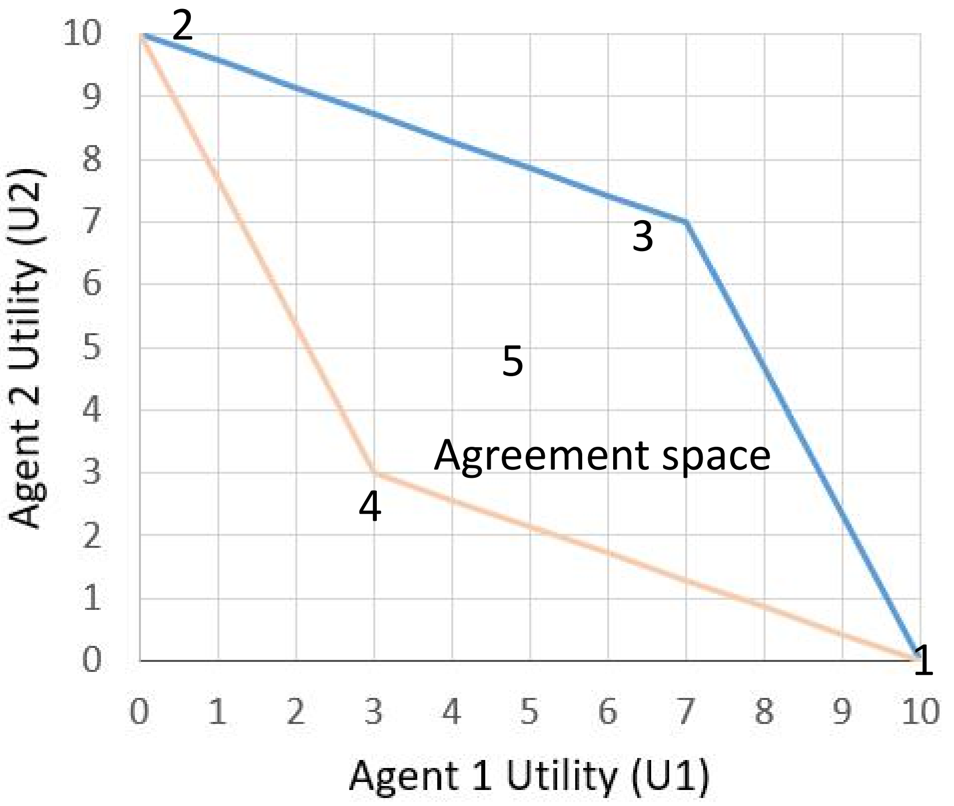
Offer	o_1	o_2	U_1	U_2
1	1	1		
2	0	0		
3	1	0		
4	0	1		
5	0.5	0.5		



Example Utility Space (Answer)

