Aalto University School of Science and Technology

Systems Analysis Laboratory Research Reports E24, November 2010

CONTINUOUS LEARNING METHODS IN TWO-BUYER PRICING PROBLEM

Kimmo Berg Harri Ehtamo

Aalto University
School of Science
and Technology

Distribution:

Systems Analysis Laboratory

Aalto University

P.O. Box 11100

00076 AALTO, FINLAND

Tel. +358-9-470 23056

Fax +358-9-470 23096

systems.analysis@tkk.fi

This report is available at

www.e-reports.sal.tkk.fi/pdf/E24.pdf

Series E - Electronic Reports

www.e-reports.sal.tkk.fi

ISBN 978-952-60-3476-8

ISSN 1456-5218

Title: Continuous Learning Methods in Two-Buyer Pricing Problem

Authors: Kimmo Berg and Harri Ehtamo
Systems Analysis Laboratory
Aalto University School of Science and Technology
P.O. Box 11100, 00076 Aalto, Finland
kimmo.berg@tkk.fi
www.sal.tkk.fi/en/personnel/kimmo.berg/

Date: November, 2010

Status: Systems Analysis Laboratory Research Reports E24 November 2010

Abstract: This paper presents continuous learning methods in a monopoly pricing problem where the firm has uncertainty about the buyers' preferences. The firm designs a menu of quality-price bundles and adjusts them using only local information about the buyers' preferences. The learning methods define different paths, and we compare how much profit the firm makes on these paths, how long it takes to learn the optimal tariff, and how the buyers' utilities change during the learning period. We also present a way to compute the optimal path in terms of discounted profit with dynamic programming and complete information. Numerical examples show that the optimal path may involve jumps where the buyer types switch from one bundle to another, and this is a property which is difficult to include in the learning methods. The learning methods have, however, the benefit that they can be generalized to pricing problems with many buyers types and qualities.

Keywords: pricing, learning, limited information, buyer-seller game,
mechanism design

Aalto University School of Science and Technology
Systems Analysis Laboratory Research Reports

Continuous learning methods in two-buyer pricing problem

Kimmo Berg · Harri Ehtamo

November 4, 2010

Abstract This paper presents continuous learning methods in a monopoly pricing problem where the firm has uncertainty about the buyers' preferences. The firm designs a menu of quality-price bundles and adjusts them using only local information about the buyers' preferences. The learning methods define different paths, and we compare how much profit the firm makes on these paths, how long it takes to learn the optimal tariff, and how the buyers' utilities change during the learning period. We also present a way to compute the optimal path in terms of discounted profit with dynamic programming and complete information. Numerical examples show that the optimal path may involve jumps where the buyer types switch from one bundle to another, and this is a property which is difficult to include in the learning methods. The learning methods have, however, the benefit that they can be generalized to pricing problems with many buyers types and qualities.

Keywords pricing · learning · limited information · buyer-seller game · mechanism design

1 Introduction

In nonlinear pricing problem a monopolistic firm designs a menu of products to serve a population of buyers with different valuations. The model can basically be applied in any product pricing where the firm may sell multiple units or versions with different quality attributes (Wilson 1993). The problem is also an example of mechanism design (Rochet and Stole 2003), which has been studied extensively in economics, game theory and recently computer science

K. Berg · H. Ehtamo
Aalto University School of Science and Technology, Systems Analysis Laboratory,
P.O. Box 11100, 00076 Aalto, Finland

K. Berg
kimmo.berg@tkk.fi, Tel.: +358-9-47023066; fax: +358-9-47023096

(Armstrong 1996; Nisan and Ronen 2001; Conitzer and Sandholm 2002). This paper examines a situation where the firm has uncertainty about the buyers' preferences and learns the optimal tariff by selling the product repeatedly. The firm adjusts the quality-price bundles using only local information about the buyers' preferences.

The pricing problem was originally studied by Mussa and Rosen (1978), Spence (1980) and Maskin and Riley (1984), who developed the conditions and properties that characterize the optimal bundles. The multidimensional pricing problem has been studied by Wilson (1993), Armstrong (1996), Rochet and Chone (1998), Nahata et al (2002) and Basov (2005). See also Armstrong (2006) and Stole (2007) for recent surveys on price discrimination. Many papers on pricing assume so-called single-crossing condition, which restricts the shape of buyers' possible utility functions. There are, however, few papers that examine a model without it, e.g., (Araujo and Moreira 1999; Nahata et al 2004; Berg and Ehtamo 2008). In this paper we assume the single-crossing condition to illustrate the learning idea better and the learning methods can be applied in the more general model as well.

Learning under limited information is currently under active research (Fudenberg and Levine 1999; Sandholm 2007). Learning in the pricing problem has been studied by Braden and Oren (1994) and Brooks et al (2002); see also the related field of dynamic pricing (Elmaghraby and Keskinocak 2003; Garcia et al 2005; Lin 2006). Brooks et al (2002) study the one-dimensional problem and the cost of learning for different kinds of tariffs. They find that the complicated tariffs with more parameters take more time to learn but finally produce better profits. Our pricing model is more general and we study continuous learning methods. One of these methods is the gradient method, which has been studied by Bowling and Veloso (2002) and Hofbauer and Sigmund (2003). See also Raju Chinthalapati et al (2006) and Vengerov (2008) for reinforcement learning models under competition.

This paper is based on the recent development by Ehtamo et al (2010) and Berg and Ehtamo (2009, 2010). The adjustment approach was proposed by Ehtamo et al (2010) in the one-dimensional problem with two buyer types. The model was generalized to have multiple buyer types in Berg and Ehtamo (2009), where it is explained how the bunching and exclusion problem can be solved using only limited information about the buyers' utility functions. Both of these papers use discrete steps in the adjustment, which poses the problem of choosing a good step size. This paper suggests a solution to this problem by using continuous learning paths. This reduces the learning problem to only choosing the adjustment direction. The learning methods are compared in terms of the firm's profit, the buyers' utilities and the learning time. We also compute the optimal path as a reference to measure how good the learning methods are that use only limited information. This makes it possible to estimate the value of information.

The rest of the paper is organized as follows. In Section 2, we give an interpretation of our approach, introduce the pricing problem and the simplifying assumptions. In Section 3, we present the continuous learning methods using

differential equations, the optimal path using the dynamic programming algorithm and the criteria to compare the methods. The numerical simulations are given in Section 4. The one-dimensional example illustrates the learning paths and shows the firm's profits and the buyers' utilities over the learning period. The second example demonstrates that the adjustment idea can also be used in the more general multidimensional problem. Finally, Section 5 is the discussion.

2 Pricing setting

2.1 Interpretation

Let us examine how a firm should price its product when the firm faces unknown stationary demand, which means that the demand is fixed for the learning period. This setting has two interpretations. First, the setting could describe a firm producing a new product or a service. In the second interpretation, the firm may have sold the product for some time but there has been a demand shock. Thus, the firm needs to adjust its price schedule to the new demand. The difference in these two situations is that the firm may have a good approximative solution in the latter case, whereas in the former case the initial price schedule may be far off from the optimal one.

