Maximizing Seller's Profit for Electronic Commerce

Kostas Kalpakis

(joint work with B. Belegradek & Ye. Yesha)

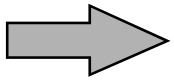
e-mail: kalpakis@csee.umbc.edu

Outline

- Motivation
- Previous work
- Model
- Loss functions
- Admissible pricing algorithms
- Algorithm to minimize total loss
- Minimizing the maximum loss
- Conclusions

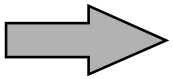
Motivation

Computers are used to store and disseminate product information *(catalogs, virtual shops, advertisement)*



Product information can be changed quickly and/or frequently without human interaction. The changes are inexpensive and can be transmitted to customers quickly.

Internet is used to facilitate direct business-to-customer electronic trade *(on-line ordering, payment and delivery of digital products)*

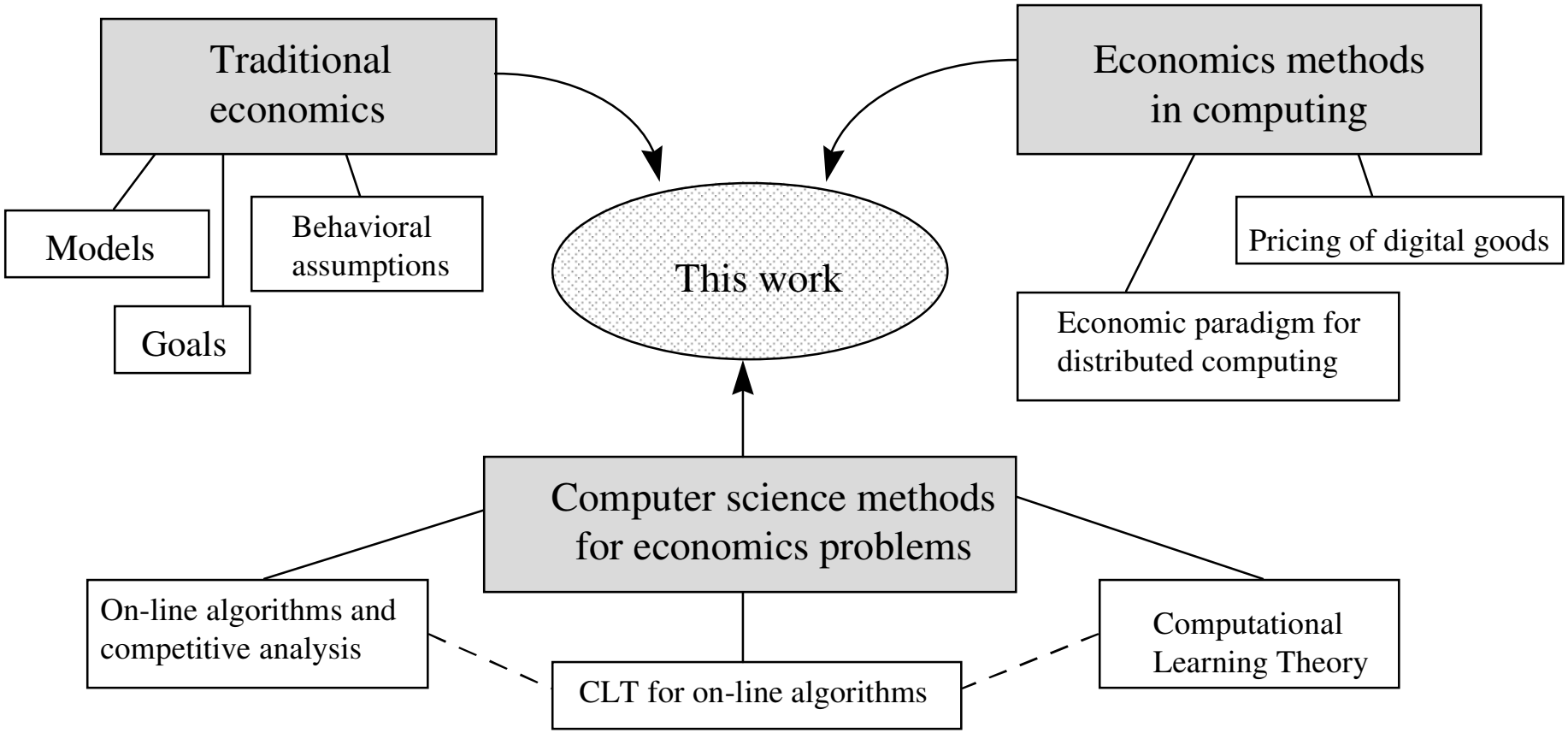


Implicit information about buyers can be collected and analyzed automatically.

Motivation

- What strategy should a seller follow to maximize his or her profits?
- What actions does the sales protocol allow?
- How much information is made available to buyers and sellers?
- Are buyers and sellers assumed capable of strategic behavior?

Previous work



General Model

- Sellers are rational; buyers are rational and risk-neutral.
- Sellers use the sales history (prices and quantities of product sold), as the sole input of pricing algorithms.
- Buyers do not know the sales history and they do not behave strategically.
- *Buyer's valuation:* the difference between the maximum price the buyer is willing to pay per unit of a product and the cost of such a unit to the seller.
- Discrete time periods.

