Lecture notes for Multi-Agent Systems

Danny Kuminov

20-Dec-2006

1 Introduction

This is a summary of the lecture by Moshe Tennenholtz at the MAS course. The normal text (such as this) represents a nearly verbatim (up to mistakes and permutations of sentences) recording of the lecture.

> (DEK) *Any additions/clarifications/open questions added by the transcriber (Danny Kuminov, dannykv@tx.technion.ac.il) will be written like this and prefaced by (DEK).*

2 Exploration vs Exploitation

2.1 R-max continued

The basic concept that R-max is based on is well known in the AI community as the "Optimism under uncertainty" heuristic. It can be formulated as the following assumption (that the agent makes): any state/action is maximally beneficial until proven otherwise. Obviously, this assumption is generally incorrect and may cause the agent to choose a sub-optimal strategy, but this sub-optimal strategy will cause an actual loss of utility only if the agent encounters a situation about which he has incorrect information. Therefore, R-max either explores efficiently or exploits efficiently - the adversary can influence whether we exploit or explore, but it cannot prevent us from doing both. This approach has two advantages over algorithms that resolve the "Exploration vs Exploitation" dilemma explicitly, like Q-learning:

- The decision whether to explore or to exploit usually is controlled by a parameter (which determines the length of the exploration stage relative to the exploitation stage, like T in Q-learning). This parameter often has to be tuned manually for the concrete problem, which is an additional optimization problem all by itself.
- Given a specific value for the parameter, the adversary may exploit his knowledge of it by behaving in such a way (equivalently, the environment in the worst case may behave in such a way) that our algorithm learns

very little in the exploration stage and then, after the exploration stage (which length is controlled by the parameter) ends, this lack of knowledge costs us in the exploitation stage.

(DEK) *IMHO, the "Optimism under uncertainty" principle would be better named "Optimism under incomplete information" - at least in the setting which R-max handles, the uncertainty is a result of the stochastic nature of the environment and the agent is actually neutral with respect to risk (as reflected by the fact that we use the expected actual payoff as the performance criterion, without factoring the variance into the equation or using the maximal/minimal payoff realization, for example).*
A trick question*: Suppose we have the following SG (which is actually as MDP, because the adversary has only one action in all states):*
Two states, in the first state the agent has two actions, one that gives 1 and keeps him in the same state, and the other that transfer him to the second state, in which he will get 0 forever - his only possible action will keep him in the same state. Obviously, the optimal strategy, given that we start in the first state, is to stay there forever and get 1.
How exactly can R-max (or any learning algorithm, for that matter) know that he should never try the second action?
Answer*: He cannot, but he doesn't make any such promise either! Recall that we have defined in the previous lecture the value of a policy as the* minimum over all starting states *of the liminf of the minimal expected average payoff. Under this definition, both the optimal policy and R-max have the same value 0 in the above game.*

Formally, to be sure that R-max will not keep "being surprised for the worst" forever, we have to note that:

- We have only polynomially many parameters to learn
- It is enough to sample each entry in the game matrices a polynomial number of times to get a good estimate of the transition probabilities

Therefore, as the IEE lemma implies, the number of non-exploit T-step executions is polynomially bounded. In addition, when we do exploit (i.e. play "known" entries), our expected reward is ϵ-close to optimal. Finally, we have to compensate for sub-optimal behavior while learning, and a polynomial number of exploitation steps is enough to compensate for this loss (because there where at most polynomially many "bad" steps, and the loss at each of them is bounded).

The R-max algorithm made three assumptions that are worth special attention:

- That the opponent's actions are observable.

- That the value of R_{max} is known a priori. The intuitive solution for this – just pick a "big" default value and if we encounter a bigger one during the game, update all unknown entries with the new value – may not work, because the default value may not be big enough to motivate exploration of "unknown" states, in which bigger values may hide. This assumption can[1] be relaxed in a more careful way, resulting in an algorithm that is polynomial in the true value of R_{max}. This may be good enough for practical purposes, but is scoffed at by the theoretical computer scientists, that note that the value R_{max} is *exponential* in the size of the input that the algorithm receives (the payoffs after each game) and therefore the improved algorithm is exponential in the size of the input.