We study how the firm can learn the optimal price schedule by selling the product and observing the sales, and this is known as the online learning scheme. We develop local adjustment methods under different informational assumptions. The local adjustment means that we study continuous learning paths, where the firm decides a direction of change and updates the bundles a little to this direction. The local adjustment has the benefit that we do not have to determine step lengths in the method. The choice of step lengths may be problematic, because it affects how fast the optimal schedule is found. A big step in the right direction may improve the profit considerably, but in the wrong direction may decrease the profit as well. Also, the firm may have reliable demand estimates only around the current price schedule, and thus the local adjustment with small changes is justified.

We examine two informational assumptions: a complete information setting and a setting where the firm only knows the buyers' valuations locally around the current price schedule. The complete information setting gives the optimal price schedule and the optimal learning path, when the firm fully knows the demand. This is the best path the firm can achieve, and we compute the path by discretizing the quality-price space and using dynamic programming. The limited information setting is the more interesting one, where the firm makes the adjustment using only local information. There are many different heuristics to do the adjustment, and we compare some of them against the optimal path. We note that the learning methods are not comparable with the optimal path, since its computation requires complete information.

It is assumed that the buyers are myopic, see Fudenberg and Levine (1999), and Ellison (1997), i.e., they maximize their utility each round and do not try to affect the firm's learning task. One explanation for myopicity is a model, where the buyers are randomly chosen to act from a large population. The population consists of groups of identical buyers, and each buyer behaves myopically since it is unlikely that they are chosen to act in the following rounds. This way the buyers do not have the incentive to manipulate the firm's pricing, and the firm can learn about the buyers' valuations by observing the buyers' purchases.

2.2 Nonlinear pricing model

Let us formulate the multidimensional quality pricing model with discrete buyer types. A monopolistic firm produces a product of quality q to a population of buyers, where q is a k-dimensional vector of the product's qualities. The cost of producing a product with quality q is $c(q)$, and the cost is assumed to be smooth, increasing, and convex. The product line, i.e., the range of possible qualities, is assumed to be the positive quadrant, that is, $q \in \mathbb{R}^k_+$, and the zero vector represents no consumption and the corresponding cost is zero, i.e., $c(0,\dots,0) = 0$.

The population of buyers consists of n distinctive types, which are indexed by $i \in I = \{1,\dots,n\}$. Each buyer $i \in I$ consumes at most one unit of the product and has a separable quasi-linear utility

$$U_i(q,p) = u_i(q) - p, \quad \forall i \in I, \tag{1}$$

where p is the price of the product, and $u_i(q)$ is the buyer i's maximum willingness to pay for a product with quality q. The utility functions $u_i(q)$ are smooth, single-humped functions, and $u_i(0,\dots,0) = 0$, $\forall i \in I$. A utility function represents a group of identical buyers, and we assume that the buyers in a group always behave in the same way whenever they are called to act. Note that the buyers' preferences are linear in money, i.e., they enjoy a five dollar discount from a low quality product and a high quality product the same.

We define a *bundle* as a vector of product's qualities and the corresponding price. The firm designs a bundle for each buyer $i \in I$ and maximizes its profit

$$\pi(q_i,p_i) = \sum_{i=1}^{n} f_i \cdot (p_i - c(q_i)), \tag{2}$$

where f_i is the fraction (or weight) of buyers i in the population. Note that the cost is separable, which means that the costs of different bundles do not depend on each other and the fractions f_i are not in the argument of the cost function.

The profit is maximized under *individual rationality* (IR) constraints

$$U_i(q_i,p_i) = u_i(q_i) - p_i \geq 0, \quad \forall i \in I, \tag{3}$$

and *incentive compatibility* (IC) constraints

$$U_i(q_i, p_i) \geq U_i(q_j, p_j), \quad \forall i, j \in I, i \neq j. \tag{4}$$

The IR constraints can be formulated as IC constraints by defining a dummy type 0 with quality $q_0 = (0, \ldots, 0)$ and price $p_0 = 0$. Now, both constraints can be written symmetrically as

$$U_i(q_i, p_i) \geq U_i(q_j, p_j), \quad \forall i \in I, j \in I^+ \setminus \{i\}. \tag{5}$$

where $I^+ = \{0, \ldots, n\}$. The buyers are assumed to be friendly (Nahata et al 2004), i.e., they choose the most profitable bundle for the firm when they are indifferent between two or more bundles.

The firm's problem can now be formulated as

$$\max_{q_i, p_i} \sum_{i=1}^{n} f_i \cdot (p_i - c(q_i)) \tag{6}$$

$$s.t. \;\; u_i(q_i) - p_i \geq u_i(q_j) - p_j, \forall i \in I, \forall j \in I^+ \setminus \{i\}. \tag{7}$$

To make the problem mathematically more tractable, it is assumed that the *efficient* (first-best) qualities q_i^e defined by

$$\max_{q \in \mathbb{R}_+^k} u_i(q) - c(q), \tag{8}$$

are strictly positive and finite, and they satisfy $\nabla u_i(q_i^e) = \nabla c(q_i^e)$. We also assume that there is a quality q^m such that qualities above this limit are not profitable to produce, e.g., $u_i(q^m) < c(q^m)$, $\forall i \in I$. These assumptions guarantee the existence of the solution, and the conditions are satisfied in practical situations.

2.3 Illustrative examples

We present the learning methods in a simplified setting to illustrate the ideas better. The methods apply to the more general models and they do not require the following assumptions. The first assumption is that there are only two buyer types. This means that there are only two possible solution structures, which are explained shortly. The second assumption is that there is only one quality dimension and the single-crossing assumption is satisfied. With more buyer types and multiple dimensions, there are more possible solution structures and finding the optimal structure is more complicated (Berg and Ehtamo 2008, 2009; Nahata et al 2004).

With two buyer types, there are two characteristic solutions to the pricing problem: a) the firm can serve both types with efficient bundles, when the types are not interested in each other's bundles, and b) one bundle is distorted from the efficient quality to make it less attractive to the other type; see Figure 1. In a), the firm gets the maximal possible profit, and the buyers get zero

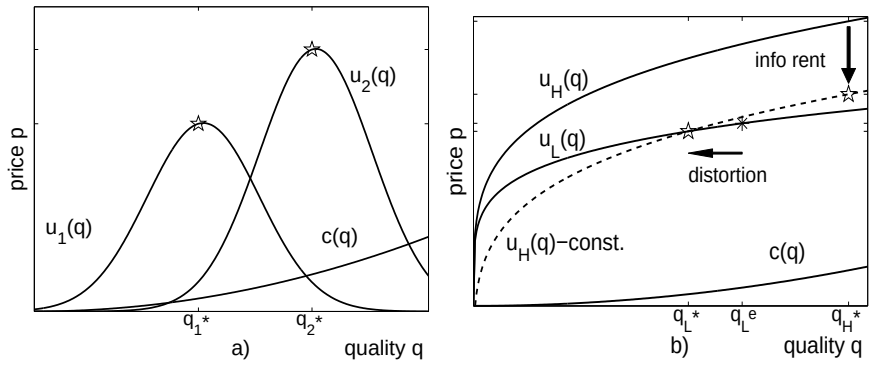

Fig. 1 Two characteristic solutions.

utilities. In b), the firm loses some of its profit, because the quality of type L is not efficient, i.e., $q_L^* \neq q_L^e$, and type H gets positive utility due to the price discount, which is called as *informational rent*.