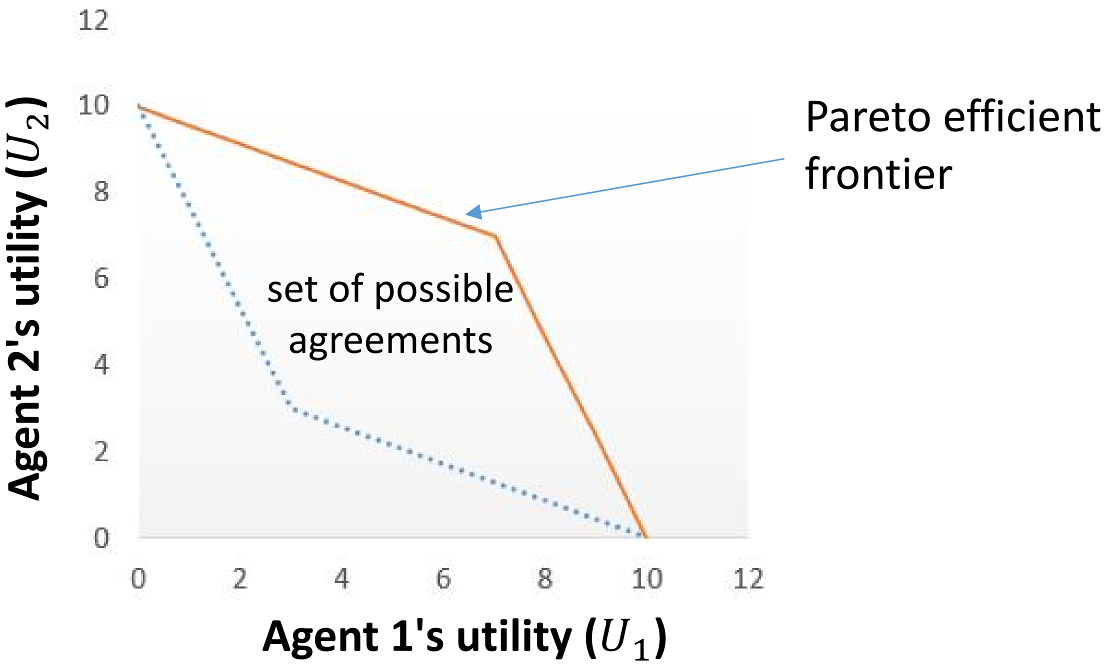
$$U_1(o) = 7 * o_1 + 3 * o_2$$
$$U_2(o) = 3 * (1 - o_1) + 7 * (1 - o_2)$$

Offer	o_1	o_2	U_1	U_2
1	1	1	10	0
2	0	0	0	10
3	1	0	7	7
4	0	1	3	3
5	0.5	0.5	5	5



Pareto-Efficient Agreements

- An agreement is said to be **Pareto efficient** (or Pareto optimal) if no further improvement is possible in the utility of one agent, without reducing the utility of the other agents
- **Pareto efficient frontier**: Set of all Pareto efficient agreements
- Taking previous example:



Wilfredo Pareto

Desirable properties

- Agreements should be *individually rational: U(o)>U(disagreement)*
- Agreements should be *Pareto efficient*
- Agreements should be *fair*

Fairness

Many different concepts of fairness. The main ones are:

- **Utilitarian social welfare:** maximise the *sum* of utilities (aka social welfare). Formally:

$$max_{o \in O} U_1(o) + U_2(o)$$

- **Egalitarian social welfare**: maximise the minimum utility. Formally:

$$max_{o \in O} min_{i \in \{1,2\}} U_i(o)$$

- **Nash bargaining solution**: maximise the *product* of the utility of the agents (minus the disagreement payoff)

$$max_{o \in O}(U_1(o) - U_1(disagreement)) \cdot (U_2(o) - U_2(disagreement))$$

- **Envy-freeness:** no agent prefers the resources allocated to other agents (only makes sense in resource allocation settings)

Nash Bargaining Solution

John Nash (also known for his Nash equilibrium and the movie Beautiful Mind) showed that the Nash bargaining solution is *uniquely* characterised by the following *axioms* or properties

- **Individual rationality**: the solution always satisfies U1 ≥ U1(dis) and U2 ≥ U2(dis)
- **Pareto efficiency**
- **Invariance to equivalent utility representations**: the solution is insensitive to *affine transformations*
- **Independence of irrelevant alternatives (IIA):** remove all the non-optimal agreements, and the optimal agreement remains the same
- **Symmetry (SYM):** if the agents have the same preferences, then the solution gives them the same utilities

Envy-freeness

An agent is envious of the other agent, if it would prefer the allocation received by that agent. A solution is **envy free** if no agent prefers the allocation of another agent, i.e. no agent is envious.

Note: this concept only makes sense in case of resource allocation problems

Example: suppose we have two pies (2 issues) and 2 agents, and o_j is the share of pie *j* that goes to agent 1, and $1 - o_j$ is the share going to agent 2.