- That T, the mixing time of the optimal policy, is known. This assumption can be removed by trying $T = 1, 2, \ldots$ sequentially (increasing it forever), since there is no harm in assuming a value of T that is too big (except, maybe, for an increase in computation time).

In the special case of *fixed-sum repeated games* (a.k.a *adaptive competitive decision process*) the R-max algorithm can be simplified considerably (note that here $T = 1$) and is written as follows:

- Initialize matrix entries to $(R_{max}, 0)$
- Mark all entries "unknown"
- Mark entry "known" after played once
- Play optimal strategy according to current model

Note that in this specific setting, two players playing R-max are at an equilibrium: R-max guarantees each of them a value that is ϵ-close to his safety level, and since it is a zero-sum game, neither can do better as long as the other gets his safety level.

> **(DEK)** *Formally, strategies* $(s_1, \ldots, s_n)$ *are at a Nash equilibrium in n-players repeated game under the "limit of expected average payoff" criterion if given the other player's strategies, no player can replace his strategy with one that gives him better payoff (according to the above criterion). Note that:*
>
> - *The players' strategies must specify an action for all possible histories (otherwise, it is impossible to evaluate a possible deviation by a player). Usually, equilibrium strategy prescribes that if some other player deviates, the player changes his behavior to a "punishment" strategy that minimizes the payoff of the deviator (even if it means gaining less for himself in this "what-if"*

[1] In the lecture, as well as in the paper, Prof. Tennenholtz used the expression "probably can"; I am unsure as to what it actually means - **DEK**

scenario). Note that this apparently sub-optimal behavior is not irrational, because it does not affect the player's payoff in the equilibrium profile (the equilibrium payoff is computed based on equilibrium strategies, not deviations).

- *Because the game is infinite and we assume that the players use the "limit of expected average payoff at the infinite horizon" criterion, a player does not increase his value in the game if he deviates only once (at a specific stage of the repeated game), even if the other players ignore this transgression. This is so because the contribution of this deviation to his average payoff converges to zero as the number of games goes to infinity. A corollary of this is that given a profile of strategies such that the payoff of each player (under the "limit of average expected payoff" criterion) is no less than his safety level, one could construct an alternative profile that allocates the same payoffs and is at a Nash equilibrium, by instructing players to play punishment strategy against any deviator (which would ensure that in the limit, a player will get exactly his safety level if he deviates).*
- *This is one of the well-known "Folk theorems" of game theory, so called because they have no specific author and their origins are lost in the mists of time.*

The above definition applies to repeated games with complete information. The learning models that we discuss here deal with incomplete information – situation where the agents do not know beforehand which game they are playing (that is selected out of some predefined set of games), and the notion of equilibrium must be adjusted accordingly. Specifically, we will use the notion of non-Bayesian equilibrium*, which requires that each agents policy is the best response against the other agents' policies for every possible payoff matrix.*

R-max does not define an equilibrium in the case of repeated non-fixed-sum game. However, R-max can be used as a component in constructing equilibrium strategies for general two-player repeated games, as follows:

- Decide on an ordering of the joint actions, and ask each agent to select its actions according to this order, until the game matrix is fully known. When the game matrix is known the agent will be asked to apply its equilibrium strategy (if the game has more than one equilibrium, he should choose one according to some predefined ordering, which is the same for both agents, in order to make sure that both agents play the same equilibrium).
- If an agent has deviated in step 1, then the other agent will move into

a punishing mode and use R-max where his payoffs are taken as complementary to the deviators payoffs.

Note that in the "punishing" mode the player treats the game as a zero-sum game played against the deviator. In this case, R-max guarantees him (payoff arbitrary close to) the safety-level in the zero-sum game, and therefore the deviator will get no more than his safety level. Since in the original equilibrium (that the agents would have played had there been no deviation) each of the agents gets at least his safety level, there is no profit to deviation. Thus, R-max is used here to solve the problem of effectively punishing an agent that deviates before the game is fully known. Also, it should be noted that once the game is known and happens to have multiple equilibria, the players must play the same equilibrium without being able to communicate - we must define (as part of the equilibrium strategies) some lexicographic ordering on Nash equilibria and have the players choose the first one under this ordering in the game. One can also use similar ideas to define an equilibrium in a Stochastic Game.