In a), both IR constraints are active, the optimal qualities are given by $u_i'(q_i^*) = c'(q_i^*)$, i.e., $q_i^* = q_i^e$, and the optimal prices are $p_i^* = u_i(q_i^*)$, $i = 1, 2$. The firm can sell these bundles, since neither bundle attracts the other type, i.e., $u_i(q_j^*) - p_j^* < 0 = u_i(q_i^*) - p_i^*$, $i \neq j$. In b), there are two possibilities: either $q_L^* = 0$, or $q_L^* > 0$. When the fraction of type L, f_L, is relatively low and type L values the product relatively less, then type L is *excluded*, i.e., $q_L^* = 0$, and type H is served efficiently, i.e., $q_H^* = q_H^e$ and $p_H^* = u_H(q_H^*)$. In the more interesting situation when $q_L^* > 0$, the optimum is given by (Ehtamo et al 2010)

$$f_L(u_L'(q_L^*) - c'(q_L^*)) = f_H(u_H'(q_L^*) - u_L'(q_L^*)), \tag{9}$$
$$u_H'(q_H^*) = c'(q_H^*), \tag{10}$$
$$p_L^* = u_L(q_L^*), \tag{11}$$
$$p_H^* = p_L^* + u_H(q_H^*) - u_H(q_L^*). \tag{12}$$

Eq. (11) means that the IR constraint is active for type L (IRL from now on), i.e., type L gets zero utility. Eq. (12) means that type H is indifferent between the bundles, i.e., the IC constraint is active for type H (ICH from now on). Eq. (9) determines how much L's quality is distorted. The optimal quality q_L^* is a compromise between the profit $u_L(q_L) - c(q_L)$ weighted by f_L and the information rent $u_H(q_L) - u_L(q_L)$ weighted by f_H. The informational rent, which type H gets and the firm loses, is $u_H(q_H^*) - p_H^* = u_H(q_L^*) - u_L(q_L^*)$.

We assume the standard *single-crossing condition* (Spence 1980)

$$u_H'(q) > u_L'(q), \quad \forall q > 0. \tag{13}$$

This means that a) is no longer possible, and the optimum must be of characteristic b); this structure is more generally known as chain (Nahata et al

2004). We also assume that $q_L^* > 0$ and the second-order condition (Ehtamo et al 2010)

$$u_H''(q) \geq u_L''(q), \quad \forall q \geq 0, \tag{14}$$

which makes Eq. (9) as the sufficient optimality condition for L's quality, and the solution is unique. This follows from the fact that the convexity assumption (14) makes the function defined by Eq. (9) strictly monotone in q_L, and then it cannot have multiple roots.

3 Continuous learning methods

In the previous section the solution was characterized when the firm knows the buyers' utility functions. Now, we examine how the firm can learn the solution by adjusting the quality-price bundles continuously. It is assumed, for simplicity, that the firm knows the fractions of buyers, f_i, but only knows the utility functions, $u_i(q)$, and their slopes, $u_i'(q)$, locally around the currently sold bundles. We present different learning methods using the local information: the gradient method that takes the steepest ascent direction to the firm's profit, and modified methods that aim to solve the optimality conditions. In Section 3.4, the different criteria are presented to compare these methods, which include the discounted profit over the learning period, and the time it takes to learn the optimal bundles. In Section 5, it is discussed how the firm can get the local information by selling the product and collecting the sales data.

3.1 Gradient method

Let us define a continuous solution path $x(t) = (q_L(t), q_H(t), p_L(t), p_H(t))$, $t \geq 0$, starting from an initial solution $x_0 = x(0)$ and ending in x^*. The solution path is defined through differential equations, which give the rate of change locally. The gradient method is given by

$$\dot{x} = \nabla\pi(x) = \left(\frac{\partial\pi}{\partial q_L}, \frac{\partial\pi}{\partial q_H}, \frac{\partial\pi}{\partial p_L}, \frac{\partial\pi}{\partial p_H}\right)^T (x), \tag{15}$$

where the gradient $\nabla\pi(x)$ is the steepest ascent direction to the firm's profit at the current solution x; see a similar *best-response dynamics* in Hofbauer and Sigmund (2003) in a matrix game context. We remind that $\dot{x}(t) = (\dot{q}_L(t), \dot{q}_H(t), \dot{p}_L(t), \dot{p}_H(t))$. Now, we define $\nabla\pi(x)$ in different regions of x. We assume that x_0 is feasible, i.e., satisfies IR and IC constraints, and we only need to check IRL and ICH constraint during the iteration. Thus, we have four possible regions: (a) no constraints are active, (b) only IRL is active, (c) both IRL and ICH are active, and (d) only ICH is active.

In region (a), we have

$$\nabla\pi(x) = \begin{pmatrix} -f_L c'(q_L) \\ -f_H c'(q_H) \\ f_L \\ f_H \end{pmatrix}, \tag{16}$$

which we get simply by taking the partial derivates of the profit in Eq. (2).

In region (b), we need to compute the gradient while satisfying the constraint $p_L = u_L(q_L)$. We substitute the equation into the profit,

$$\pi = f_L(u_L(q_L) - c(q_L)) + f_H(p_H - c(q_H)), \tag{17}$$

and thus reduce the problem into three dimensions (q_L, q_H, p_H). The gradient is now computed by taking the appropriate partial derivates, and $\dot{p}_L(t)$ is solved by differentiating the constraint, i.e., $\dot{p}_L = u'_L(q_L)\dot{q}_L$. We get

$$\nabla_R\pi(x) = \begin{pmatrix} f_L(u'_L(q_L) - c'(q_L)) \\ -f_H c'(q_H) \\ u'_L(q_L)\dot{q}_L \\ f_H \end{pmatrix}, \tag{18}$$

where ∇_R defines the reduced gradient under the constraint $p_L = u_L(q_L)$, and $\dot{q}_L$ in the third component is equal to the first component, i.e., $\dot{q}_L = f_L(u'_L(q_L) - c'(q_L))$. By examining the update directions, we can see from the first equation that q_L is updated towards the efficient value q_L^e.

In region (c), we need to satisfy both constraints $p_L = u_L(q_L)$ and $p_H = p_L + u_H(q_H) - u_H(q_L)$. Again, we reduce the dimensions by substituting these into the profit function. By differentiating, we get

$$\nabla_R\pi(x) = \begin{pmatrix} f_L(u'_L(q_L) - c'(q_L)) - p_H(u'_H(q_L) - u'_L(q_L)) \\ f_H(u'_H(q_H) - c'(q_H)) \\ u'_L(q_L)\dot{q}_L \\ \dot{p}_L + u'_H(q_H)\dot{q}_H - u'_H(q_L)\dot{q}_L \end{pmatrix}, \tag{19}$$

where the last two components are solved by differentiating the corresponding constraints and $\dot{q}_L$, $\dot{q}_H$, and $\dot{p_L}$ are equal to the first three components, in the same way as $\dot{q}_L$ was defined in Eq. (18). The first two equations mean that q_L is updated towards the optimal value q_L^* and q_H towards $q_H^* = q_H^e$.

In the same way, we get for region (d)

$$\nabla_R\pi(x) = \begin{pmatrix} -f_L c'(q_L) - f_H u'_H(q_L) \\ f_H(u'_H(q_H) - c'(q_H)) \\ f_L + f_H \\ \dot{p}_L + u'_H(q_H)\dot{q}_H - u'_H(q_L)\dot{q}_L \end{pmatrix}. \tag{20}$$

To implement the gradient method, the firm needs an initial solution x_0 and the region of $x(t)$ for different time instances $t \geq 0$. The region depends on the active constraints, i.e., values p_L, $u_L(q_L)$, p_H, $u_H(q_H)$, and $u_H(q_L)$.