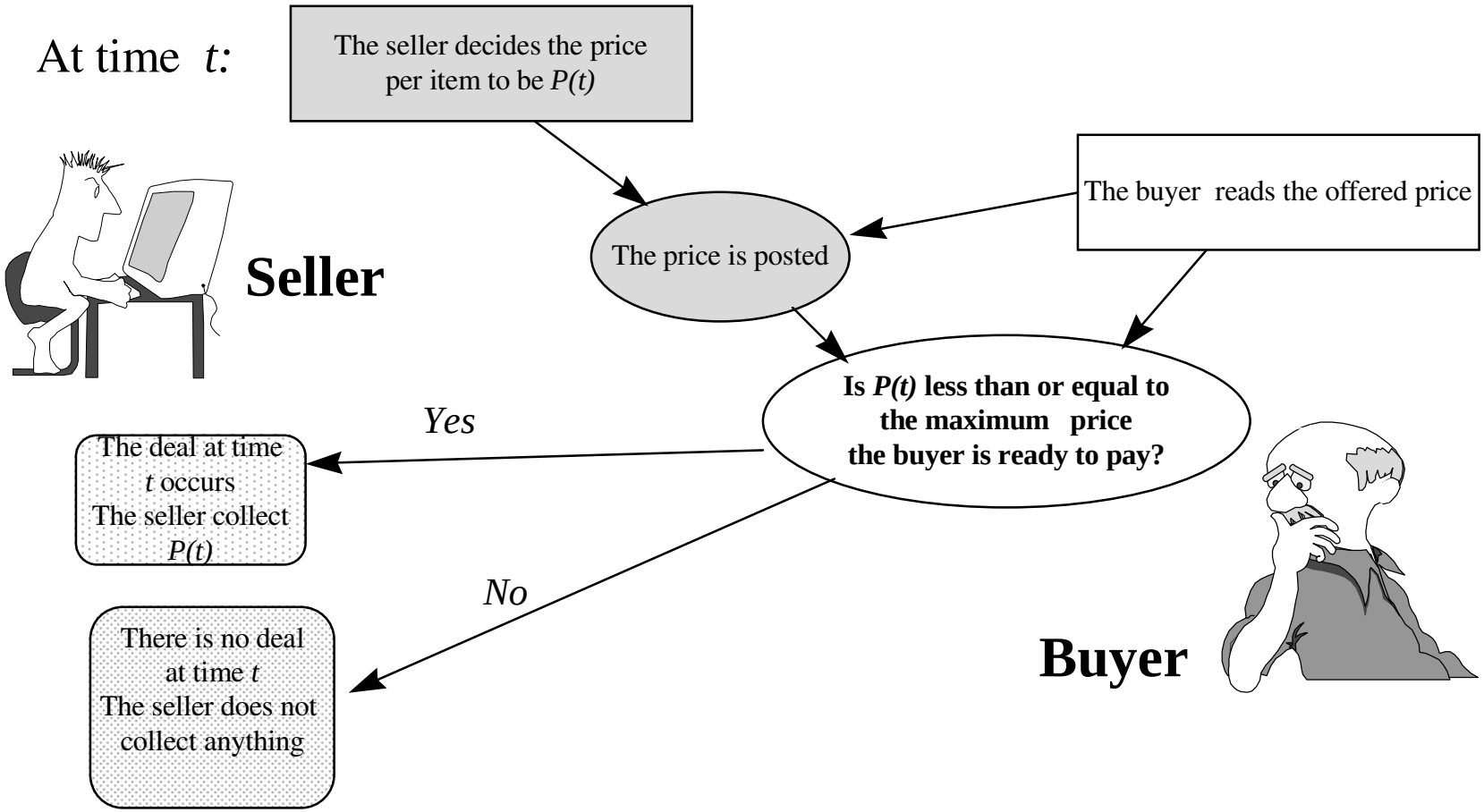
General Model

- At time period t the following actions take place:
  1. The seller sets the current price to $p(t)$.
  2. The seller posts current price and the number of units for sale.
  3. Buyers access the posted information and decide whether they will buy.
  4. The first buyer whose valuation exceeds the posted price buys.
  5. The seller adds a record of sales, if successful, into sales history.

Simple Model

- Here, we further make the following simplification assumptions:
  - There is a single buyer and a single seller in the market.
  - The buyer's valuation is a constant in a given range, known to the seller.
  - At each time period, there is 1 unit of the product for sale.

Sales Protocol



Loss Functions

Let $\boldsymbol{X}$ be the buyer's valuation.

Let $P(t)$ be the price set by a pricing algorithm $\mathcal{A}$ at time t.

The loss of a pricing algorithm $\mathcal{A}$ at time t, a measure of lost opportunity to make a profit, is equal to the difference between the maximum profit the seller can obtain (namely **X**) and the profit actually obtained at time t.