(DEK) *As an example for R-max not being at an equilibrium, consider the following game:*

R,C	1	2
1	4,1	0,0
2	0,0	1,4

First of all, this game has three Nash equilibria: one (pure) that is excellent for column player and relatively bad for row player, one (pure) that is excellent for row player and relatively bad for column player, and one (mixed) in which both player get the same low payoff ($\frac{4}{5}$) (which, as we have already seen in this course, is also their safety level).
Question: *Suppose that the players play this game repeatedly, they don't know the game beforehand and both of them use the R-max strategy. Are they at an equilibrium?*
Answer: *No. Recall that the players know a priori that in this game $R_{max} = 4$. Now, suppose that in the first turn, the players' actions are $(1, 1)$ and they learn that the associated payoff is $(4, 1)$. Now, the row player observes that in this profile of actions, he actually gets R_{max}, so he is still indifferent between all actions; suppose he sticks with 1. The column player will switch to 2, however, because now it appears to be a (weakly) dominant strategy; the players learn that the payoff there is $(0, 0)$. And now comes the deviation: given the above history, R-max strategy says that the row player should play 2; however, given that the column player uses R-max (and therefore will choose 1 as long as he learns nothing new), the row player is better off playing 1 forever. This way, he will gain 4, whereas R-max would have collected $\frac{4}{5}$ in the limit, because it uses a safety-level*

strategy. Note that, somewhat ironically, this deviation also benefits player 2. The above deviation is possible only in the sub-set of histories in which players choose joint action that gives $(4, 1)$ *in the first turn; however, since we require the algorithms to be optimal for all games (chosen from some set* S*) in order to be at an equilibrium, one can easily verify that if* S *includes all games derived from the above game by row/column permutations, they cannot avoid doing so in all games).*

2.2 Discussion: Exploration vs Exploitation in other adversarial settings

This section is based on two papers:
"Dynamic Non-Bayesian[2] Decision Making" by Dov Monderer and Moshe Tennenholtz, and
"Dynamic Non-Bayesian Decision Making in Multi-Agent Systems" by Dov Monderer and Moshe Tennenholtz.

Definition 1 (Competitive Ratio)**.** Given a two-player game G, we can interpret it as a game between "agent" (player 1 in the original game) and "nature" (player 2 in the original game). The actions of "nature" represent possible states of the environment, and we assume nothing about the way those states are chosen (in particular, player 2 disregards his payoffs, so we can ignore them too). Let S be the set of possible game states, A be a set of possible actions by the agent and $u(a, s) > 0$ be the payoff of action $a \in A$ in state $s \in S$, then:

$$R(a, s) = \frac{\max_{a' \in A} u(a', s)}{u(a, s)}$$

is the *competitive ratio* of action a in state s.

Assuming the agent knows the game, but has no knowledge of the current game state (not even probabilistic), picking an action $a \in arg \min_{a' \in A} \max_{s \in S} R(a', s)$ guarantees him a fixed loss ratio relative to the best strategy (which selects the actions based on the current game state). This action is called *minimal regret action.*

> **(DEK)** *Technically speaking,* any *action has a fixed loss ratio relative to the best strategy, simply because the ratio between payoffs in a finite game is bounded. What the minimal regret action guarantees is the best worst-case bound on the loss ratio among all possible actions (the "worst-case" applies to the selection of states by the nature).*

Since the above setting does not constitute a game in the usual sense (there is only one strategic agent), we will use the term "decision problem" to describe

[2] The term "non-bayesian" refers to the fact that we do not assume any prior distribution on the environment

it. We will use the term “Perfect monitoring” to describe a setting in which an agent can observe both the state and the payoff after each stage, and “Imperfect monitoring” to describe a setting in which only the payoff is known.