Actually, the firm needs not know $u_L(q_L)$ exactly, and it suffices to know whether $p_L < u_L(q_L)$ or not. If it happens that $p_L > u_L(q_L)$ then the firm observes this since type L does not simply buy the bundle. Similarly, the firm needs not know $u_H(q_H)$ and $u_H(q_L)$ exactly, and it is enough to evaluate whether $p_H < p_L + u_H(q_H) - u_H(q_L)$ or not. Again, if it happens that $p_H > p_L + u_H(q_H) - u_H(q_L)$ then the firm knows this since type H chooses type L's bundle.

The firm also needs to know $u'_L(q_L)$, $u'_H(q_H)$, and $u'_H(q_L)$ to compute the adjustment directions in Eqs. (16)-(20). We note that in region (a) the firm needs not know anything about the utility functions to make the adjustment. Indeed, $u'_L(q_L)$ is only needed when IRL is active, and $u'_H(q_H)$ and $u'_H(q_L)$ are only needed when ICH is active. Under the single-crossing condition (13) and Eq. (14) the gradient method, like all local methods, always converge to the optimal solution from any feasible starting point. When Eq. (14) does not hold there may be many local optima, where these methods may converge to.

3.2 Modified methods

The gradient method is only one of the possible learning methods for the problem. Now, we examine the directions that are both profit-increasing for the firm and acceptable for the buyers. It is also shown how the optimality conditions can be used in determining the update for the quality.

Let us examine the region (a) where none of constraints are active. Given some quality-price bundle, the directions that give more profit to the firm are given by the tangent plane of the cost function, see Fig. 2a). These directions are in the half-space of the tangent plane in the direction of increasing price. In region (a), all directions are acceptable for buyer i since buyer i gets positive utility, see Fig 2b). When IR or some IC constraint is active for buyer i, then the firm must make sure that buyer i gets the same or better utility from the new bundle than the current bundle. Now, the directions that are both more profitable for the firm and acceptable for buyer i are presented in Fig. 2c). The most profitable direction is the one going along buyer i's indifference curve, which corresponds to the direction given by the gradient method.

We can see that the firm has more options when the constraints are not active. Especially, the quality may either be increased or decreased. It is actually possible to narrow down these directions by requiring that the quality should be updated towards the optimal value. This can be done with local information by evaluating the optimality conditions if the optimal structure is known. For example, if the structure is a) of Fig. 1, then the quality update is determined by evaluating the condition $u'_i(q) = c'(q)$. If $u'_i(q) > c'(q)$ holds, then $q < q^*$ and q should be increased, and similarly if $u'_i(q) < c'(q)$ holds, then q should be decreased. If the structure is b) of Fig. 1, then the quality update for type L is determined by evaluating the condition of Eq. (9). Again, if $f_L(u'_L(q) - c'(q)) > f_H(u'_H(q) - u'_L(q))$, then $q < q^*$ and q should be increased. This result follows from the concavity of $u(q)$, convexity of $c(q)$, and

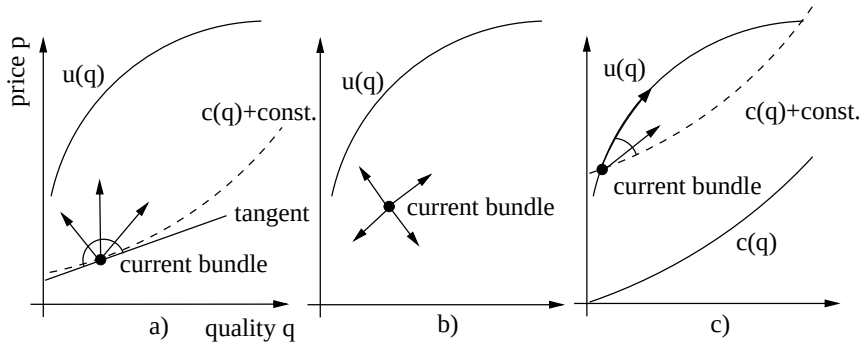

Fig. 2 Feasible update directions.

Eq. (14). This way, the firm can update the quality towards the optimal value using only local information.

We now define two modified methods: *price raise* and *constant direction* methods. In the price raise method, the adjustment direction is the same as in the gradient method, except when the constraints are not active for the bundle. When this happens, the adjustment direction is defined by $d_R = [\dot{q}_i \;\; \dot{p}_i] = [0 \;\; 1]$. This means that only the price is increased but the quality remains the same.

The constant direction method is defined in similar way. If there are no active constraints for a bundle, then the adjustment direction is $0.8d_R + 0.2d_C$, where $d_C = [\mp 1 \;\; c'(q)]$. The direction d_C is where the firm's profit remains locally the same for the bundle, and $\mp$ depends on the sign of the corresponding optimality condition of Eqs. (9) and (10) as discussed earlier. If it holds that $q_i < q_i^*$, then $+$ is chosen, and the quality of bundle i is increased. These two methods converge to the optimal solution like the gradient method, and they basically require the same information about the buyers' utility functions.

3.3 Optimal path by dynamic programming

Let us now compute the optimal path using complete information about the buyers' utility functions. The path is computed by using the dynamic programming algorithm (Bertsekas 2005). In this approach, the path is optimized over the learning period. It is assumed that the firm discounts the future profits, and the learning time is infinite. This means that the firm has enough time to learn the optimal solution, and there is no need to specify the length of the learning period. We also need to discretize the continuous quality-price space, define the bundles that are reachable in one time step from the current solution, and define the profit-to-go function for the algorithm. The computed path is not, however, optimal due to the discretization. But if the discretization is fine enough, the computed path is very close to the optimal continuous path.

We use a regular grid, which divides the four dimensions uniformly between some lower and upper bounds. The time is also discretized and denoted by $k = 0, 1, \ldots$, and x_0 is the initial solution. For each point $x(k) = (q_L(k), q_H(k), p_L(k), p_H(k))$, $k \geq 0$, we define a profit-to-go function $J_k(x_k)$ to present the future profits from the point x_k and time k on. In one time period, the qualities and prices can be moved one step in the uniform grid, i.e., each dimension can be increased, decreased, or kept the same. Thus, each point has $3^4 = 81$ neighboring bundles.

The optimal profit $J^*(x_0)$ is computed by doing value iterations

$$J_{k+1}(x_k) = \max_{y \in N(x_k)} \pi(y) + \delta J_k(y), \tag{21}$$

where $k \geq 0$, $N(x_k)$ are the neighbors of x_k, $\pi(y)$ is the instant profit of point y, and δ is the discount factor. The iteration is initialized by setting $J_0(x) = 0$ for all grid points. For the points on the edge of the grid we define $J_k(x) = -M < 0$, where $k \geq 0$ and M is large enough constant. Also, the value iteration (21) is only applied to the points that are not on the edge. In this way, the learning path stays inside the grid. The value iteration is repeated approximately k^* times, which is the number of steps it takes from x_0 to the final point.