- Formally, using the example:
  - Agent *1* is envious if: $U_1(1 - o_1, 1 - o_2) > U_1(o_1, o_2)$
  - Agent 2 is envious if: $U_2(o_1, o_2) > U_2(1 - o_1, 1 - o_2)$
- Note: agents do not care about utility of other agent, only the share/allocation received by other agents

Exercise: Example 1 Revisited

Considering the example from before, which offer(s) satisfy:

- Utilitarian social welfare solution
- Egalitarian social welfare solution
- Nash Bargaining solution
- Envy-free

Example 1 Revisited

$U_1(o) = 7 * o_1 + 3 * o_2$
$U_2(o) = 3 * (1 - o_1) + 7 * (1 - o_2)$

Try and fill out the table below yourself and to come up with the answers.

Offer	o_1	o_2	U_1	U_2	**Utilitarian Sum**	**Egalitarian Min**	**NBS Product**
1	1	1	10	0			
2	0	0	0	10			
3	1	0	7	7			
4	0	1	3	3			
5	0.5	0.5	5	5			

Example 1 Revisited: Envy Free

$$U_1(o) = 7 * o_1 + 3 * o_2$$
$$U_2(o) = 3 * (1 - o_1) + 7 * (1 - o_2)$$

Compute agent utilities when their allocation is REVERSED (i.e. Agent 2 is getting what Agent 1 is getting, and vice versa)

Offer	o_1	o_2	U_1	U_2	U'_1	U'_2	**Envious?**
1	1	1	10	0			
2	0	0	0	10			
3	1	0	7	7			
4	0	1	3	3			
5	0.5	0.5	5	5			

Example 1 Revisited (Answers)

$$U_1(o) = 7 * o_1 + 3 * o_2$$
$$U_2(o) = 3 * (1 - o_1) + 7 * (1 - o_2)$$

Offer	o_1	o_2	U_1	U_2	Utilitarian Sum	Egalitarian Min	NBS Product
1	1	1	10	0	10	0	0
2	0	0	0	10	10	0	0
3	1	0	7	7	14	7	49
4	0	1	3	3	6	3	9
5	0.5	0.5	5	5	10	5	25

Example 1 Revisited: Envy Free

$$U_1(o) = 7 * o_1 + 3 * o_2$$
$$U_2(o) = 3 * (1 - o_1) + 7 * (1 - o_2)$$

Compute agent utilities when their allocation is REVERSED (i.e. Agent 2 is getting what Agent 1 is getting, and vice versa)

Offer	o_1	o_2	U_1	U_2	U'_1	U'_2	Envious?
1	1	1	10	0	0	10	Agent 2
2	0	0	0	10	10	0	Agent 1
3	1	0	7	7	3	3	No
4	0	1	3	3	7	7	Agent 1 & Agent 2
5	0.5	0.5	5	5	5	5	No

There are many others. E.g. (1.0,0.1) is envy free as well, and even (1.0,0.2). However, (1.0,0.3) is no longer envy free, since at that point agent 2 prefers agent 1's allocation. CHECK YOURSELF!

Part 3

Learning Outcomes/Aims

This lecture will cover:

- Basic negotiation *strategies*
- Uncertainty about the opponent
- Preference uncertainty
- Concluding remarks

Negotiation Strategies. Approaches:

- Game theoretic
  - Assumes rules of the game, preferences & beliefs of all players are common knowledge
  - Assumes full rationality on the part of all players (=unlimited computation)
  - Preferences encoded in a (limited) set of player **types**
  - Closed systems, predetermined interaction, small sized games
  - **Nash equilibrium (discussed later in the module)**
- Heuristic perspective
  - No common knowledge or perfect rationality assumptions needed
  - Agent behaviour is modeled directly
  - Suitable for open, dynamic environments
  - Space of possibilities is very large

