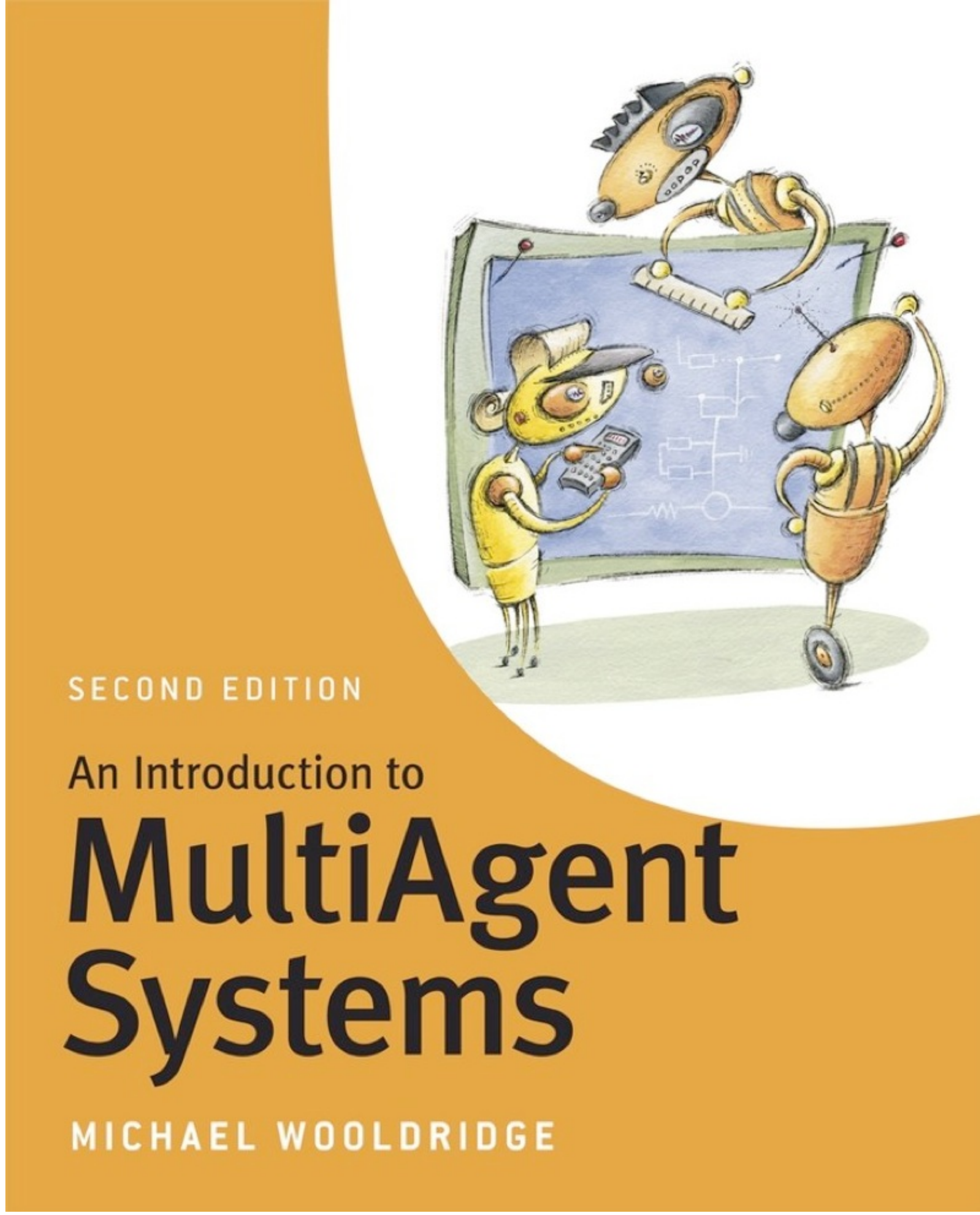
COMP310
Multi-Agent Systems
Chapter 14 - Allocating Scarce Resources

Dr Terry R. Payne
Department of Computer Science
UNIVERSITY OF LIVERPOOL



Overview

- Allocation of scarce resources amongst a number of agents is central to multiagent systems.