We note that there is a modified version of the algorithm that converges in k^* iterations, and it gives $J^*(x_0)$ and the optimal path starting from x_0. In the modified version, the maximization in Eq. (21) is replaced by minimization, $-M$ by M, and $\pi(y)$ by $g(y) = \pi^* - \pi(y)$, where $g(y)$ is the instant cost, and π^* is the maximum profit on the grid. The stopping condition is when $J_{k+1}(x_0) = J_k(x_0)$, and it is in at most in k^* iterations. This is because the iteration finds in one step the correct values for the points that are one step away from the optimum, and similarly the values are correct after two iterations for the points that are two steps away from the optimum and so on.

3.4 Criteria for comparison

We examine four criteria when comparing the methods: the profits over the learning period, the present value of the profits, the time it takes to learn the optimal bundles, and the buyers' utilities over the learning period. The profits and the utilities are easily computed from the simulations but comparing the learning times is more problematic for two reasons. First, the methods in Sections 3.1 and 3.2 are continuous whereas the optimal path of Section 3.3 is discrete. Second, the way the differential equations are solved affects the learning path.

The continuous methods are simulated with an ordinary differential equation (ODE) solver, and it produces a sequence $x(t)$ for different time instances t. It is not, however, reasonable to compare the methods' simulation times, since the ODE solver and the magnitude of $\dot{x}(t)$ affects how $x(t)$ is computed, and thus how fast the optimum is found. Instead, we use the path length to

measure the learning time, and this does not depend so much on how the ODE is solved. By using the path length as the learning time, we compute the profits and present values the following way. For example, if the time step is Δ, we calculate profits $\pi(0)$, $\pi(\Delta)$, $\pi(2\Delta)$, ..., and so on. The present value is then the sum of discounted profits over some predetermined period. If the final time is T, the present value is $v = \sum_{t=0}^{T} \delta^t \pi(t) = \pi(0) + \delta\pi(\Delta) + \delta^2\pi(2\Delta) + \ldots$, where δ is the discount factor.

4 Numerical experiments

4.1 Learning and optimal paths

We examine gradient, price raise and constant direction methods in a test problem. The firm's cost function is $c(q) = q^2$, the buyers' utility functions are $u_L(q) = 2q^{1/5}$ and $u_H(q) = 3q^{1/4}$, and the fractions are $f_L = 0.7$ and $f_H = 0.3$. The optimal bundles are $(0.327, 1.60)$ to L and $(0.571, 1.939)$ to H, and the initial bundles are chosen as $(0.2, 1.2)$ and $(0.45, 1.5)$. The discount factor is $\delta = 0.95$, a time step $\Delta = 0.04$, and a final time $T = 0.44$. The grid is 24 points between the qualities 0.1 and 0.7, and 33 points between the prices 1.0 and 2.1. This makes approximately 600 000 grid points, and the quality and price steps are 0.025 and $0.0\overline{3}$, respectively.

The learning paths are presented in Fig. 3. The solid, dashed, and dash-dotted curves are gradient, price raise, and constant direction methods, respectively. The thicker curves are the buyers' and the firm's indifference curves. The letters (a)-(c) and the asterisks present the active constraints on the learning path for the gradient method, i.e., the different regions of Section 3.1. Finally, the black dots and the white stars are the initial and the optimal bundles, respectively.

We can see that the path for the gradient method is the longest and for the constant direction method the shortest. One reason for this is that the initial qualities are lower than the optimal ones and the gradient method decreases the quality until some constraint is active. On the other hand, the constant direction method always points towards the optimal quality, and thus finds the optimal quality faster.

We also observe that IRL is the first constraint to become active for the gradient method, i.e., the method switches from region (a) to region (b). From that on, the lower bundle is updated towards the efficient quality. But here ICH constraint becomes active before the efficient quality is reached, and the quality does not go over the optimal value, i.e., the method switches from (b) to (c) at quality 0.296, and it is lower than the optimal value 0.327. After ICH has become active, the quality approaches the optimal value. The higher bundle is updated towards the optimal quality only after ICH has become active. For the constant direction method, both qualities approach the optimal qualities immediately from the initial bundles.

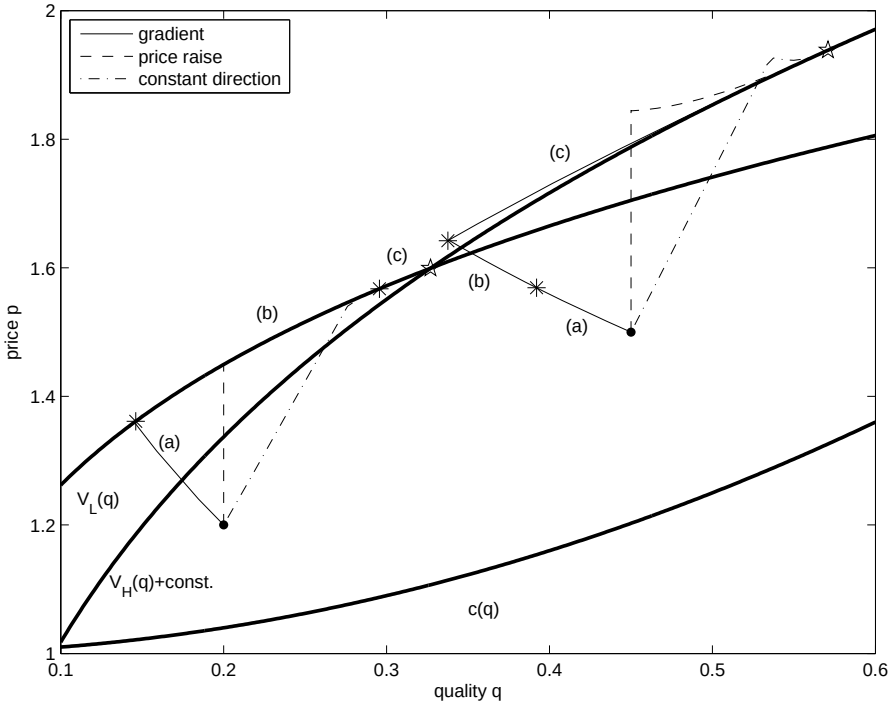

Fig. 3 The illustration of the learning methods.

The optimal path on the grid is presented in Fig. 4. The solid, dashed, and the dash-dotted curves are, respectively, the gradient method, the optimal path, and the *restricted* optimal path under the constraint that the high bundle cannot be sold to the low type. The asterisks are the end points of the optimal paths, and they give the best profit on the grid.

The optimal path looks similar to the gradient method, but it is characteristically different. The optimal path involves a shutdown of the low quality bundle, i.e., there is a period when both buyer types take the high quality bundle and nobody takes the low quality bundle. First, the low quality bundle is made less attractive for type L by decreasing the quality and increasing the price. When the bundle goes above L's utility function, L switches to H's bundle. From this on, the high quality bundle stays below L's utility function, and the low quality bundle approaches the optimal bundle from the infeasible side. A switch happens again when the low quality bundle comes below L's utility function again, and then the high quality bundle need not stay below the low type's utility function any more.