Heuristics

- Heuristics are often used when there is some unknowns about the opponent, e.g. the strategy they are using
- We can divide a strategy into:
  - **Concession strategy:** what should be the target utility I think I should achieve at a particular point in the negotiation?
  - **Multi-Issue offer producing strategy:** once a target utility is established, what offer to produce at or around that target utility (to ensure Pareto efficiency). Trivial in case of single issues.
- Well-known concession strategies include:
  - **Time-dependent tactics:** only depend on time/current round of the negotiation and not on the opponent’s action
  - **Tit-for-tat**

Concession: Time-dependent tactics

- Let U_{max} and U_{min}denote the agent's maximum and minimum acceptable utility (e.g. reserve price) respectively
- Target utility offered at time t will be:

$$U_{\text{target}}(t) = U_{min} + (1 - F(t))(U_{max} - U_{min})$$

- Where $F(t)$ is a function between 0 and 1 and gives the fraction of the distance between best and worst offer, e.g. using a function such as:

$$F(t) = \left(\frac{\min(t, T_{max})}{T_{max}}\right)^{\frac{1}{\beta}}$$

where T_{max} is the deadline and β is a constant

- Note that, $F(0) = 0$, and $F(T_{max}) = 1$

Time-dependent tactics (Cont.)

- Hard-headed (β->0)

  No concessions. Sticks to the initial offer throughout (hoping the opponent will concede)

- Linear time-dependent concession (β=1)

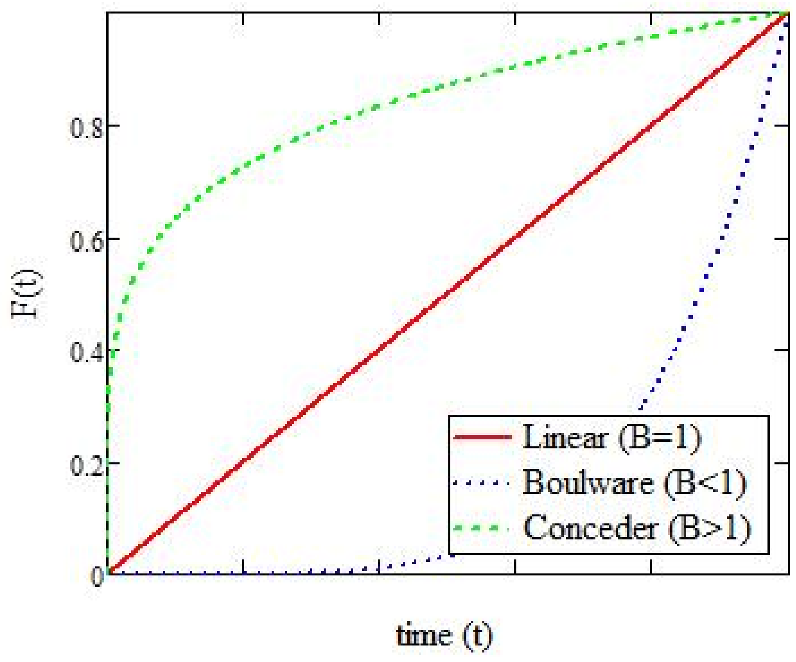
  Concession is linear in the time remaining until the deadline

- Boulware (β<1)

  Concedes very slowly; initial offer is maintained until just before the deadline

- Conceder (β>1)