Claim 1 (Perfect monitoring)**.** *Given a game G, $\epsilon > 0, 1 > \delta > 0$, there exists an integer $T = poly\left(|G|, \frac{1}{\epsilon}, \frac{1}{\delta}\right)$ and a policy P, such that after T iterations of the corresponding repeated game G_∞, the policy P yields average competitive ratio that is ϵ-close to the optimal competitive ratio of G, with probability of failure of at most δ.*

> **(DEK)** *Note that in the paper, the requirement from the policy is written as: that after some time T, the policy plays the minimal regret action almost all (i.e. $1-\epsilon$-fraction of) the time. It is easy to see that the requirements are equivalent – if the minimal regret action is played most of the time, since payoffs are bounded it implies an average regret that is close to optimal, and if the average regret is ϵ-close to optimal for small enough ϵ, it implies that the minimal regret action is played very often (because the game matrix is finite).*

The policy is rather intuitive: we will keep two matrices:

- an approximate game matrix, initialized with 'unknown' and updated with the observed payoff after each stage.
- an approximate competitive ratio matrix, which contains 1 in entry (a, s) if the payoff of action a in state s is 'unknown' and the competitive ratio of action a in state s relative to the known payoffs for that state otherwise.

In each stage, we will do one of the following:

- With small probability, we will play a random action. This ensures that we will try all actions in all states, except perhaps some states that the nature rarely visits and therefore they have small effect on the average competitive ratio.
- Usually, we will play the minimal regret action according to our knowledge so far.

After each stage, we will update our matrices according to the observed state and payoff.

Claim 2 (Imperfect monitoring, impossibility)**.** *Given an imperfect monitoring setup, where the agent observes only the payoff obtained at any stage and not the nature's state, no such policy exists.*

As an example, consider a setting in which one of the following payoff matrices may be used:

	s_1	s_2	s_3
a_1	2a	2b	2c
a_2	a	b	c

	s_1	s_2	s_3
a_1	2a	2b	2c
a_2	b	c	a

where $a > 4b > 16c > 0$. Note that in the first matrix, a_1 is the unique minimal regret action and in the second – a_2 is such. Now suppose nature chooses between states s_1, s_2, s_3 uniformly at random independently at each stage. Obviously, given this distribution the agent cannot distinguish between the two decision problems, and therefore has no chance of learning to play the minimal regret action.

Claim 3 (Imperfect monitoring, possibility: "Two arms")**.** *Given that the agent can perform two actions simultaneously in the imperfect monitoring case (where the agent observes only the payoff obtained at any stage and not the nature's state), such a policy exists.*

The assumption in the "Two arms" model is that the "arms" do not affect each other (that is, their payoff is determined by the same matrix in the same state of nature, and the matrix is the same for all stages). Let s be the unknown world state and a_1, a_2 be the actions (w.l.o.g $u(a_1, s) > u(s_2, s)$), then

$$R(a_2, s) \geq \frac{u(a_1, s)}{u(a_2, s)}$$

(that is, using two actions simultaneously, we can bound the competitive ratio of our actions). Therefore, we can keep for each action the worst competitive ratio observed for it so far (as defined above) and do one of the following at each stage:

- With small probability, let the left arm do one of the optimal actions according to our observations so far (selected uniformly at random) and let the right arm do a random action. This ensures that we will try out all actions in all states (except, maybe, some states that the nature rarely visits and therefore they have small effect on the average competitive ratio). If worse competitive ratio that was known so far is observed for one of the actions, update our knowledge accordingly.
- Usually, let both arms play the same optimal action (selected uniformly at random).

If the "small probability" from above is chosen well, we will get a near-optimal strategy under the average competitive ratio criterion.

3 Mechanism Design

3.1 Games with incomplete information

There are several ways to model uncertainty in games. The most classical way to model uncertainty is by referring to partial information about agents' utility functions. The utility function of an agent is taken to depend on a random parameter called the agents type, which is typically known to the agent but it is unknown to the other agents. An agents type is, for example, his willingness to pay for a particular good. The typical assumption is that the distribution on the agents types is known, each agent knows his/her type but in general he/she does not know the type of other agents.

The equilibrium concept has been generalized to games with incomplete information (Harsanyi) as:

Definition 2. A joint strategy of the agents is in equilibrium, if each agent applies its expected best response against the strategies of the other agents, given the distribution on agents types.