This is an interesting phenomena for the optimal path, since the continuous adjustment never finds this kind of jumps from a bundle to another. Of course, this happens only with some utility functions and initial bundles. Nevertheless,

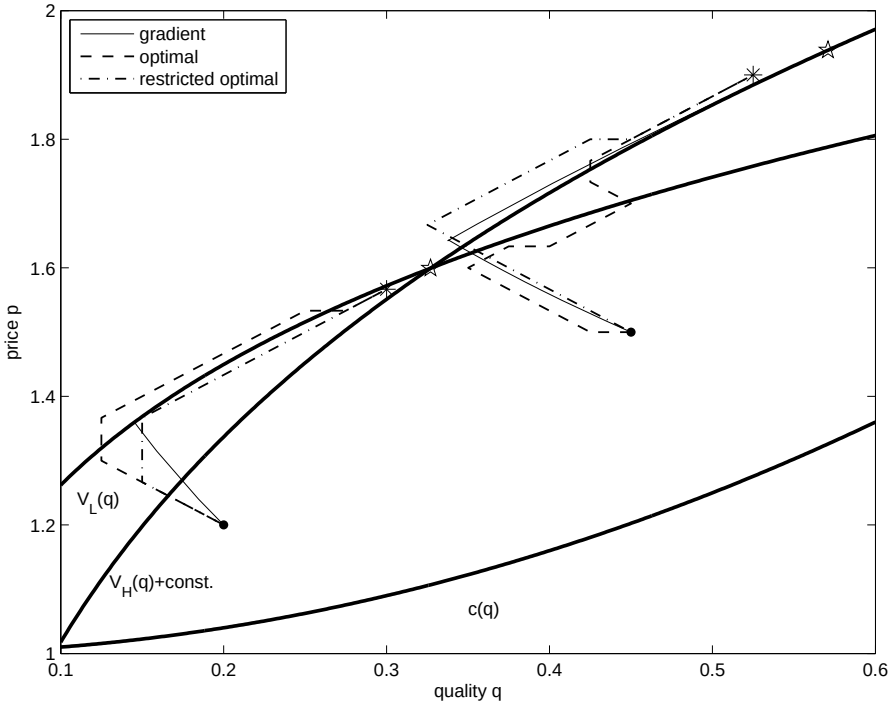

Fig. 4 The illustration of the optimal path.

to find the optimal path, the firm should also examine the option to erase and add bundles strategically, which makes the learning task more challenging.

The profits from each bundle as a function of path length are presented in Figs. 5a) and 5b); the thicker curves are for the optimal and the restricted optimal paths. For the low bundle, the optimal path gives dramatically better profits than the other methods, mainly due to the jump from the low to the high bundle. The gradient and the restricted optimal path look pretty similar by increasing the profit fast in the beginning but giving less profit in the end. The price raise and the constant direction method increase the profit more slowly but give more profit in the end. For the high bundle, the optimal path looks once again different than the others by getting less profit from the high bundle to make the overall profit bigger; remember the fractions $f_L = 0.7 > 0.3 = f_H$. For the other method, the pattern looks the same as for the low bundle.

The present values of the paths from 0 to 0.44 with step 0.04 are presented in Table 1. We can see the compromise the optimal path does by getting little lower profit from the high bundle but still getting clearly better profit from the low bundle. The price raise method gives the highest profit of the three methods. The restricted optimal path should give higher profit than the three

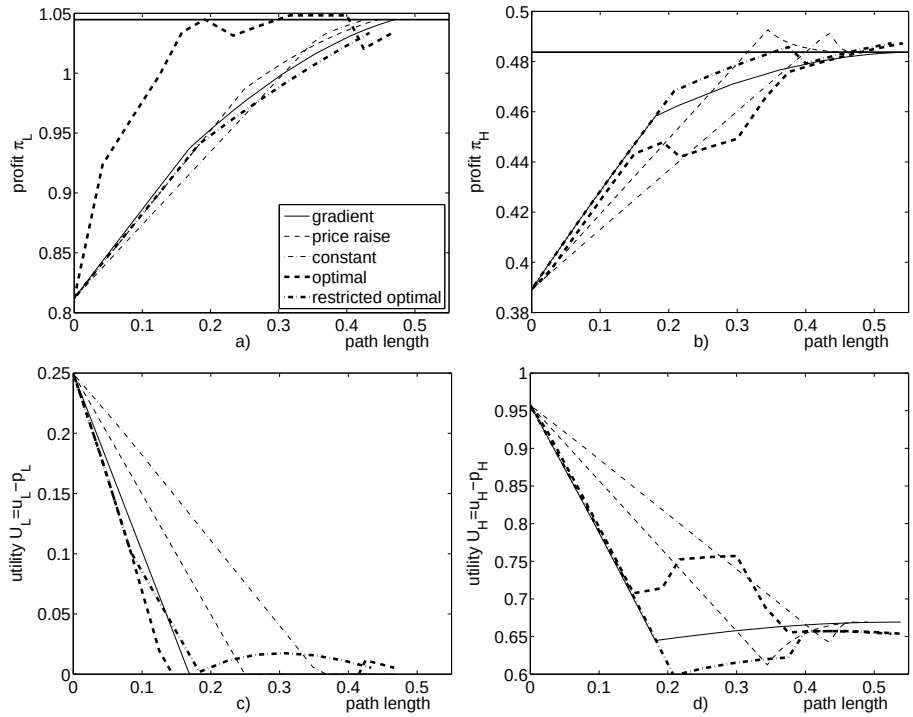

Fig. 5 The profits and the buyers' utilities.

methods but the discretization lowers the profit. For the same reason, the optimal path gives higher profit than the discrete path computed here.

Table 1 The present values (v) of the learning paths.

	gradient	price raise	constant	restricted	optimal
v_L	8.59	8.60	8.53	8.54	9.10
v_H	4.10	4.08	4.00	4.12	4.02
$f_L v_L + f_H v_H$	7.240	7.244	7.174	7.215	7.580

The buyers' utilities from each bundle are presented in Figs. 5c) and 5d). If we compare the learning methods, the gradient method decreases the buyers' utilities the fastest and the constant direction method the slowest. The difference to the low type is dramatic; at the point when the gradient method gives zero utility, the constant direction method gives half the initial utility to L. The differences in H's utility are also significant between the methods. From the buyers' perspective, the constant direction method is the most favorable.

4.2 Two-dimensional case

The methods were presented under the single-crossing condition and in a single-dimensional problem. Now, we demonstrate that the methods also work in the more general settings and discuss what needs to be modified. Let us examine the following two dimensional example. Let $c(q) = q_1^2 + q_2^2$, $u_L(q) = -\exp(1-1.5q_1)-\exp(1-2q_2)+5.6$, $u_H(q) = 2.5q_1^{1/5}+3q_2^{1/4}-1.0327$, $f_L = 0.6$, and $f_H = 0.4$; here, q_1 denotes the quality of dimension 1.

This is an interesting example because the solution is a mixture of the characteristic solutions of Section 2.3. The firm cannot sell the efficient bundles because H will take L's bundle, and Eq. (9) does not give the optimal quality for L, since then the price p_H in Eq. (12) is too high for H. The IRL, IRH, and ICH constraints are active, and the optimal bundles are $(0.777, 0.718, 4.11)$ to L and $(0.463, 0.571, 3.72)$ to H. H gets the efficient bundle but L gets over-efficient bundle, since $(0.777, 0.718) > (0.707, 0.687) = q_L^e$. Solving Eq. (9) gives the quality vector $(0.810, 0.732)$ to L, which is not correct, and the qualities are distorted too much. The correct equations are (Berg and Ehtamo 2008)

$$f_L(\nabla u_L(q_L^*) - \nabla c(q_L^*)) = \lambda(\nabla u_H(q_L^*) - \nabla u_L(q_L^*)), \tag{22}$$

$$u_H(q_L^*) = u_L(q_L^*), \tag{23}$$

$$0 \le \lambda \le f_H \tag{24}$$

where λ is the Lagrange multiplier of ICH constraint. We solve these equations by adjusting λ, and solving q_L from the first equation, and then evaluating the second equation. The correct value of $\lambda \approx 0.257$ can be found, e.g., by the bisection method since the equations are monotone in λ.

