Agent-Based Systems

Michael Rovatsos
mrovatso@inf.ed.ac.uk

Lecture 12 – Bargaining

Where are we?

- Different auction types and properties
- Combinatorial Auctions
- Bidding Languages
- The VCG mechanism

Today . . .

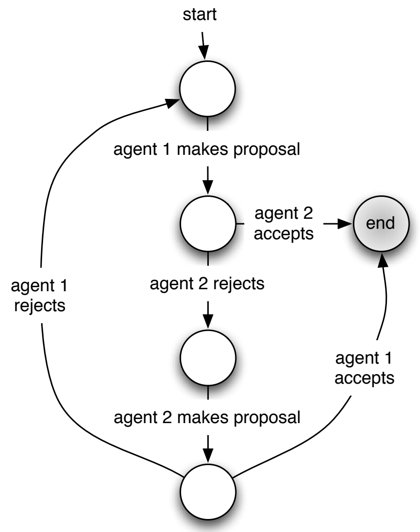
- **Bargaining**

Bargaining

- Reaching agreement in the presence of conflicting goals and preferences (a bit like a multi-step game with specific protocol)
- Negotiation setting:
  - The **negotiation set** is the space of possible proposals
  - The **protocol** defines the proposals the agents can make, as a function of prior negotiation history
  - **Strategies** determine the proposals the agents will make (private)
- Number of issues:
  - Single-issue, e.g. price of a good
  - **multiple-issues**, e.g. buying a car: price, extras, service
    - **Concessions** may be hard to identify in multiple-issue negotiations
    - Number of possible deals: m^n for n attributes with m possible values
- Number of agents:
  - **one-to-one**, simplified when preferences are **symmetric**
  - **many-to-one**, e.g. auctions
  - **many-to-many**, $n(n-1)/2$ negotiation threads for n agents

Alternating Offers

- Common one-to-one protocol



- Negotiation takes place in a sequence of rounds
- Agent 1 begins at round 0 by making a proposal x^0
- Agent 2 can either accept or reject the proposal
- If the proposal is accepted the deal x^0 is implemented
- Otherwise, negotiation moves to the next round where agent 2 makes a proposal

Scenario: Dividing the Pie

- Scenario: Dividing the pie
  - There is some resource whose value is 1
  - The resource can be divided into two parts, such as
    1. The values of each part must be between 0 and 1
    2. The sum of the values of the parts sum to 1
  - A **proposal** is a pair $(x, 1-x)$ (agent 1 gets x, agent 2 gets $1-x$)
  - The negotiation set is: $\{(x, 1-x) : 0 \leq x \leq 1\}$
- Some assumptions:
  - Disagreement is the worst outcome, we call this the **conflict deal** Θ
  - Agents seek to maximise utility

THE UNIVERSITY of EDINBURGH informatics Agent-Based Systems

Negotiation Rounds

- The **ultimatum game**: a single negotiation round
  - Suppose that player 1 proposes to get all the pie, i.e. $(1, 0)$
  - Player 2 will have to agree to avoid getting the conflict deal Θ
  - Player 1 has all the power
- Two rounds of negotiation
  - Agent 1 makes a proposal in the first round
  - Player 2 can reject and turn the game into an ultimatum
- If the number of rounds is fixed, whoever moves last gets all the pie
- If there are no bounds on the number of rounds:
  - Suppose agent 1's strategy is: propose $(1, 0)$, reject any other offer
  - If agent 2 rejects the proposal, the agents will never reach agreement (the conflict deal is enacted)
  - Agent 2 will have to accept to avoid Θ
  - Infinite set of Nash equilibrium outcomes (of course agent 2 must understand the situation, e.g. given access to agent 1's strategy)

THE UNIVERSITY of EDINBURGH informatics Agent-Based Systems

Time

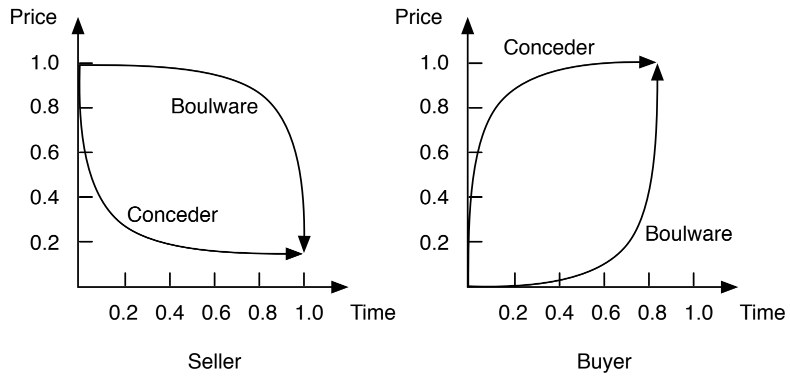
- Additional assumption: **Time is valuable** (agents prefer outcome x at time t_1 over outcome x at time t_2 if $t_2 > t_1$)
- Model agent i's patience using **discount factor** δ_i $(0 \leq \delta_i \leq 1)$