An alternative solution concept, for example, is to consider a strategy profile in which agents play weakly dominant strategies. An example for such solution is an auction in which the highest bid wins and pays the second-highest bid (this is a so-called second-price auction, as opposed to a first-price auction, in which the winner pays his bid). It is easy to see that in this setting, bidding one's true valuation is a weakly dominant strategy.

> **(DEK)** *Note that a profile that consists of weakly dominant strategies is, in particular, at equilibrium, but not vice versa.*

Proof. To show that it is a weakly dominant strategy to bid truthfully, one needs to show that truth-telling is a best response to any profile of opponents' actions for any type of the agent. Let x be the agent's type (valuation of the good) and y be the highest bid among all other agents. Now, there are two cases:

- $x > y$ - the agent wins if he bids truthfully (and gets $x - y$). In this case, any bid $x' > y$ wins the auction and gives the same (positive) payoff $x - y$; any bid $x' < y$ loses the auction and gives payoff 0. Therefore, the agent cannot gain more by bidding untruthfully.

- $x \leq y$ - the agent gets 0 if he bids truthfully (either by losing the auction, or, in the case $x = y$, by winning the tie-breaking procedure and paying exactly his valuation). In this case, any bid $x' > y$ wins the auction but loses money, since $y - x < 0$; any bid $x' < y$ loses the auction and gives payoff 0. Therefore, the agent cannot gain more by bidding untruthfully.

□

An immediate and important question is: What is more effective to the seller (sells the item at higher price) - first-price or second-price auction?
The answer is not trivial (and we will discuss it comprehensively later), because while in second-price we can expect agents to bid truthfully, in first-price auctions agents bid less than their valuation (obviously - they want to leave some margin for profit).

> **(DEK)** *Practical experiments show that actual humans bid truthfully in second-price auction only* after *the above reasoning was explained to them. Therefore, it seems that the game theory would show better practical results if its basics were taught to everybody in elemental school, so that we could be sure the agents are behaving rationally.*

An important general conclusion is that a small change in the protocol may drastically change the agents' behavior. This is the main difference between *Protocol Design* and *Mechanism Design* – both try to optimize some criterion for "good behavior" , but the value of a mechanism is defined only given an environment and a solution concept. A mechanism together with an environment defines a game; this game must be analyzed using some solution concept in order to predict the agents' behavior, which then defines the value of the mechanism.

3.2 Introduction to Auctions

Auctions are the most widely-studied economic mechanism. Generally speaking, an auction mechanism is a mechanism for resource allocation with self-motivated participants.

> **(DEK)** *Some sceptics say that auction theory is the* only *successful application of Mechanism Design.*

The classical assumptions of auction theory are:

- Independent valuations for objects (IPV – independent private values) – the agents' types are independent random variables. This means, among other things, that an agent cannot use his valuation to deduce knowledge about other players' valuations. This is not necessarily a reasonable assumption (usually, there is some consensus regarding the item's value and therefore the valuations are positively correlated), but the analysis is too often infeasible without it.

- Free disposal – if an agents is allocated with more items that what he bid for, his utility does not decrease as a result.

- No externalities – agent's utility depends only on the items he is allocated with and the price he pays (and not, for example, on whether some other agent wins or loses the auction).

- Constant risk attitude – the attitude of an agent towards risk is usually modeled through his utility function $u : \Re \rightarrow \Re$, that translates his monetary gain to utility: for example, an agent that is risk seeking is modeled by a convex[3] utility function, because then $E(u(x)) \geq u(E(x))$, which means that an agent strictly prefers a lottery to getting the expected value of the lottery with probability 1. If the agent is risk-neutral, his utility is a linear function of his monetary gain and maximizing the expected utility is the same as maximizing expected monetary gain, which simplifies the analysis. In any case, it is assumed that the agent attitude towards risk is the same regardless of the size of the prizes involved (in particular, that if he prefers a constant prize of 1$ to a fair lottery between 0$ and 2$, then the prefers a constant prize of 1000$ to a fair lottery between 0$ and 2000$, and a constant prize of 1001$ to a fair lottery between 1000$ and 1002$).

