CM30174 + CM50206
Intelligent Agents

Marina De Vos, Julian Padget

Negotiation / version 0.4

November 1, 2011

Authors/Credits for this lecture

- Chs. 14, 15 and 9 of “An Introduction to Multiagent Systems” [Wooldridge, 2009].

Content

Content

Reaching Agreements

- How do self-interested agents reach agreements?
- In an extreme case (zero sum encounter) no agreement is possible — but in most scenarios, a mutually beneficial agreement can be concluded
- The capabilities of negotiation and argumentation are central to the ability of an agent to reach such agreements.
  - Consider an offer as $\vec{v_i}$ s.t. $v_i \in \mathbb{R}^n$
  - Valuation is then $\Sigma_{i=0}^{n} w_i v_i$ such that given a threshold value, a decision can be made
  - Simple negotiation requires each agent to change $\vec{v_i}$ such that the valuation monotonically approaches the threshold
  - Argumentation is the process of one agent getting another to change its $\vec{w_i}$

Protocols and Strategies

- Negotiation is governed by a particular mechanism, or protocol.
- The mechanism defines the “rules of encounter” between agents.
  - Auctions are a large class of “useful” mechanisms
- Mechanism design is the process of designing mechanisms so that they have certain desirable properties.
- Given a particular protocol, how can a particular strategy be designed that individual agents can use?
  - What is the *dominant* strategy for a particular mechanism?

Content

Auctions

- An auction takes place between an agent known as the auctioneer and a collection of agents known as the bidders.
- The goal of the auction is for the auctioneer to allocate the good to one of the bidders.
- In most settings the auctioneer desires to maximise the price; bidders desire to minimise price.

Auction Parameters

- Goods can have:

private value	OR	public/ common value	OR	correlated value

- Winner may pay:

first price OR second price OR n^{th} price

- Bids may be:

open cry OR sealed bid

- Bidding may be:

one shot OR ascending OR descending

English Auctions

- English auction characteristics:
  - first-price,
  - open cry,
  - ascending.
- Susceptible to:
  - Winner's curse
  - Shills
- Dominant strategy is for agent successively to bid a small amount more than the current highest bid until it reaches their valuation, then withdraw.

Dutch Auctions

- Dutch auctions characteristics:
  - open-cry
  - descending
  - auctioneer starts price at artificially high value;
  - auctioneer lowers offer price until some agent makes a bid equal to the current offer price;
  - the good is then allocated to the agent that made the offer.
- Best strategy is to bid only at own valuation

Sealed-Bid Auctions

- Sealed bid auction characteristics:
  - first-price
  - sealed-bid
  - one-shot
- Single round;
- Bidders submit a sealed bid for the good;
- Good is allocated to agent that made highest bid.
- Winner pays price of highest bid.
- Best strategy is to bid less than true valuation.

Vickrey Auctions

- Vickrey auctions characteristics:
  - second-price
  - sealed-bid
  - one-shot
- Good is awarded to the agent that made the highest bid; at the price of the second highest bid.
- Vickrey auctions susceptible to antisocial behavior: untruthful bids can distort market
- Bidding to your true valuation is dominant strategy in Vickrey auctions.
  - Overbid $\rightsquigarrow$ risk of paying above valuation
  - Underbid $\rightsquigarrow$ reduced chance of success

Continuous Double Auction

- Perhaps overlooked because it is simple, but it is also effective and the basis for many real-world mechanisms
- Given buyer B and seller S, proceed in rounds:
  - Step 1: The seller announces an offer price p_1
  - Step 2: The buyer announces a bid price p_2, assume $p_1 > p_2$
  - Step 3: If $p_2 \geq p_1$, sale is agreed at $\frac{p_1+p_2}{2}$
  - Step 4: Seller *reduces* offer giving $p_i, i \in 3, 5, \ldots$
  - Step 5: Buyer *increases* bid giving $p_j, j \in 4, 6, \ldots$
  - Return to Step 3
- Description in terms of two agents, but readily adapts for multiple buyers and multiple sellers
- Example of a mechanism that is more generally applicable in an electronic than a physical setting.