- A resource might be:
  - a physical object
  - the right to use land
  - computational resources (processor, memory, . . .)

- It is a question of supply vs demand
  - If the ***resource isn't scarce***…, or if there is ***no competition*** for the resource...
    - ...then there is no trouble allocating it
  - If there is a ***greater demand than supply***
    - Then we need to determine how to allocate it

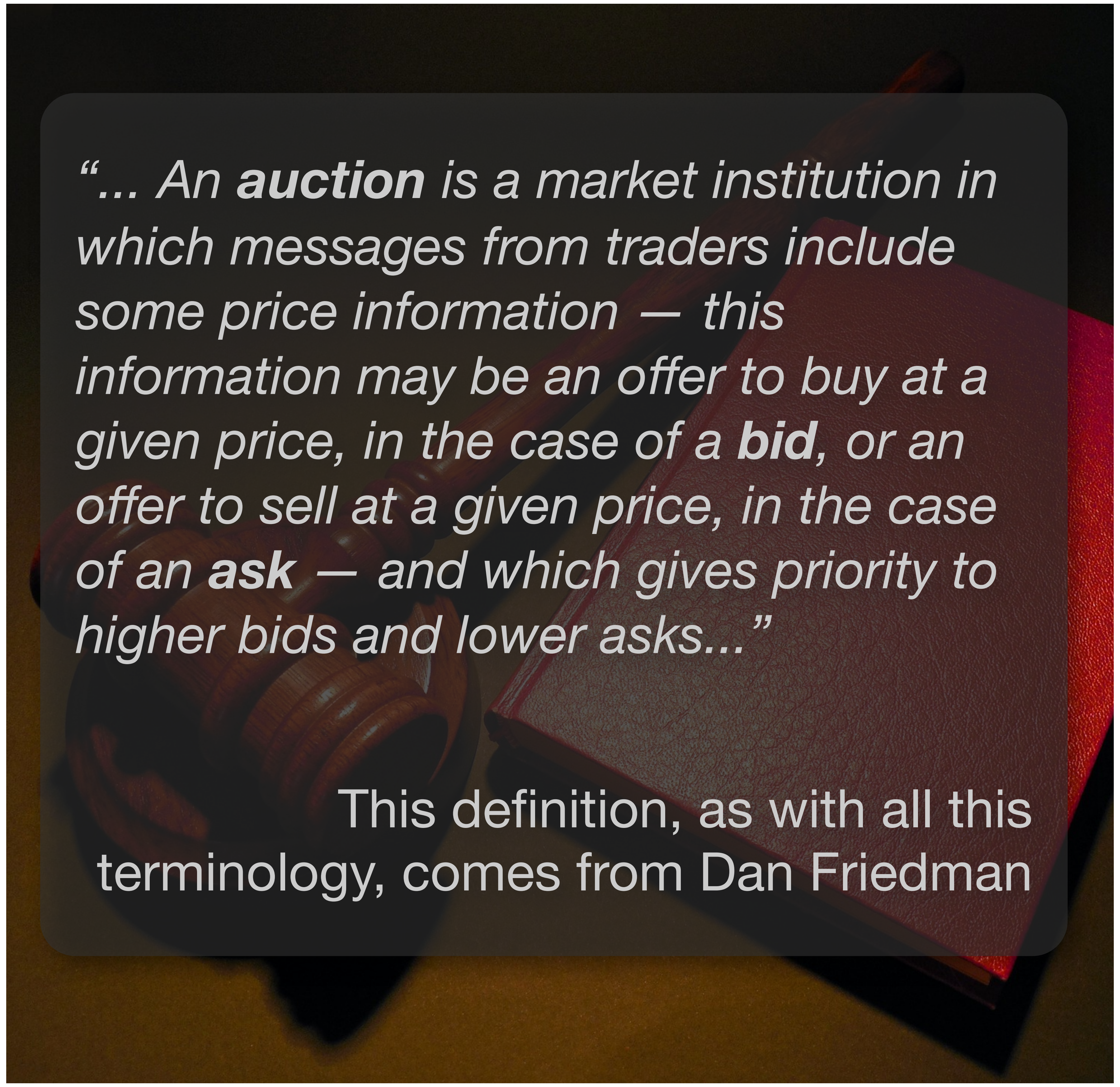
Overview

- In practice, this means we will be talking about auctions.
  - These used to be rare (and not so long ago).
  - However, auctions have grown massively with the Web/Internet
    - Frictionless commerce