  Concedes to the reservation value very quickly



Tit-for-tat concession strategy

- The agent detects the concession the opponent makes during the previous negotiation round, in terms of increase in its own utility function
- The concession the agent makes in the next round is equal to (or less than) the concession made by the opponent in the previous round,

$$concession \leq U_{own}(o^{t}_{opponent}) - U_{own}(o^{t-1}_{opponent})$$

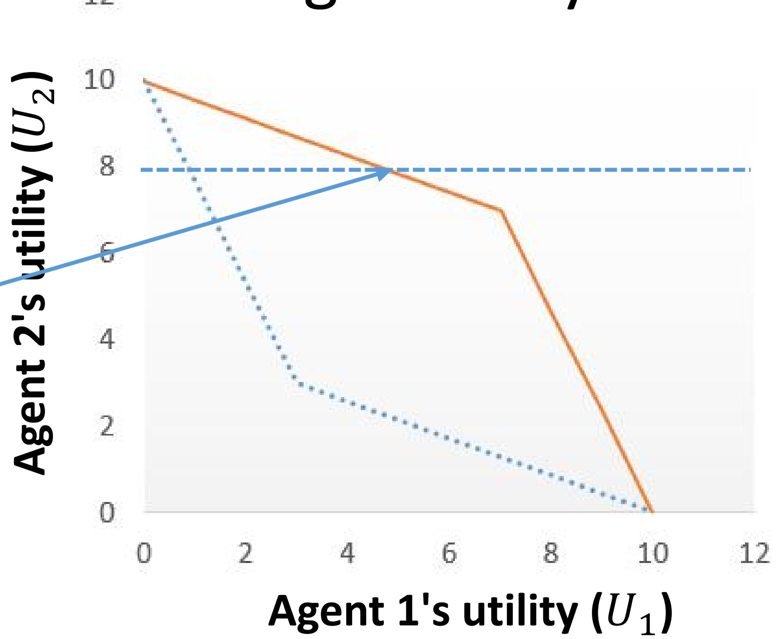
- As long as offer falls in the acceptable region (e.g. for price negotiation, above the reservation price of the seller and below that of the buyer)

Optimal Concession Strategy

- If everything is known, then it is easy to compute a *best response, i.e.* the optimal concession given the utility and strategy of the opponent.
- Note that the optimal strategy depends on the opponent, but also the time pressure (e.g. time discounting or deadline)
- However, often, there is a lot of uncertainty about the opponent's utility function and strategy. In that case, machine learning techniques can be used to try and **learn the opponent model**.
- Also, opponent might be doing the same. Therefore, *game-theoretic* approaches (where you reason about the opponent) can be useful

Offer-producing strategy

- So far, the strategies discussed only provide a *target utility*. In a multi-issue negotiation, you need to generate a value for each issue.
- It is a good idea to ensure that the offer is always Pareto efficient
- if the utility functions are known, it is possible to calculate the Pareto-efficient offer at a certain target utility level



Suppose agent 2’s target utility is set to $U_{target} = 8$

Pareto efficient offer at target utility is $o = (\frac{2}{3}, 0)$

VERIFY THIS AT HOME &

TRY YOURSELF WITH OTHER

TARGET UTILITIES

Unknown Opponent Utility

- Typically the opponent utility is not known, i.e. this is so-called *private information*
- Possible to guess the opponent utility function based on the offers received so far and the concessions observed
- E.g. the opponent is likely to concede on their least preferred issues first (in case of additive utility functions), and so you can use this to guess the weight of that issue
- Many approaches in the literature and this will be discussed in more detail in the coursework *labs*

Preference Uncertainty and Elicitation

- In many cases even “own” utility function is not (fully) known by the agent (who is negotiating on behalf of a human)
- Utility function obtained through a process called *preference elicitation*
  - Has associated “cognitive costs”
  - Trade-off between minimizing cognitive cost and maximising utility
- Some recent work has looked at negotiation with *preference uncertainty*: incomplete information about own utility
- Preference uncertainty and elicitation will be explored as part of the coursework and more details will be discussed in a future lecture and the labs.

Conclusions

- Agent-based negotiation is an extensive field of research
- Focus on bilateral bargaining (between two agents) and multi-issue negotiation (allowing for mutually beneficial outcomes).
- An important aspect of the agent is the negotiation strategy, which is challenging to design because of uncertainty about the opponent
- Recent approaches consider incomplete information about own utility function, and the cost to elicit these preferences