Zero Intelligence Traders I

- Original idea set out in [Gode and Sunder, 1993]
- Trader submits random bids and offers
- Simulations using experimental CDA markets demonstrate the transaction price time-series is *human-like*:
  - Appearing to converge to the theoretical equilibrium price
  - Yielding allocative efficiency comparable to human markets
- Experiments with combinations of three kinds of agents:
  - ZI-U: unconstrained agents, settle at any price
  - ZI-C: constrained agents, may not make a loss
  - human agents

Zero Intelligence Traders II

- Conclusions:
  - No intelligence needed to trade in a CDA as long as not permitted to trade at a loss
  - Market structure ensures allocative efficiency—the ‘invisible’ hand? (Smith)
- [Cliff and Bruten, 1997] identified pathological market conditions for ZI traders, supported by empirical results, specifically:
  1. symmetric supply and demand
  2. flat supply
  3. flat supply and demand curves with excess supply
  4. flat supply and demand curves with excess demand
- ZI-C only converges to equilibrium price in case 1
- Need memory + adaptation for cases 2–4

Zero Intelligence Plus (ZIP)

- Augment ZI-traders with basic machine-learning mechanism
- ZIP traders adapt their profit margin on the basis of four factors:
  - whether an agent still needs to buy or sell
  - was the last quote an offer (seller) or a bid (buyer)
  - was the last quote accepted or rejected
  - was the last quote bigger or smaller than own quote
- At time t trader i calculates
  - the shout price $p_i(t)$ for a unit j with
  - limit price $\lambda_{i,j}$ using
  - profit-margin $\mu_i(t)$, such that $p_i(t) = \lambda_{i,j}(1 + \mu_i(t))$

ZIP profit margin algorithm

- For sellers:
  1. if last shout accepted at price q then
     1. any seller s_i whose $p_i \leq q$ raises its margin
     2. if last shout was a bid then any seller s_i whose $p_i \geq q$ lowers its margin
  2. else
     1. if last shout was an offer then any seller s_i whose $p_i \geq q$ lowers its margin
- For buyers:
  1. if last shout accepted at price q then
     1. any buyer b_i whose $p_i \geq q$ raises its margin
     2. if last shout was an offer then any buyer b_i whose $p_i \leq q$ lowers its margin
  2. else
     1. if last shout was a bid then any buyer b_i whose $p_i \leq q$ lowers its margin

ZIP Adaptation

- Adapation arises from the alteration of the profit margin using the Widrow-Hoff “delta rule”, widely used in back-propagation: $A(t+1) = A(t) + \Delta(t)$
- $\Delta(t)$ is the change in output, determined by the product of a learning rate β and the difference between $A(t)$ and the desired output at time t, denoted $D(t)$: $\Delta(t) = \beta(D(t) - A(t))$
- if $D(t)$ is constant the update rule gives asymptotic convergence of $A(t)$ to $D(t)$ at a speed determined by β.
- When a trader has to change its profit margin, compute a target price $\tau_i(t)$ and use the Widrow-Hoff rule to compute the shout price at the next time step, $p_i(t+1)$
- Full details in [Cliff and Bruten, 1997]

Combinatorial Auctions I

- Auctions supposed to achieve (economically) efficient, effective allocation
- Preceding mechanisms depend on “intelligent” and “rational” bidding
- But need complete information for a Pareto optimal allocation
- CA aims to achieve optimal allocation by putting all the decision making in the auctioneer, hence we have
- The Winner Determination Problem: given a set of bids in a CA, find an allocation of items (not necessarily all) to bidders to maximize revenue

Combinatorial Auctions II

- Revenue is maximized by the allocation that maximizes the sum over all bidders of the bidders' valuations for the subset of items they receive.
- Bids are specified as valuations for a subset of items—called a *bundle*
- WDP can be written as an integer linear program that is equivalent to the weighted set packing problem and hence NP-hard.
- Problem is hard because need to check for each subset of the bids whether the subset is feasible (no bids share items) and how much revenue results... For k bids, there are 2^k subsets.