- Now feasible to auction things that weren't previously profitable:
  - eBay
  - Adword auctions

What is an auction

- Auctions are effective in allocating resources efficiently
  - They also can be used to reveal truths about bidders
- Concerned with ***traders*** and their allocations of:
  - Units of an indivisible ***good***; and
  - Money, which is divisible.
- Assume some initial allocation.
- ***Exchange*** is the free alteration of allocations of goods and money between traders



Types of value

- There are several models, embodying different assumptions about the nature of the good.
  - Private Value / Common Value / Correlated Value
    - With a common value, there is a risk that the winner will suffer from the ***winner's curse***, where the winning bid in an auction exceeds the intrinsic value or true worth of an item
- Each trader has a value or ***limit price*** that they place on the good.
  - Limit prices have an effect on the behaviour of traders

Private Value

Good has an value to me that is independent of what it is worth to you.

- *For example: John Lennon's last dollar bill.*

Common Value

The good has the same value to all of us, but we have differing estimates of what it is.

- *Winner's curse.*

Correlated Value

Our values are related.

- *The more you're prepared to pay, the more I should be prepared to pay.*

Auction Protocol Dimensions

- ***Winner Determination***
  - Who gets the good, and what do they pay?
    - e.g. first vs second price auctions
- ***Open Cry*** vs ***Sealed-bid***
  - Are the bids public knowledge?
    - Can agents exploit this public knowledge in future bids?
- ***One-shot*** vs ***Iterated Bids***
  - Is there a single bid (i.e. one-shot), after which the good is allocated?
  - If multiple bids are permitted, then:
    - Does the price ascend, or descend?
    - What is the terminating condition?

English Auction

- This is the kind of auction everyone knows.
  - Typical example is sell-side.
- Buyers call out bids, bids increase in price.
  - In some instances the auctioneer may call out prices with buyers indicating they agree to such a price.
- The seller may set a ***reserve price***, the lowest acceptable price.
- Auction ends:
  - at a fixed time (internet auctions); or when there is no more bidding activity.
  - The “last man standing” pays their bid.

English Auction

Classified in the terms we used above:

- ***First-price***
- ***Open-cry***
- ***Ascending***

Around 95% of internet auctions are of this kind. The classic use is the sale of antiques and artwork.