The gradient method for the problem is presented in Fig. 6. The black dots and stars, the upward-pointing and the left-pointing triangles are the initial and optimal bundles, the efficient bundle for L, and the bundle of Eq. (9) for L, respectively. The shaded and the see-through surfaces are the utility functions for L and H, respectively. We can see that the solution is between the triangles, i.e., it is a mixture of the two characteristic solutions q_L^e (corresponding $\lambda = 0$) and the full distorted bundle ($\lambda = f_H$) as in Eq. (9).

In the gradient path, IRL is the first constraint to become active and then ICH becomes active. The simulation was run with the equations of Section 3.1 extended to the multidimensional case with two exceptions. We check IR constraint for type H (IRH) and update accordingly if it becomes active before ICH. We also stop the iteration of L's bundle when all IRL, IRH, and ICH constraints are active, since then we are at the "mixed" optimum if we start from feasible bundles. This is the point when Eq. (23) is satisfied.

5 Discussion

This paper examines a monopoly pricing problem, which belongs to the class of mechanism design problems. The model is also a Stackelberg game, where the

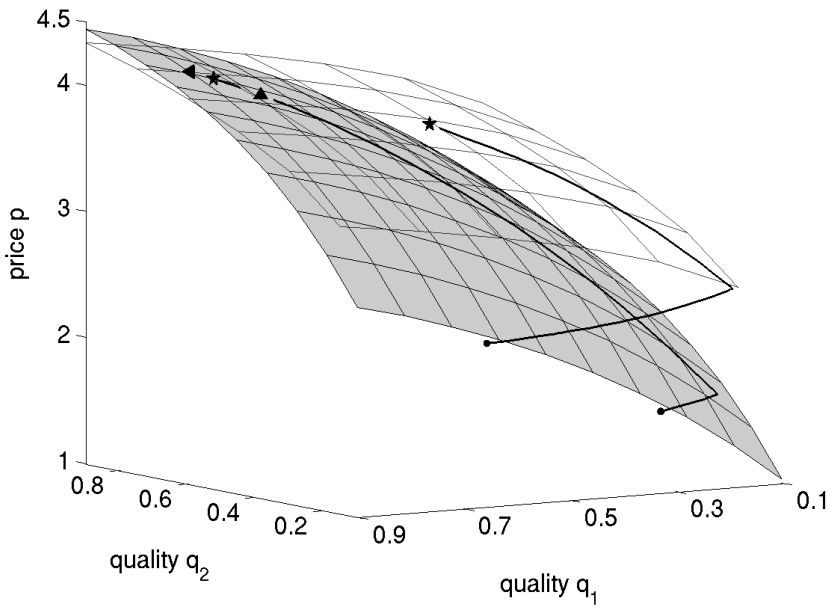

Fig. 6 The two-dimensional gradient path.

firm first designs a nonlinear tariff and then the buyers self-select the quality-price bundles they wish to consume. We study a setting where the firm has uncertainty about the buyers' preferences. Instead of doing a comprehensive demand data mining and computing the optimal tariff offline as in Wilson (1993), it is suggested that the solution is learned as the demand data is collected. This kind of situation could arise in new product development or in electronic commerce. In electronic marketplaces the prices can be updated rapidly and the offers can be tailored based on the different characteristics and conditions, like the buyer's purchase history and seasonal offerings. If the product is an information good, then the attributes can also be changed in real-time at low cost.

We propose various continuous learning methods and characterized the feasible adjustment directions for the problem, i.e., the directions that increase the firm's profit and are acceptable for the buyers. It is possible to compare the learning methods in terms of different criteria, and we notice that choosing a good method may be problematic. For example, the gradient method that uses the direction of greatest profit increase may not be the method that minimizes the learning time or maximizes the discounted profit on the learning path. An interesting future research direction is to study the learning methods with real demand data or with certain shapes of buyers' utility functions.

This paper characterizes the optimal path for the pricing problem. The path is computed by discretizing the problem and using dynamic programming.

We note that the optimal path may involve buyer types jumping from one bundle to another, which is something that is difficult to include in the learning methods. This issue is related to adding and removing the bundles strategically in order to increase the firm's profit. For example, it is not possible to estimate with local information the effect of adding a new bundle to the price schedule. This is something that just has to be tried and it adds complexity to the learning method.

It is assumed that there are only two buyer types, and the main example had only one quality dimension. These assumptions are made to simplify the notation, and the methods generalize to the multidimensional model with multiple buyer types with small modifications. The gradient and the price raise methods can be implemented in these models as well. The methods that adjust towards the optimal qualities are more problematic since it is more difficult to know the optimal structure of the problem and the correct equations for adjustment. This is a feature of the multidimensional model that follows from the possibly more general shapes of utility functions, i.e., the single-crossing condition may be violated and the buyers' valuations may change their ordering in different quality dimensions; see, e.g., Berg and Ehtamo (2008). What the firm can do is guess and update the structure of the solution based on the active constraints. We note that the multidimensional model may be more difficult to solve mathematically, but it may allow the firm to get more profits by tailoring suitable bundles for the buyer types.

The implementation of the adjustment can be interpreted in two ways (Ehtamo et al 2010). In the first, the monopoly gets the local information needed in (16)-(20), i.e., $u'_L(q_L)$, $u'_H(q_L)$, and $u'_H(q_H)$, by offering test bundles near the currently sold bundles and observing the realized sales. For example, the quality q_L can be changed a little and the price can be raised and lowered. By offering the two bundles side by side, and changing the price of the new bundle, the firm learns the two prices when the low and high types are indifferent between the bundles. With this information, the slopes of the buyers' utility functions near q_L can be estimated. In the second implementation, the same information can be revealed by offering linear tariffs. When the firm offers a linear tariff, a myopic buyer will choose a bundle from the tariff so that the slope of the tariff equals the slope of the buyer's utility function. Thus, the chosen bundle reveals the buyer's marginal valuation at the chosen quality, which makes it possible for the firm to create more profitable tariffs.