  the value of slice x at time 0 is $\delta_i^0 x = x$
  the value of slice x at time 1 is $\delta_i^1 x = \delta_i x$
  the value of slice x at time 2 is $\delta_i^2 x = (\delta_i \delta_i) x$
- More patient players (larger δ_i) have more power
- Games with two rounds of negotiation
  - The best possible outcome for agent 2 in the second round is δ_2
  - If agent 1 initially proposes $(1-\delta_2, \delta_2)$, agent 2 can do no better than accept
- Games with no bounds on the number of rounds
  - Agent 1 proposes what agent 2 can enforce in the second round
  - Agent 1 gets $\frac{1-\delta_2}{1-\delta_1\delta_2}$, agent 2 gets $\frac{\delta_2(1-\delta_1)}{1-\delta_1\delta_2}$

THE UNIVERSITY of EDINBURGH informatics Agent-Based Systems

Negotiation Decision Functions

- Non-strategic approach, does not depend on how other's behave
- Agents use a time-dependent decision function to determine what proposal they should make
- **Boulware** strategy: exponentially decay offers to reserve price
- **Conceder** strategy: make concessions early, do not concede much as negotiation progresses



Task-oriented domains (I)

- A **task-oriented domain** (TOD) is a triple $\langle T, Ag, c\rangle$ with
  - T a finite set of tasks, Ag a set of agents, and
  - $c : \mathbf{2}^T \rightarrow \mathbb{R}^+$ function describing cost of executing any set of tasks (symmetric for all agents)
- We assume that $c(\emptyset) = 0$, and that c is **monotonic** i.e.

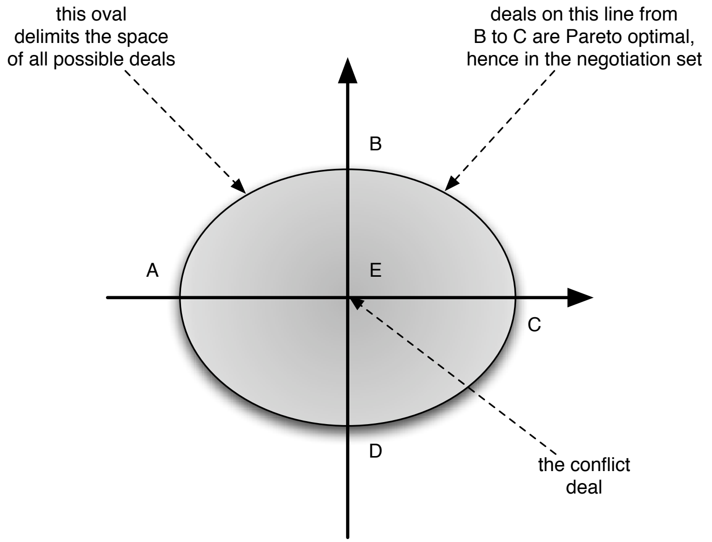
$$T_1, T_2 \subseteq T \ \wedge \ T_1 \subseteq T_2 \ \Rightarrow \ c(T_1) \leq c(T_2)$$

- An **encounter** in a TOD is a collection $\langle T_1, \ldots, T_n\rangle$ such that each $T_i \subseteq T$ is executed by agent $i \in Ag$
- Below, we only consider one-to-one negotiation scenarios where a **deal** is a pair $\delta = \langle D_1, D_2\rangle$ such that $D_1 \cup D_2 = T_1 \cup T_2$
- Agent i will execute D_i in a deal with
  - $cost_i(\delta) = c(D_i)$, and
  - $utility_i(\delta) = c(T_i) - cost_i(\delta)$

Task-Oriented Domains (II)

- Utility represents how much agent has to gain from the deal
- If no agreement is reached, **conflict deal** is $\Theta = \langle T_1, T_2\rangle$
- A deal δ_1 dominates another deal δ_2 (denoted $\delta_1 \succ \delta_2$) iff
  1. Deal δ_1 is at least as good as δ_2 for every agent:
  $$\forall i \in \{1, 2\}, utility_i(\delta_1) \geq utility_i(\delta_2)$$
  2. Deal δ_1 is better for some agent than δ_2:
  $$\exists i \in \{1, 2\}, utility_i(\delta_1) > utility_i(\delta_2)$$
- If δ_1 is not dominated by any other δ_2, then δ is **Pareto optimal**
- A deal is **individually rational** if it weakly dominates (i.e. is at least as good as) the conflict deal Θ

Task-Oriented Domains (III)