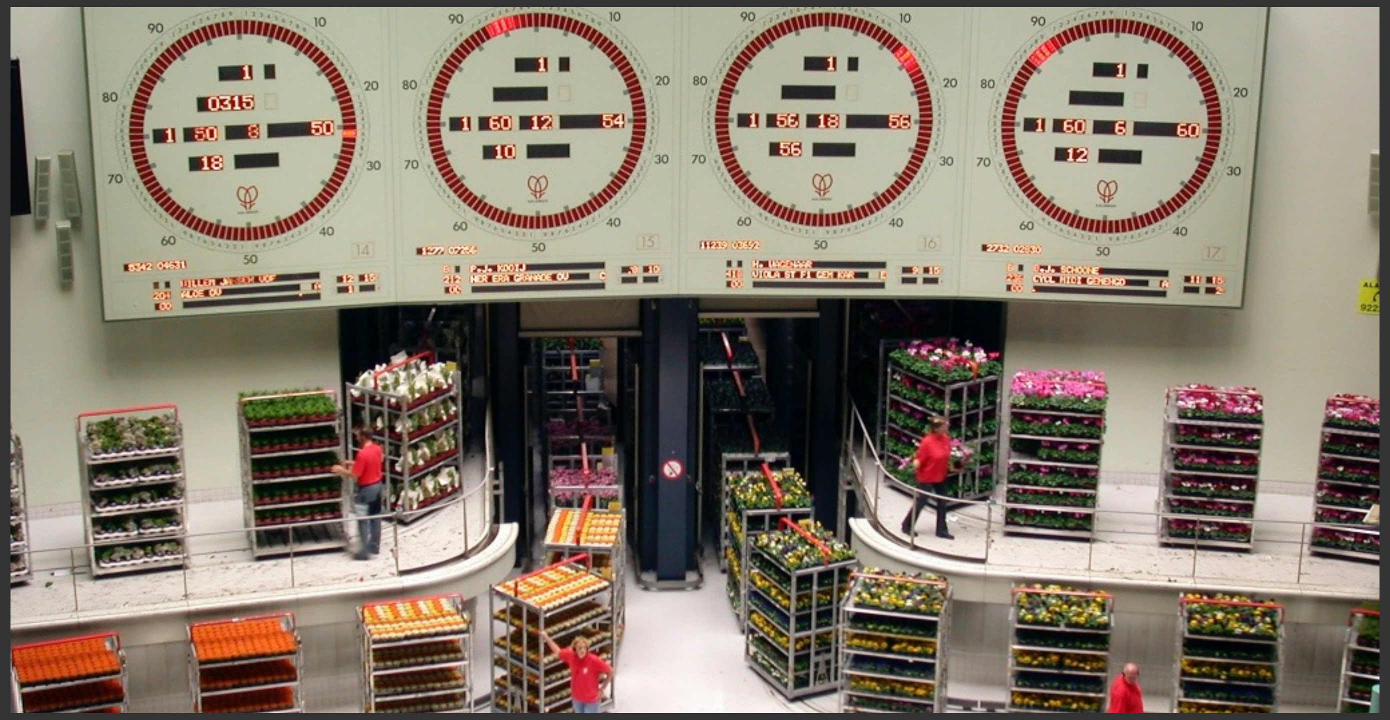
Susceptible to:

- *Winner’s curse*
- *Shills*

Dutch Auction

- Also called a “descending clock” auction
  - Some auctions use a clock to display the prices.
- Starts at a ***high price***, and the auctioneer calls out ***descending prices***.
  - One bidder claims the good by indicating the current price is acceptable.
    - *Ties are broken* by restarting the descent from a slightly higher price than the tie occurred at.
- The winner pays the price at which they “stop the clock”.

Dutch Auction



Classified in the terms we used above:

- ***First-price***
- ***Open-cry***
- ***Descending***

High volume (since auction proceeds swiftly). Often used to sell perishable goods:

- *Flowers in the Netherlands (eg. Aalsmeer)*
- *Fish in Spain and Israel.*
- *Tobacco in Canada.*

First-Price Sealed-Bid Auction

- In an English auction, you get information about how much a good is worth.
  - Other people's bids tell you things about the market.
- In a ***sealed bid auction***, none of that happens
  - at most you know the winning price after the auction.
- In the First-Price Sealed-Bid (FPSB) auction the ***highest bid wins as always***
  - As its name suggests, the winner pays that highest price (which is what they bid).

FPSB

Classified in the terms we used above:

- ***First-price***
- ***Sealed Bid***
- ***One-shot***

Governments often use this mechanism to sell treasury bonds (the UK still does, although the US recently changed to Second-Price sealed Bids).

Property can also be sold this way (as in Scotland).

Vickrey Auction

- The Vickrey auction is a sealed bid auction.
  - The winning bid is the ***highest bid***, but the winning bidder ***pays the amount of the second highest bid***.
- This sounds odd, but it is actually a very smart design.
  - Will talk about why it works later.
- It is ***in the bidders' interest to bid their true value***.
  - ***incentive compatible*** in the usual terminology.
- However, it is not a panacea, as the New Zealand government found out in selling radio spectrum rights
  - Due to interdependencies in the rights, that led to strategic bidding,
    - one firm bid NZ$100,000 for a license, and paid the second-highest price of only NZ$6.

Vickrey Auction

Classified in the terms we used above:

- ***Second-price***
- ***Sealed Bid***
- ***One-shot***

Historically used in the sale of stamps and other paper collectibles.

Why does the Vickrey auction work?