$$loss_{\mathcal{A}}(X,t) = \begin{cases} X, & \text{if } X < P(t) \\ X - P(t), & \text{otherwise} \end{cases}$$

The loss of $\mathcal{A}$ for the buyer's valuation X is:

$$loss_{\mathcal{A}}(X) = \sum_{t=0}^{\infty} loss_{\mathcal{A}}(X,t)$$

Loss Functions

- The ***total loss*** of a pricing algorithm $\mathcal{A}$ is

$$total_loss_{\mathcal{A}} = \sum_{X=0}^{N} loss_{\mathcal{A}}(X) \qquad \text{(average case loss)}$$

- The ***maximum loss*** of a pricing algorithm $\mathcal{A}$ is

$$max_loss_{\mathcal{A}} = \max_{0 \leqslant X \leqslant N} loss_{\mathcal{A}}(X) \qquad \text{(worst case loss)}$$

where the [0,N] is the range that contains the buyer's valuation.

Admissible Pricing Algorithms

•A pricing algorithm can be optimal only if it is able to find any buyer's valuation in a finite amount of time, thus it must provide finite losses on all the possible inputs.

•We call a pricing algorithm ***admissible***, if it makes progress towards finding the maximal buyer's price at every time step

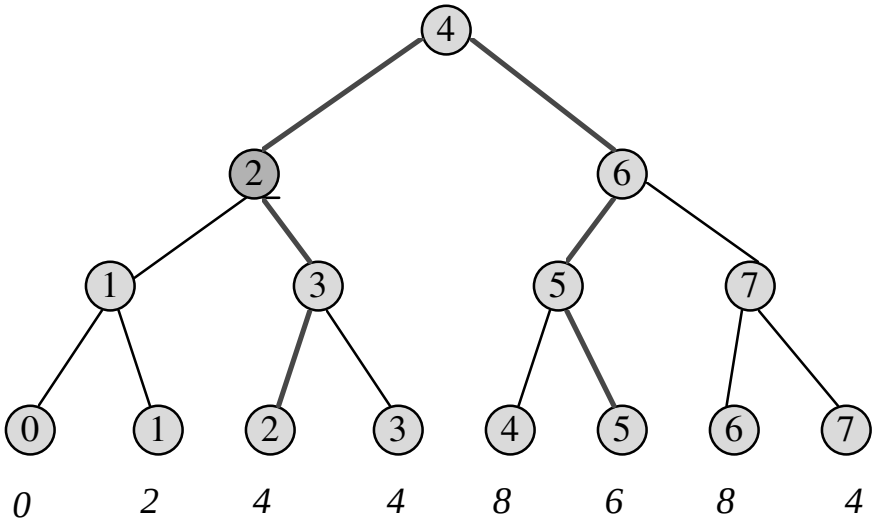
•We show that:

> *An admissible pricing algorithm $\mathcal{A}$ for the range [0…N] can be represented as a binary "search" tree with (N+1) leaves and N internal nodes, each of degree 2.*
>
> *There exist optimal admissible pricing algorithms for both the total loss and maximum loss functions.*

•Thus, the search for optimal admissible algorithms is reduced to a search in the space of such trees.

Examples of Admissible Pricing Strategies

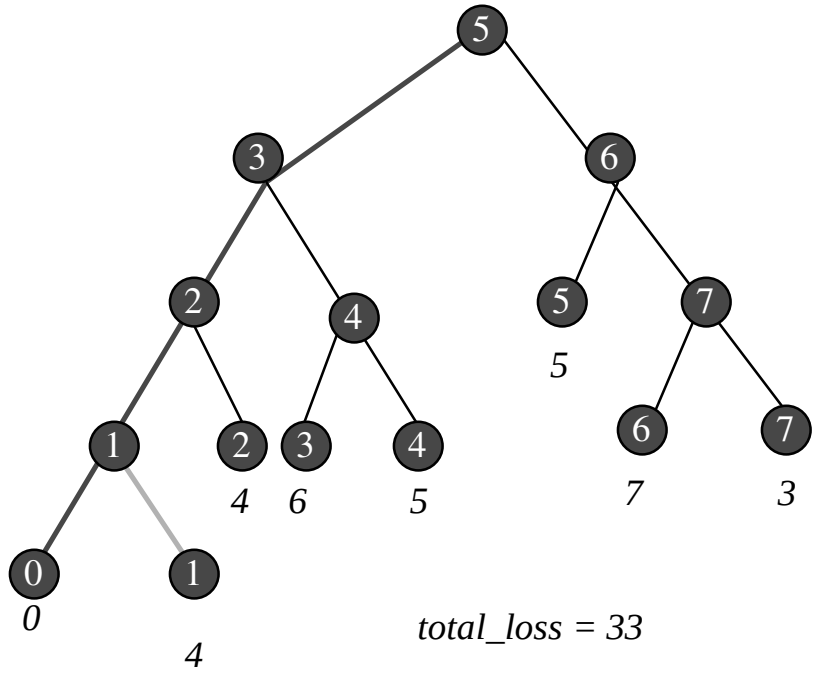
Balanced binary search tree on *[0...7]*