Combinatorial Auctions III

- Example: given *n* items for sale (1,2,3,4,5,6,7,8,9)

Bidder	Bid	Bundle
1	45	1,2
2	98	1,4,7,8
3	86	9,4,5,1,2
4	62	9,4

- Looking at individual bids, the revenue from bid 2 is greatest
- But bids 1 and 4 are for non-overlapping bundles, and so can *both* be satisfied and maximizes revenue by selling elements 1,2,4,9 for 107 (45+62).

Combinatorial Auctions IV

- (Naïve) Algorithmic solution builds a matrix of all possible combinations, then searches for sets that generate the greatest utility.
  - Take first combination
  - Then examine any other combinations that
    1. match and
    2. have a higher score,

    until done
- Practical solvers essentially take the same approach, but have good heuristics for pruning the search tree.

Which auction?

- For bundles: CA is optimal, but not necessarily practical
- For simple encounters: CDA is effective
- Economic theory says there is no difference between the rest in general, although individuals may differ

Content

Negotiation

- Auctions good for resource allocation
- Need negotiation for reaching agreements
- Characteristics of a negotiation:
  - A negotiation set: possible proposals agents can make
  - A protocol: moves in the negotiation process
  - Strategies, individual and private to each agent
  - A rule is deal struck and what is it
- Negotiation usually proceeds in a series of rounds
- Each agent makes a proposal at each round

Negotiation scenario

- A delivery problem:
  - Agent A_1 has two goods, G_1 and G_2
  - Agent A_2 has two goods, G_3 and G_4
  - Goods G_1 and G_3 need to be delivered to location L_1, and
  - Goods G_2 and G_4 to L_2
- Each good delivery problem is a task
- Worst case is that each agent goes to L_1 and L_2
- Better to exchange one task so one agent goes to L_1 and the other to L_2
- Task-oriented domains formalize this kind of problem

Task Oriented Domains

- A task-oriented domain (TOD) comprises:
  - $\mathcal{T} = t_1, \ldots, t_m$: the set of tasks
  - $\mathcal{A} = a_1, \ldots, a_n$: the set of agents
  - $c : 2^{\mathcal{T}} \rightarrow \mathbb{R}^+$ is cost of executing each subset of tasks, where $2^{\mathcal{T}}$ denotes all the finite subsets of $\mathcal{T}$
  - The cost function is *monotonic*: adding tasks never decreases the cost, i.e.

$$\mathcal{T}_i, \mathcal{T}_j \subseteq \mathcal{T} \text{ and } \mathcal{T}_i \subseteq \mathcal{T}_j \Rightarrow c(\mathcal{T}_1) \leq c(\mathcal{T}_2)$$

- An encounter is the assignment of an ordered list of tasks $T_1, \ldots, T_n$, such that $T_i \in \mathcal{T}$, to agent $a_i \in \mathcal{A}$
- But can agent A_i do better by negotiating a reallocation of its tasks?

Deals in TODs

- Given agents $\{A_1, A_2\}$ and encounter $\langle T_1, T_2 \rangle$:
  - A deal allocates tasks $T_1 \cup T_2$ to agents a_1 and a_2
  - In $\delta = \langle D_1, D_2 \rangle$, agent a_i is allocated tasks D_i with no tasks left over: $D_1 \cup D_2 = T_1 \cup T_2$
  - The cost of the deal to agent a_i is $c(D_i)$
- The utility of deal δ to agent a_i:

$$\text{utility}_i(D) = c(T_i) - c(D_i)$$

- In the absence of (re-)allocation, agents take the conflict deal, $\Theta = \langle T_1, T_2 \rangle$:

$$\text{utility}_i(\Theta) = 0, \ \forall a_i \in \mathcal{A}$$

- Deal δ is individual rational if it has positive utility

Deal Dominance

- The deal $\delta_1 \succ \delta_2$ (dominates) iff:
  - C1: Deal δ_1 is at least as good as δ_2 for every agent:

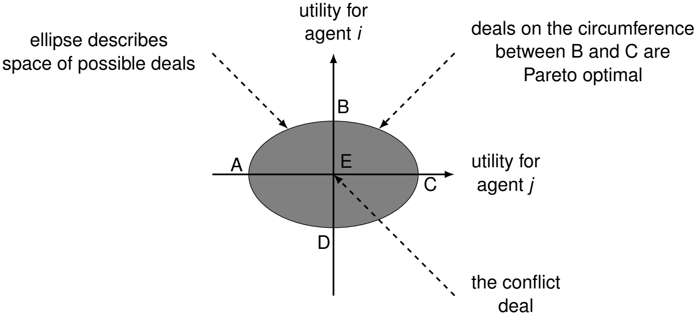
$$\bigwedge_{i=1}^{n} utility_i(\delta_1) \geq utility_i(\delta_2)$$

  - C2: Deal δ_1 is better than δ_2 for some agent:

$$\bigvee_{i=1}^{n} utility_i(\delta_1) > utility_i(\delta_2)$$

- A deal $\delta_1 \succeq \delta_2$ (weakly dominates) if at least C1 holds
- A deal is Pareto optimal if $\nexists \delta_i$ such that $\forall j \delta_i \succ \delta_j, j \neq i$
- A deal δ is individual rational if $\delta \succeq \Theta$
- The negotiation set is the set of deals that are:
  - individual rational *and*
  - Pareto efficient

The negotiation set



- ABCD = possible deals
- utility(E) $>$ all deals left of B-D for agent *j*
- utility(E) $>$ all deals below A-C for agent *i*

The Monotonic Concession Protocol

- Negotiation proceeds in rounds
  1. Round 1: agents simultaneously propose a deal from the negotiation set
  2. Agreement if one agent finds that the deal proposed by the other is at least as good or better than its proposal
  3. Otherwise, a new round of simultaneous proposals
  4. At each round, no agent may propose a deal is less preferred by the other agent than previously, i.e. utility must increase
  5. If neither agent makes a concession, then negotiation terminates with the conflict deal.

The Zeuthen Strategy

Three questions:

- What should an agent's first proposal be?
  Its most preferred deal
- On any given round, who should concede?
  The agent least willing to risk conflict.
- If an agent concedes, then how much should it concede?
  Just enough to change the balance of risk.

Willingness to Risk Conflict

- Suppose agent has conceded a lot, then:
  - Its proposal is close to conflict deal
  - If conflict deal occurs, it is not much worse off
  - Therefore, it is willing to risk conflict
- Propensity to risk conflict rises as utility(δ) – utility(Θ) $\rightarrow$ 0

Nash Equilibrium

The Zeuthen strategy is in Nash equilibrium: if one agent uses Zeuthen, the other cannot do better than use the same itself...

- Important for the design of automated agents
- No need for secrecy about strategy
- Situation cannot be exploited by using a different strategy

Returning to the scenario

- The tasks are: $T_1 = G_1 \to L_1 \quad T_2 = G_2 \to L_2$
  $T_3 = G_3 \to L_1 \quad T_4 = G_4 \to L_2$
- With encounter $\langle\{T_1, T_2\}, \{T_3, T_4\}\rangle$
- Let $c(\mathcal{T}_1) = 5 + 4$ and $c(\mathcal{T}_2) = 5 + 4$
- For $\delta_1 = \langle\{T_1, T_3\}, \{T_2, T_4\}$, let $c(\mathcal{T}_1) = 5$ and $c(\mathcal{T}_2) = 4$
- Hence utility$_1(\delta_1) = 9 - 5 = 4$ and utility$_2(\delta_1) = 9 - 4 = 5$
- Thus δ_1 is individual rational
- and Pareto optimal
- as is $\delta_2 = \langle\{T_2, T_4\}, \{T_1, T_3\}$