- Suppose you bid more than your valuation.
  - You may win the good.
  - If you do, you may end up paying more than you think the good is worth.
  - Not so smart.
- Suppose you bid less than your valuation.
  - You stand less chance of winning the good.
  - However, even if you do win it, you will end up paying the same.
  - Not so smart.

Proof of dominance of truthful bidding

- Let v_i be the bidding agent i's value for an item, and b_i be the agent's bid
  - The payoff for bidder i is:

$$p_i = \begin{cases} v_i - max_{j \neq i} b_j & \text{if } b_i > max_{j \neq i} b_j \\ 0 & \text{otherwise} \end{cases}$$

- Assume bidder i bids $b_i > v_i$ (i.e. ***overbids***)
  - If $max_{j \neq i}\ b_j < v_i$, then the bidder would win whether or not the bid was truthful. Therefore the strategies of bidding truthfully and overbidding have equal payoffs
  - If $max_{j \neq i}\ b_j > b_i$, then the bidder would loose whether or not the bid was truthful. Again, both strategies have equal payoffs
  - If $v_i < max_{j \neq i}\ b_j < b_i$, then the strategy of overbidding would win the action, but the payoff would be negative (as the bidder will have overpaid). A truthful strategy would pay zero.

Proof of dominance of truthful bidding

- Let v_i be the bidding agent i's value for an item, and b_i be the agent's bid
  - The payoff for bidder i is:

$$p_i = \begin{cases} v_i - max_{j \neq i} b_j & \text{if } b_i > max_{j \neq i} b_j \\ 0 & \text{otherwise} \end{cases}$$

- Assume bidder i bids $b_i < v_i$ (i.e. ***underbids***)
  - If $max_{j \neq i}\ b_j > v_i$, then the bidder would loose whether or not the bid was truthful. Therefore the strategies of bidding truthfully and underbidding have equal payoffs
  - If $max_{j \neq i}\ b_j < b_i$, then the bidder would win whether or not the bid was truthful. Again, both strategies have equal payoffs
  - If $b_i < max_{j \neq i}\ b_j < v_i$, then only the strategy of truthtelling would win the action, with a positive payoff (as the bidder would have). An underbidding strategy would pay zero.

Not examined in 2017-2018

Collusion

- None of the auction types discussed so far are immune to collusion
  - A ***grand coalition of bidders*** can agree beforehand to collude
    - Propose to artificially lower bids for a good
    - Obtain true value for good
    - Share the profit
  - An auctioneer could employ bogus bidders
    - ***Shills*** could artificially increase bids in open cry auctions
    - Can result in ***winners curse***

Combinatorial Auctions

- A combinatorial auction is an ***auction for bundles of goods***.
  - A good example of bundles of goods are spectrum licences.
  - For the 1.7 to 1.72 GHz band for Brooklyn to be useful, you need a license for Manhattan, Queens, Staten Island.
  - Most valuable are the licenses for the same bandwidth.
  - But a different bandwidth license is more valuable than no license
    - a phone will work, but will be more expensive.
- (The FCC spectrum auctions, however, did not use a combinatorial auction mechanism)

Combinatorial Auctions

- Define a set of items to be auctioned as:

- Given a set of agents $Ag = \{1,\ldots,n\}$, the preferences of agent i are given with the ***valuation function*** opposite:
  - If that sounds to you like it would place a big requirement on agents to specify all those preferences, you would be right.
  - If $v_i(\emptyset) = 0$, then we say that the valuation function for i is ***normalised***.
    - i.e. Agent i does not get any value from an empty allocation

- Another useful idea is ***free disposal***:
  - In other words, an agent is never worse off having more stuff.

Set of items for auction

$$\mathcal{Z} = \{Z_1, \ldots, Z_m\}$$

Valuation Function

$$v_i : 2^{\mathcal{Z}} \rightarrow \mathbb{R}$$

meaning that for every possible bundle of goods $Z \subseteq \mathcal{Z}$, $v_i(Z)$ says how much Z is worth to i.