Negotiation set contains individually rational and Pareto optimal deals

The monotonic concession protocol

- Start with simultaneous deals proposed by both agents and proceed in rounds
- Agreement reached if
  - $utility_1(\delta_2) \geq utility_1(\delta_1)$ or
  - $utility_2(\delta_1) \geq utility_2(\delta_2)$
- If both proposals match or exceed other's offer, outcome is chosen at random between δ_1 and δ_2
- If no agreement, in round $u + 1$ agents are not allowed to make deals less preferred by other agent than proposal made in round u
- If no proposals are made, negotiation terminates with outcome Θ
- Protocol verifiable and guaranteed to terminate, but not necessarily efficient

The Zeuthen strategy

- The above protocol doesn't describe *when* and *how much* to concede
- Intuitively, agents will be more willing to risk conflict if difference between current proposal and conflict deal is low
- Model agent i's willingness to risk conflict at round t as

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

- Formally, we can calculate risk as a value between 0 and 1

$$risk_i^t = \begin{cases} 1 & \text{if } utility_i(\delta_i^t) = 0 \\ \frac{utility_i(\delta_i^t) - utility_i(\delta_j^t)}{utility_i(\delta_i^t)} & \text{otherwise} \end{cases}$$

The Zeuthen strategy (II)

- Agent with smaller value of risk should concede on round t
- Concession should be *just good enough* but of course this is inefficient, smallest concession that changes balance of risk
- Problem if agents have equal risk: we have to flip a coin, otherwise one of them could defect (and conflict would occur)
- Looking at our protocol criteria:
  - Protocol terminates, doesn't always succeed, simplicity? (too many deals), Zeuthen strategy is Nash, no central authority needed, individual rationality (in case of agreement), Pareto optimality
- Zlotkin/Rosenschein also analysed a number of scenarios in which agents lie about their tasks:
  - Phantom/decoy tasks: advantage for deceitful agent
  - Hidden tasks: agents may benefit from hiding tasks (!)

Bargaining for Resource Allocation (I)

- A **resource allocation setting** is a tuple $\langle Ag, \mathcal{Z}, v_1, \ldots, v_n \rangle$,
  - Agents $Ag = \{1, \ldots, n\}$
  - Resources $\mathcal{Z} = \{z_1, \ldots, z_m\}$
  - **Valuation functions** $v_i : \mathbf{2}^{\mathcal{Z}} \to \mathbb{R}$
- An **allocation** $Z_1, \ldots, Z_n$ is a partition of resources over the agents
- Negotiating a change from P_i to Q_i ($P_i, Q_i \in \mathcal{Z}$ and $P_i \neq Q_i$) will lead to
  - $v_i(P_i) < v_i(P_i)$,
  - $v_i(P_i) = v_i(P_i)$ or
  - $v_i(P_i) > v_i(P_i)$
- Agents can make **side payments** as compensations

Bargaining for Resource Allocation (II)

- A **pay-off vector** $p = \langle p_1, p_2, \ldots, p_n \rangle$ is a tuple of side payments such that $\sum_{i=1}^{n} p_i = 0$
- A **deal** is a triple $\langle Z, Z', \bar{p} \rangle$, where $Z, Z' \in alloc(\mathcal{Z}, Ag)$ are distinct allocations and $\bar{p}$ is a payoff vector
- $\langle Z, Z', \bar{p} \rangle$ is **individually rational** if $v_i(Z'_i) - p_i > v_i(Z)$ for each $i \in Ag$, p_i is allowed to be 0 if $Z_i = Z'_i$
- **Pareto optimal:** every other allocation that makes some agents strictly better off makers some other agent strictly worse off

Protocol for Resource Allocation

1. Start with initial allocation Z^0
2. **Current allocation** is Z^0 with 0 side payments
3. Any agent is permitted to put forward a deal $\langle Z, Z', \bar{p} \rangle$
4. If all agent agree and the **termination condition** is satisfied (i.e. Pareto optimality) then the negotiation terminates and deal Z' is implemented with payments $\bar{p}$
5. If all agents agree but the termination condition is not satisfied, then set current allocation to Z^0 with payments $\bar{p}$ and go to step 3
6. If some agent is not satisfied with the deal, go to step 3

Restricted Deals

- Finding optimal deals is NP-hard, focus on restricted deals
- **One-contracts**: move only one resource and one side payment
  - Restricts search space, agent needs to consider $|Z_i|(n-1)$ deals
  - Can always lead to socially optimal outcome, but requires agents to accept deals that are not individually rational
- **Cluster-contracts**: transfer of any number of resources greater than 1, do not receive anything in return
- **Swap-contracts**: swap one resource and make side payment
- **Multiple-contracts**: three agents, each transferring a single resource
- C-contracts, S-contracts and M-contracts do not always lead to an optimal allocation
- Constraint that each new deal must be individually rational reach a globally good outcome by using only local reasoning

Summary

- Bargaining
- Alternating offers
- Negotiation decision functions
- Task-oriented domains
- Bargaining for resource allocation
- Next time: **Argumentation in Multiagent Systems**