total_loss = 36

max_loss = 8

Tree minimizing the total loss on *[0…7]*

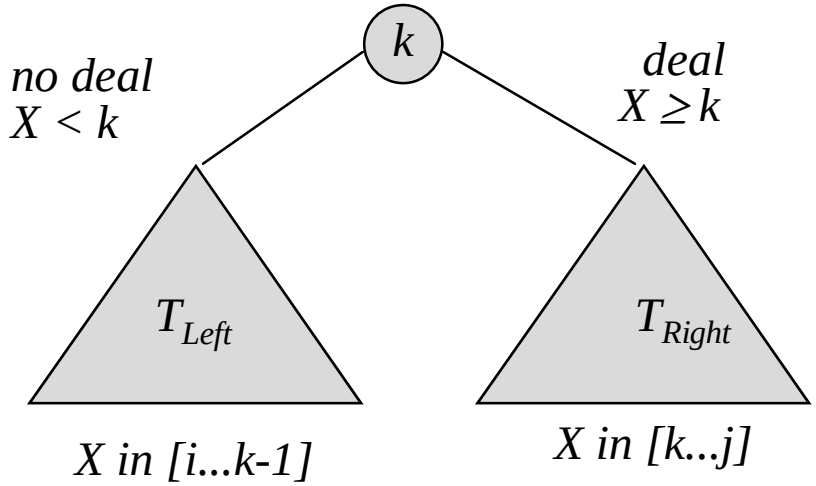


total_loss = 33

max_loss = 7

Constructing an admissible pricing algorithm minimizing the total loss

- Approach: dynamic programming.
- Suppose that the buyer's valuation X is in *[i, j] and that the 1st price selected is k:*



Constructing an admissible pricing algorithm minimizing the total loss

- When *X is* in *[i...k-1] then the total loss for X in T* is equal to:

  X + total loss for X in T_{Left}

- When X is in *[k...j] then the total loss for X* in *T* is equal to:

  X - k + total loss for X in T_{Right}

- *Hence, for the range [i,j] the total loss is:*

  $\text{total_loss}(T) = \text{total_loss}(T_{Left}) + \text{total_loss}(T_{Right}) +$

  $+(j+i)(j-i+1)/2 - k(j-i+1)$

Constructing an admissible pricing algorithm minimizing the total loss

- By minimizing over all values of k in the range [i,j], we find an admissible algorithm that minimizes the total loss for the range [i,j].

- The time to find such a pricing algorithm is O(n^3) where n=(j-i).

- This generalizes to minimize expected total loss when the buyer's valuation has a known discrete probability distribution on [0,N].

Constructing an admissible pricing algorithm minimizing the total loss

- We also show that

$$\Omega(N^2) \leqslant \textit{min. total_loss for } [0,N] \leqslant O(N^2 \lg N).$$

while for a complete binary tree the total loss is $\sim N^2 \lg N$

Minimizing the Maximum Loss

•We show that

$$N \leqslant \text{minimum maximum loss for } [0, N] \leqslant (1+o(1))\, N \lg N.$$

•We also found that there is no strategy that minimizes both the total loss and maximum loss functions simultaneously.

•The smallest N for which this happens is N=8, where among all the trees with the minimum maximum loss of 8, the minimum total loss is 47, while the minimum total loss is 46.

• *The problem of efficiently finding an admissible algorithm that minimizes the maximum loss on a given input is left open.*

Conclusions

- Currently Electronic Commerce applications deal mostly with routine aspects of commercial activities.
- To be successful in “human” areas of commerce, such as content providing and decision making, Electronic Commerce needs
  - *accurate models, both conceptual and formal*
  - *sophisticated algorithms*
- Our work makes a step in both directions
  - *we presented a simple model for on-line catalog sales*
  - *we constructed optimal (with respect to cumulative profits) pricing algorithms.*

Some Selected References

B.Awerbuch and Y.Azar. Blindly-competitive algorithms: Pricing and bidding a case study. *Extended abstract,* April 1995.

A.Blum On-line algorithms in machine learning. *Technical report, CMU,* 1996.

D.F.Ferguson, C.Nikolaou, J.Sairamesh and Y.Yamini. Economic models for allocating resources in computer systems. In *Market Based Control of Distributed Systems.* World Scientific Press, 1996.

M.Stonebraker, P.M.Aoki, A.Pfeffer, A.Sah, C.Staelin and *A.Yu.Mariposa:A Wide-Area Distributed Database System.* VLDB Journal, 5 (1), pp. 48-63, 1996.

M.P.Wellman. Market oriented programming: Some early lessons. In *Market Based Control of Distributed Systems.* World Scientific Press, 1996

A.Whinston, D.O.Stahl and S.-Y. Choi. *The economics of electronic commerce.* Macmillan Technical Publishing, 1997.

Some Selected References

- L. M. Ausebel and R.J. Deneckere, *Reputation in bargaining and durable goods monopoly*, Econometrica, 1989.
- R. Benabou, *Optimal price dynamics and speculations with a storable good*, Econometrica, 1989.
- J. Sobel, *Durable goods monopoly with entry of new consumers*, Econometrica, 1991.