Free Disposal

$Z_1 \subseteq Z_2$ implies $v_i(Z_1) \leq v_i(Z_2)$

Allocation of Goods

- An outcome is an allocation of goods to the agents.
  - Note that we don't require all off the goods to be allocated
  - Formally an allocation is a list of sets $Z_1, \ldots Z_n$, one for each agent Ag_i such that $Z_i \subseteq \mathcal{Z}$
  - and for all $i,j \in Ag$ such that $i \neq j$, we have $Z_i \cap Z_j = \emptyset$.
    - Thus no good is allocated to more than one agent.

- The set of all allocations of Z to agents Ag is: $alloc(\mathcal{Z}, Ag)$

Maximising Social Welfare

- If we design the auction, we get to say how the allocation is determined.
  - Combinatorial auctions can be viewed as different social choice functions, with different outcomes relating to different allocations of goods
  - A desirable property would be to maximize social welfare.
    - i.e. maximise the sum of the utilities of all the agents.
- We can define a social welfare function:

$$sw(\underbrace{Z_1, \ldots, Z_n}_{\text{allocations}}, \underbrace{v_1, \ldots, v_n}_{\text{valuations}}) = \sum_{i=1}^{n} v_i(Z_i)$$

Defining a Combinatorial Auction

- Given this, we can define a combinatorial auction.
  - Given a set of goods Z and a collection of valuation functions $v_1,\ldots,v_n$, one for each agent $i \in Ag$, the goal is to find an allocation $Z_1^*, \ldots Z_n^*$ that maximises sw:

$$Z_1^*, \ldots, Z_n^* = \underset{(Z_1,\ldots,Z_n) \in alloc(\mathcal{Z}, Ag)}{argmax} \quad sw(Z_1, \ldots, Z_n, v_1, \ldots, v_n)$$

- Figuring this out is called the ***winner determination*** problem.

Winner Determination

- How do we do this?
- Well, we could get every agent i to declare their valuation: $\hat{v}_i$
  - The hat denotes that this is what the agent says, not what it necessarily is.
    - Remember that the agent may lie!
- Then we just look at all the possible allocations and figure out what the best one is.

- One problem here is representation, valuations are exponential: $v_i : 2^{\mathcal{Z}} \rightarrow \mathbb{R}$
  - A naive representation is impractical.
  - In a bandwidth auction with 1122 licenses we would have to specify 2^{1122} values for each bidder.
- Searching through them is computationally intractable

Bidding Languages

- Rather than exhaustive evaluations, allow bidders to construct valuations from the bits they want to mention.
  - An atomic bid β is a pair (Z, p) where $Z \subseteq \mathcal{Z}$
  - A bundle Z' ***satisfies*** a bid (Z, p) if $Z \subseteq Z'$.
- In other words a bundle ***satisfies*** a bid if it contains at least the things in the bid.
- Atomic bids ***define valuations***

$$v_\beta(Z') = \begin{cases} p & \text{if } Z' \text{ satisfies } (Z, p) \\ 0 & \text{otherwise} \end{cases}$$

- Atomic bids alone don't allow us to construct very interesting valuations.

XOR Bids

- With XOR bids, ***we pay for at most one***
  - A bid $\beta = (Z_1, p_1)$ *XOR* ... *XOR* (Z_k, p_k) defines a valuation function v_β as follows:

$$v_\beta(Z') = \begin{cases} 0 & \text{if } Z' \text{ does not satisfy any } (Z_i, p_i) \\ \max\{p_i | Z_i \subseteq Z'\} & \text{otherwise} \end{cases}$$

  - I pay nothing if your allocation Z' doesn't satisfy any of my bids
  - Otherwise, I will pay the largest price of any of the satisfied bids.
- XOR bids are ***fully expressive***, that is they can express any valuation function over a set of goods.
  - To do that, we may need an exponentially large number of atomic bids.
    - However, the valuation of a bundle can be computed in polynomial time.

$B_1 = (\{a,b\}, 3)$ *XOR* $(\{c, d\}, 5)$

"…I would pay 3 for a bundle that contains a and b but not c and d. I will pay 5 for a bundle that contains c and d but not a and b, and I will pay 5 for a bundle that contains a, b, c and d..."