Collaboration

- Why and how to get agents work together? Motivations:
  - task sharing: components of a task are distributed to competent agents
  - result sharing: information (partial results etc) is distributed.
- Two scenarios: benevolent and self-interested

Benevolent Agents

- If we “own” the whole system, we can design agents to help each other whenever asked.
- Can assume agents are benevolent: our best interest is their best interest.
- Problem-solving in benevolent systems is cooperative distributed problem solving (CDPS).
- Benevolence simplifies the system design task enormously!

Self-Interested Agents

- If agents represent individuals or organisations, (the more general case), then cannot assume benevolence
- Assume agents to act to further their own interests, possibly at expense of others
- Potential for conflict.
- May complicate the design task enormously.
- Contract net: mechanism for cooperation among self-interested agents

Contract Net Protocol

- Basic contract net protocol:
  1. Recognition: agent cannot or prefers not to work alone
  2. Announcement:
     - Description of task itself (maybe executable)
     - Any constraints (e.g., deadlines, quality constraints).
     - Meta-task information (e.g., "bids must be submitted by... ")
  3. Bidding:
     - Can agent complete task?
     - Assess quality constraints and price
  4. Awarding + expediting
     - Initiator evaluates bids and awards contract
     - Broadcast decision to bidders
     - Successful contractor then carries out task
- Variants:
  - Recursive: potential bidder issues a CFP etc.
  - Iterated: do not reject bids, but negotiate deals

Content

Summary

- Conventional auctions
- Zero Intelligence Traders
- Combinatorial auctions
- Task Oriented Domains
- Contract net protocol

Recommended Reading

- "An Introduction to Multiagent Systems" [Wooldridge, 2009]:
  - Chapter 6, 7 and 9
  - Chapter 14, pp299-310, combinatorial auctions
- Two papers [Bigham and Du, 2003] and [Bussmann and Schild, 2000] illustrate applications of negotiation
- Cramton [Cramton et al., 2005] examines combinatorial auctions in exhaustive detail, wherein Lehmann and Sandholm [Lehmann et al., 2005] discuss the Winner Determination Problem.
- [Zlotkin and Rosenschein, 1993] is the original TOD paper.
- [Cliff and Bruten, 1997] describes ZIP traders.

References I

Bigham, J. and Du, L. (2003).
Cooperative negotiation in a multi-agent system for real-time load balancing of a mobile cellular network.
In *AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 568–575. ACM Press.

Bussmann, S. and Schild, K. (2000).
Self-organizing manufacturing control: An industrial application of agent technology.
In *ICMAS*, pages 87–94. IEEE Computer Society.

Cliff, D. and Bruten, J. (1997).
Minimal-intelligence agents for bargaining behaviours in market-based environments.
Technical Report HPL-97-91, Hewlett-Packard Laboratories.
Available via `http://www.hpl.hp.com/techreports/97/HPL-97-91.html`, last accessed November 2007.

References II

- Cramton, P., Shoham, Y., and Steinberg, R., editors (2005).
  Combinatorial Auctions.
  MIT Press.
  ISBN: 0-262-03342-9.
- Gode, D. K. and Sunder, S. (1993).
  Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality.
  The Journal of Political Economy, 101(1):119–137.
- Lehmann, D., Müller, R., and Sandholm, T. (2005).
  The Winner Determination Problem, chapter 12.
  MIT Press.
  ISBN: 0-262-03342-9. This chapter available via http://www.cs.cmu.edu/~sandholm/winner-determination-final.pdf.
- Wooldridge, M. (2009).
  An introduction to multiagent systems (second edition).
  Wiley.
  ISBN: 978-0-470-51946-2.

References III

Zlotkin, G. and Rosenschein, J. S. (1993).
A domain theory for task oriented negotiation.
In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, pages 416–422, Chambery, France.