References

Araujo A, Moreira H (1999) Adverse selection problems without the single crossing property, econometric Society World Congress 2000 Contributed Papers 1874

Armstrong M (1996) Multiproduct nonlinear pricing. Econ 64(1):51–75

Armstrong M (2006) Recent developments in the economics of price discrimination. In: Blundell R, Newey WK, Persson T (eds) Advances in Economics

and Econometrics: Theory and Applications, Ninth World Congress vol. 2, Cambridge University Press, chap 4, pp 97–141

Basov S (2005) Multidimensional Screening. Springer, Heidelberg

Berg K, Ehtamo H (2008) Multidimensional screening: online computation and limited information. In: ICEC '08: Proc 10th Int Conf Electronic Commerce, ACM, New York, NY, USA, pp 1–10, URL http://www.sal.hut.fi/Publications/pdf-files/pber08.pdf

Berg K, Ehtamo H (2009) Learning in nonlinear pricing with unknown utility functions. Ann Oper Res 172(1):375–392, URL http://www.sal.hut.fi/Publications/pdf-files/mber07.pdf

Berg K, Ehtamo H (2010) Interpretation of Lagrange multipliers in nonlinear pricing problem. Optim Lett 4:275–285, URL http://www.sal.hut.fi/Publications/pdf-files/mber09.pdf

Bertsekas DP (2005) Dynamic Programming and Optimal Control. Athena Scientific, Belmont, Massachusetts

Bowling M, Veloso M (2002) Multiagent learning using a variable learning rate. Artif Intell 136:215–250

Braden D, Oren S (1994) Nonlinear pricing to produce information. Mark Sci 13:310–326

Brooks CH, Gazzale RS, Das RD, Kephart JO, Mackie-Mason JK, Durfee EH (2002) Model selection in an information economy: Choosing what to learn. Comput Intell 18(4):566–582

Conitzer V, Sandholm T (2002) Complexity of mechanism design. In: Proc 18th Annual Conf Uncertainty Artif Intell (UAI-02)

Ehtamo H, Berg K, Kitti M (2010) An adjustment scheme for nonlinear pricing problem with two buyers. Eur J Oper Res 201(1):259–266, URL http://www.sal.hut.fi/Publications/pdf-files/meht06.pdf

Ellison G (1997) One rational guy and the justification of myopia. Games Econ Behav 19:180–210

Elmaghraby W, Keskinocak P (2003) Dynamic pricing in the presence of inventory considerations: Research overview, current practices, and future directions. Manage Sci 49(10):1287–1309

Fudenberg D, Levine DK (1999) The Theory of Learning in Games. MIT Press

Garcia A, Campos-Nanez E, Reitzes J (2005) Dynamic pricing and learning in electricity markets. Oper Res 53(2):231–241

Hofbauer J, Sigmund K (2003) Evolutionary game dynamics. Bull Am Math Soc 40(4):479–519

Lin KY (2006) Dynamic pricing with real-time demand learning. Eur J Oper Res 174:522–538

Maskin E, Riley J (1984) Monopoly with incomplete information. Rand Journal of Economics 15:171–196

Mussa M, Rosen S (1978) Monopoly and product quality. J Econ Theory 18:301–317

Nahata B, Kokovin S, Zhelobodko E (2002) Package sizes, tariffs, quantity discounts and premium. Working paper, Department of Economics, University of Louisville, KY

Nahata B, Kokovin S, Zhelobodko E (2004) Solution structures in non-ordered discrete screening problems: Trees, stars and cycles. Working paper, Department of Economics, University of Louisville, KY

Nisan N, Ronen A (2001) Algorithmic mechanism design. Games Econ Behav 35:166–196

Raju Chinthalapati VL, Yadati N, Karumanchi R (2006) Learning dynamic prices in multiseller electronic retail markets with price sensitive customers, stochastic demands, and inventory replenishments. IEEE Trans Syst Man Cybern, part C 36(1):92–106

Rochet JC, Chone P (1998) Ironing, sweeping, and multidimensional screening. Econ 66:783–826

Rochet JC, Stole LA (2003) The economics of multidimensional screening. In: Dewatripont M, Hansen LP, Turnovsky SJ (eds) Advances in Economics and Econometrics 1, Cambridge University Press, pp 150–197

Sandholm T (2007) Perspectives on multiagent learning. Artif Intell 171:382–391

Spence M (1980) Multi-product quantity-dependent prices and profitability constraints. Rev Econ Stud 47:821–841

Stole LA (2007) Price discrimination and competition. In: Armstrong M, Porter R (eds) Handbook of Industrial Organization, vol. 3, North-Holland, Amsterdam, chap 34, pp 2221–2299

Vengerov D (2008) A gradient-based reinforcement learning approach to dynamic pricing in partially-observable environments. Future Gener Comp Syst 24(7):687–693

Wilson R (1993) Nonlinear pricing. Oxford University Press

Systems Analysis Laboratory
Research Reports, Series A

A104 June 2010	Participatory approaches to foresight and prority-setting in innovation networks Ville Brummer
A103 October 2009	Applications of stochastic modeling for investment decision-making under market uncertainties Janne Kettunen
A102 February 2009	The lignum functional-structural tree model Jari Perttunen
A101 October 2008	Portfolio decision analysis for robust project selection and resource allocation Juuso Liesiö
A100 May 2008	Innovation incentives and the design of value networks Toni Jarimo
A99 November 2007	Modeling and on-line solution of air combat optimization problems and games Janne Karelahti
A98 January 2007	Applications of decision analysis in the assessment of energy technologies for buildings Kari Alanne
A97 November 2006	Interactive multi-criteria decision support – new tools and processes for practical applications Jyri Mustajoki
A96 May 2006	Escaping path dependence – Essays on foresight and environmental management Totti Könnölä
A95 March 2006	Advanced mobile network monitoring and automated optimization methods Albert Höglund
A94 February 2006	Affine equations as dynamic variables to obtain economic equilibria Mitri Kitti
A93 September 2005	Decision modelling tools for utilities in the deregulated energy market Simo Makkonen
A92 June 2005	Portfolio optimization models for project valuation Janne Gustafsson
A91 May 2005	Electricity derivative markets: Investment valuation, production planning and hedging Erkka Näsäkkälä

Systems Analysis Laboratory
Research Reports, Series B

B26 June 2006	Systeemiäly 2006 Raimo P. Hämäläinen ja Esa Saarinen, toim.
B25 May 2005	Systeemiäly 2005 Raimo P. Hämäläinen ja Esa Saarinen, toim.
B24 June 2004	Systeemiäly - Näkökulmia vuorovaikutukseen ja kokonaisuuksien hallintaan Raimo P. Hämäläinen ja Esa Saarinen, toim.

Systems Analysis Laboratory
Research Reports, Series E
Electronic Reports: www.e-reports.sal.tkk.fi

E23 September 2008	Scenario-based portfolio selection of investment projects with incomplete probability and utility information Juuso Liesiö and Ahti Salo
E22 May 2008	Multi-criteria partner selection in virtual organizations with transportation costs and other network interdependencies Toni Jarimo and Ahti Salo
E21 May 2008	Markets for standardized technologies: Patent licensing with principle of proportionality Henri Hytönen, Toni Jarimo, Ahti Salo and Erkki Yli-Juuti
E20 August 2006	Smart-Swaps - Decision support for the PrOACT process with the even swaps method Jyri Mustajoki and Raimo P. Hämäläinen
E19 May 2006	Diversity in foresight: Insights from the fostering of innovation ideas Totti Könnölä, Ville Brummer and Ahti Salo
E18 June 2005	Valuing risky projects with Contingent Portfolio Programming Janne Gustafsson and Ahti Salo
E17 June 2005	Project valuation under ambiguity Janne Gustafsson and Ahti Salo
E16 June 2005	Project valuation in mixed asset portfolio selection Janne Gustafsson, Bert De Reyck, Zeger Degraeve and Ahti Salo

The reports are downloadable at http://www.sal.tkk.fi/en/publications/

Orders for paper copies:
Aalto University
Systems Analysis Laboratory
P.O. Box 11100, 00076 Aalto, Finland
systems.analysis@tkk.fi