From this we can construct the valuation:

$$\begin{aligned} v_{\beta_1}(\{a\}) &= 0 \\ v_{\beta_1}(\{b\}) &= 0 \\ v_{\beta_1}(\{a,b\}) &= 3 \\ v_{\beta_1}(\{c,d\}) &= 5 \\ v_{\beta_1}(\{a,b,c,d\}) &= 5 \end{aligned}$$

Not examined in 2017-2018

OR Bids

- With OR bids, ***we are prepared to pay for more than one bundle***
  - A bid $\beta = (Z_1, p_1)\ OR\ \ldots\ OR\ (Z_k, p_k)$ defines k valuations for different bundles
  - An allocation of goods Z' is assigned given a set W of atomic bids such that:
    - Every bid in W is satisfied by Z'
    - No goods appear in more than one bundle; i.e. $Z_i \cap Zj = \varnothing$ for all i,i where $i \neq j$
    - No other subset W' satisfying the above condition is better:

$$\sum_{(Z_i, p_i) \in W'} p_i > \sum_{(Z_j, p_j) \in W'} p_j$$

$B_1 = (\{a,b\}, 3)\ OR\ (\{c, d\}, 5)$

"…I would pay 3 for a bundle that contains a and b but not c and d. I will pay 5 for a bundle that contains c and d but not a and b, and I will pay 8 for both bundles that contain a combination of a, b, c and d..."

From this we can construct the valuation:

$$\begin{aligned} v_{\beta_1}(\{a\}) &= 0 \\ v_{\beta_1}(\{b\}) &= 0 \\ v_{\beta_1}(\{a,b\}) &= 3 \\ v_{\beta_1}(\{c,d\}) &= 5 \\ v_{\beta_1}(\{a,b,c,d\}) &= 8 \end{aligned}$$

Note that the **cost of the last bundle is different to that when the XOR bid** was used

Not examined in 2017-2018

OR Bids

- Here is another example!
  - B_3 =({e, f, g}, 4) OR ({f, g}, 1) OR ({e}, 3) OR ({c, d}, 4)

- This gives us:

$$\begin{aligned}
\upsilon_{\beta_3}(\{e\}) &= 3\\
\upsilon_{\beta_3}(\{e,f\}) &= 3\\
\upsilon_{\beta_3}(\{e,f,g\}) &= 4\\
\upsilon_{\beta_3}(\{b,c,d,f,g\}) &= 4+1=5\\
\upsilon_{\beta_3}(\{a,b,c,d,e,f,g\}) &= 4+4=8\\
\upsilon_{\beta_3}(\{c,d,e\}) &= 4+3=7
\end{aligned}$$

- Remember that if more than one bundle is satisfied, then you pay for each of the bundles satisfied.
  - Also remember free disposal, which is why the bundle {e,f} satisfies the bid ({e}, 3) as the agent doesn't pay extra for f

Not examined in 2017-2018

OR Bids

- OR bids are ***strictly less expressive*** than XOR bids
  - Some valuation functions cannot be expressed:
  - $\upsilon(\{a\}) = 1,\ \upsilon(\{b\}) = 1,\ \upsilon(\{a,b\}) = 1$
- OR bids also ***suffer from computational complexity***
  - Given an OR bid β and a bundle Z, computing $\upsilon_\beta(Z)$ is NP-hard

Winner Determination

- Determining the winner is a combinatorial optimisation problem (NP-hard)
  - But this is a worst case result, so it may be possible to develop approaches that are either ***optimal*** and run well in many cases, or ***approximate*** (within some bounds).
- Typical approach is to code the problem as an ***integer linear program*** and use a standard solver.
  - This is NP-hard in principle, but often provides solutions in reasonable time.
  - Several algorithms exist that are efficient in most cases
- Approximate algorithms have been explored
  - Few solutions have been found with reasonable bounds
- Heuristic solutions based on ***greedy algorithms*** have also been investigated
  - e.g. that try to find the largest bid to satisfy, then the next etc

The VCG Mechanism

- Auctions are easy to strategically manipulate
  - In general ***we don't know*** whether the agents valuations ***are true valuations***.
  - Life would be easier if they were…
  - … so can we make them true valuations?
- Yes!
  - In a generalization of the Vickrey auction.
    - Vickrey/Clarke/Groves Mechanism
- Mechanism is incentive compatible: ***telling the truth is a dominant strategy***.