- Usually, either risk-neutral or risk-averse behavior is assumed, since economical and psychological theories postulate that this is the model that people use in economical decision-making (although there is some evidence of risk-seeking behavior in Internet environment, in particular in Internet auctions)

Finally, it should be stated that the basic model of single-item auction can be extended in two ways:

- *Combinatorial (multi-dimensional) auctions* lead to hard computational problems, but are more expressive. In those auctions, the auctioneer sells a multiset[4] of items and participants place bids for one or more multisets of items. This is a problematic setting, because regardless of the game theory aspects, the problem of efficient allocation given a profile of bids (i.e. one that maximizes the seller's revenue) cannot be solved efficiently.

  > **(DEK)** *The problem, in fact, is NP-hard – can be proved by a reduction from the knapsack problem, which is the problem of finding a subset of a set of items, each having two parameters – size and cost, such that the total size is no bigger than some limit and the total cost is maximal.*

- *Multi-round auctions* lead to complex equilibrium analysis and multiple equilibria. This is a setting in which the same item may be offered by the same seller to the same buyers in several consecutive auctions, usually because the reserved price wasn't met and there was no sale (the reserve price is the minimal price the seller is willing to accept for the item).

[3] Having positive second derivative

[4] Set that may contain the same item several times but, like a regular set, implies no ordering on them.

3.3 Known Auction Protocols

3.3.1 Single-unit English auction

This is the protocol in which participants call out ascending prices, and the agent with the highest bid gets the item and pays his bid. The auction ends when some time limit expires – either fixed, or (Amazon's online version) extended if a bid arrives in the last X minutes before the deadline. The latter version, although natural, causes problems in so-called "Collector Item" setting – an auction that sells items (like rare post stamps) which are hard to evaluate based on the information available before the auction starts. In those auctions, typically there are two types of participants - "experts" (a.k.a "collectors") which have good evaluation skills but low budget, and "dealers", which have low evaluation skills but big budget. The problem is that the "dealers" can compensate for their lack of knowledge by approximating the item's value based on the bids of the other participants (and once they feel they have accurate enough information, they can easily outbid the "experts"). This tactics doesn't work in the fixed deadline setting, because there the "experts" can postpone declaring their bids till the last possible moment and hope that the "dealers" won't have enough time to react (as opposed to the "time extension on late bids" modification, where "dealers" always have time to react). As a consequence, the "experts" simply won't participate in this type of auctions, and without them no sale occurs.

> **DEK** *Actually, the way I see it, without the "experts" the seller's revenue just drops considerably (because then the "dealers" cannot evaluate items effectively, thus they suffer high risk of buying bogus items and then they compensate for it by bidding low - "dealers" are usually risk averse).*

3.3.2 Multi-unit English auction

This is the setting in which several equivalent items must be sold to several participants, one or more items to each. In this setting it is possible that several participants, each with different bid, win the auction and are allocated with goods, and then the question becomes: according to whose bid is the price set? There are several options:

- *Lowest accepted bid* (uniform pricing to all winners – the minimum of their bids). This pricing scheme is sometimes called "Dutch", despite the fact that it has nothing to do with Dutch auctions, which will be discussed later.
- *Highest rejected bid* (the maximum of the bids of the participants that didn't win).
- *Pay-your-bid* (discriminatory pricing). This turned out to be a problematic option in the Internet setting due to psychological reasons – on the

Internet, people can easily compare prices and discover that in the same auction, for the same item, one paid much more than the other, and then the reputation of the seller suffers – the auction is perceived as “unfair”.

3.3.3 Japanese auction

In this setting, the auctioneer calls out ascending prices and bidders are initially “in”, and drop out (irrevocably) at certain prices. Last guy standing gets the item at that price. This auction is equivalent to an English auction, up to the size of a single price change.

3.3.4 Dutch auction

Auctioneer starts with some high price and calls out descending prices.First bidder to jump in gets the good at that price. This protocol can be adjusted for multiple units as follows: bidders shout out a quantity rather than “mine”, and the appropriate number of items is allocated to that bidder at the current price. The amount of the items for sell is then decreased by the number called, and the price can either continue to drop, or reset to any value.