> Recall that we could get every agent i to declare their valuation:
>
> $$\hat{v}_i$$
>
> where the hat denotes that this is what the agent says, not what it necessarily is.
>
> - *The agent may lie!*

The VCG Mechanism

- Need some more notation.
  - ***Indifferent valuation*** function: $v^0(Z) = 0$ for all Z
    - I.e. the value for a bid that doesn't care about the goods
  - sw_{-i} is the ***social welfare function without i***:

$$sw_{-i}(Z_1, \ldots, Z_n, v_1, \ldots, v_n) = \sum_{j \in Ag, j \neq i} v_j(Z_j)$$

  - This is how well everyone ***except agent i*** does from Z_1, ..., Z_n
- And we can then define the VCG mechanism.

The VCG Mechanism

- Every agent simultaneously declares a valuation $\hat{v}_i$
  - Remember that this not be the actual valuation

- The mechanism computes the allocation $Z_1^*, \ldots, Z_n^*$:

$$Z_1^*, \ldots, Z_n^* = argmax_{(Z_1, \ldots, Z_n) \in alloc(\mathcal{Z}, Ag)} sw(Z_1, \ldots, Z_n, \hat{v}_1, \ldots, \hat{v}_i, \ldots, \hat{v}_n)$$

- Each agent i pays p_i
  - This is effectively a ***compensation*** to the other agents for their loss in utility due to i winning an allocation
  - This is the difference in social welfare to agents other than i
    - Between the outcome $Z_1', \ldots, Z_n'$ when i doesn't participate
    - And the outcome $Z_1^*, \ldots, Z_n^*$ when i does participate

$$\begin{aligned} p_i \quad = \quad & sw_{-i}(Z_1', \ldots, Z_n', \hat{v}_1, \ldots, v^0, \ldots, \hat{v}_n) \\ & - sw_{-i}(Z_1^*, \ldots, Z_n^*, \hat{v}_1, \ldots, \hat{v}_i, \ldots, \hat{v}_n) \end{aligned}$$

  - Therefore the mechanism computes, for each agent I the allocation that maximises social welfare were that agent to have declared v^0 to be its valuation:

$$Z_1', \ldots, Z_n' = argmax_{(Z_1, \ldots, Z_n) \in alloc(\mathcal{Z}, Ag)} sw(Z_1, \ldots, Z_n, \hat{v}_1, \ldots, v^0, \ldots, \hat{v}_n)$$

The VCG Mechanism

- With the VCG, each agent pays out the cost (to the other agents) of it having participated in the auction.
  - It is incentive compatible for exactly the same reason as the Vickrey auction was before.
    - No agent can benefit by declaring anything other than its true valuation
  - To understand this, think about VCG with a singleton bundle
    - The only agent that pays anything will be the agent i that has the highest bid
    - But if it had not participated, then the agent with the second highest bid would have won
    - Therefore agent i "*compensates*" the other agents by paying this second highest bid
- Therefore we get a dominant strategy for each agent that guarantees to maximise social welfare.
  - i.e. ***social welfare maximisation can be implemented in dominant strategies***

Summary

- Allocating scarce resources comes down to auctions
  - We looked at a range of different simple auction mechanisms.
    - English auction
    - Dutch auction
    - First price sealed bid
    - Vickrey auction
- The we looked at the popular field of combinatorial auctions.
  - We discussed some of the problems in implementing combinatorial auctions.
- And we talked about the Vickrey/Clarke/Groves mechanism, a rare ray of sunshine on the problems of multiagent interaction.

Class Reading (Chapter 14):

"Expressive commerce and its application to sourcing: How to conduct $35 billion of generalized combinatorial auctions", T. Sandholm. AI Magazine, 28(3): 45-58 (2007).

This gives a detailed case study of a successful company operating in the area of computational combinatorial auctions for industrial procurement.