3.3.5 Sealed-bid auction: First-price, Second-price . . . k-price

Each bidder submits an closed envelope with a bid. The envelopes are opened and the highest bid wins. The price paid can be any of: first, second, . . . , k’th price (i.e. k’th bid in the decreasing order of bids).

3.3.6 Reverse (procurement) auctions

This is the subset of auctions in which service suppliers compete for a contract by offering low prices. All the protocols from above can be adopted to this setting (which is essentially equivalent to a “usual” auction); the only reason why it is worth mentioning is because this is by far the most common and the most significant (in terms of the revenue collected) type of auctions in modern economics.

3.3.7 Some known relations between auction protocols

- Dutch $\equiv$ 1st-price sealed bid. Intuition: The participants in Dutch auction and in 1st-price auction solve the same equation in order to decide which bid to submit (or at which price to call ’I buy!’) - which is the optimal price so that my margin of profit (the difference between my valuation and that price) times the probability that none of the other participants will call a higher price is maximal.
- English $\equiv$ Japanese. Intuition: It does not matter whether the participants or the auctioneer raise the bid.

- English ≡ 2nd-price sealed bid under IPV. Intuition: a participant wins the English auction when the second highest bidder drops out, and the price he pays is the last bid of the second highest bidder (plus the minimal bid increase, which is negligible) - the same as in 2nd-price sealed bid auction.

It is worth noting that it is hard to build a mathematical model for protocols that involve repeated interaction between agents at arbitrary points in time, because there is no simple and sufficient mathematical model for time.

The Amazon online auction (which is technically an English first-price auction) allows agents to delegate the bidding to an automaton (called proxy), which is initialized with the maximal price the agent is willing to pay for the product and automatically raises to the minimal price that is needed to beat the current bids of other agents, as long as he does not pass the limit set by the agent. It can be easily verified that using a proxy initialized with the true value is a weakly dominant strategy and assuming all agents use proxies, the auction essentially becomes a second-price auction.

3.4 Expected revenue

In general, the expected revenue of the seller depends drastically on the buyers' attitude towards risk. This is not surprising, since auction is essentially a (possibly unfair) lottery among all participants. Particular known results include:

- When the agents are risk-neutral, all k-price auctions are revenue equivalent (Myerson).

- When agents are strictly risk-averse, then first-price and Dutch are preferable to second and English (Maskin and Riley, Riley and Samuelson). The intuition is that agents although agents wont't bid higher that their real valuation, they will be afraid not to get the item and therefore will bid very closely to it, and therefore their bids will be close to those in second-price auction (with same valuations) – but given that the bids are the same, obviously first-price auction collects more revenue.

- When the agents are risk-seeking, $(k+1)$-price is preferable to k-price. The intuition is that in k-price auction ($k > 2$), agents overbid (i.e. bid more than their true valuation), and they overbid by a bigger margin as k increases. On top of that, risk-seeking agents will overbid even more in the hope of getting the pig prize (i.e. winning the auction) and therefore will pay more, and the margin by which they overbid is multiplied by the effect of increasing k.

Claim 4 (The Revenue Equivalence Theorem)**.** *In all auctions for k units with the following properties:*

- *Buyers are risk neutral,*

- *IPV, with values independently and identically distributed over* $[a, b]$ *(technicality: distribution must be atomless, i.e. there is no* $x \in [a, b]$ *so that* $Pr(\{x\}) > 0$*),*
- *Each bidder demands at most 1 unit,*
- *Auction allocates the units to the bidders with the k highest valuations,*
- *The bidder with the lowest valuation has a surplus of 0 (that is, his expected utility is 0),*

a buyer with a given valuation will make the same expected payment, and therefore all such auctions have the same expected revenue.

All of the above analysis deals with a fixed number of participants. Recently, probably because of the introduction of Internet auctions, the following question was asked: "How does the seller's revenue behave when the number of participants grows?". As of now, we know the following:

- An upper bound on the expected seller's revenue is the expected highest valuation (note that this is so in spite of the fact that agents may overbid).
- When the number of participants is large – English auctions approach the upper bound.
- In addition, English auction with k+1 participants is better for the seller than *any* auction with k participants, so a practical conclusion is that Internet auctioneers are better off investing in marketing than in protocol design!