Product Pricing using Adaptive Real-Time Probability of Acceptance Estimations based on Economic Regimes

Alexander Hogenboom
Erasmus Sch. of Econ.
Erasmus University Rotterdam
287485ah@student.eur.nl

Wolfgang Ketter
Rotterdam Sch. of Mgmt.
Erasmus University Rotterdam
wketter@rsm.nl

Jan van Dalen
Rotterdam Sch. of Mgmt.
Erasmus University Rotterdam
jdalen@rsm.nl

Uzay Kaymak
Erasmus Sch. of Econ.
Erasmus University Rotterdam
kaymak@ese.eur.nl

John Collins
Computer Science and Engr.
University of Minnesota
jcollins@cs.umn.edu

Alok Gupta
Carlson Sch. of Mgmt.
University of Minnesota
agupta@csom.umn.edu

ABSTRACT

In today's complex supply chains, product pricing is a vital, yet non-trivial task. We propose a product pricing approach using adaptive real-time probability of acceptance estimations based on economic regimes. Radial Basis Function Networks are trained to estimate parameters for double-bounded log-logistic distributions assumed to be underlying daily offer prices, using information available real-time. The relation between data and parameters is dynamically modeled using economic regimes (characterizing market conditions) and error terms (accounting for customer feedback). Given the parametric approximations of price distributions, acceptance probabilities are estimated using a closed-form mathematical expression, which is used to determine the price yielding a desired quota. The approach is implemented in the MinneTAC agent and tested against a price-following product pricing method in the TAC SCM game. Performance significantly improves; more customer orders are obtained against higher prices and profits more than double.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence; K.4.4 [**Computers and Society**]: E-commerce

General Terms

Algorithms, economics, experimentation, performance, theory

Keywords

Dynamic pricing, economic regimes, machine learning, supply chain management, TAC SCM

ICEC '09, August 12-15, 2009, Taipei, Taiwan

1. INTRODUCTION

In today's global economy, supply chains are everywhere. A supply chain is a complex logistics system in which raw materials are converted into products and distributed to the final users [8]. The main idea of supply chains is that every entity adds value to the final product and fulfills a function within the chain. Individual elements can be subject to optimization, as well as the supply chain as a whole.

Effective Supply Chain Management (SCM), focussing on more flexible and dynamic relationships between entities in the supply chain, is vital to the competitiveness of manufacturers within this chain. SCM can yield this effect, as it enables these manufacturers to respond to changing market demands in a timely and cost effective manner [5] and can thus improve a the agility of these manufacturers. Hence, research into performance optimization in a supply chain is important for the profit maximizing companies of today.

One of the challenges in this context is dynamic product pricing. When flexible and dynamic relationships between supply chain entities stimulate manufacturers to compete for customer orders, optimal product prices are to be determined while accounting for numerous aspects, such as competitors' strategies or market conditions [7, 19, 20, 29, 32]. This requires a sophisticated dynamic product pricing approach, which may be facilitated by automated decision support systems.

In real-time applications, the problem is that aspects relevant to the product pricing process may not be (fully) observable. Therefore, we investigate how dynamically differentiating product pricing decisions using estimations of economic regimes [15, 16, 18] can contribute to profit maximization. Our approach is validated in the Trading Agent Competition for Supply Chain Management (TAC SCM) [5], which has been organized since 2002 in order to promote and encourage high quality research into trading agents in supply chain environments. The game stimulates research with respect to more flexible and dynamic supply chain practices as opposed to the current common practices where supply chains are essentially static and rely on long-term relationships among key trading partners. In the TAC SCM game, several manufacturers compete in a component procurement market and in a sales market where assembled products are sold through reverse auctions where they can bid on requests for quotes (RFQs). The market is only partially observable.

In Section 2, we discuss related work on dynamic product pricing. Our own product pricing approach is introduced in Section 3. In Section 4, adaptivity is introduced to this approach. We evaluate the novel approach in the TAC SCM game in Section 5 and we conclude in Section 6.

2. RELATED WORK

An approximation of the probability of acceptance of offers can be used in the product pricing process, as done in [7] and [36]. Analyzing offer prices and their associated estimated probabilities of acceptance is rather intuitive in a product pricing process, because this can help a seller assessing how sales targets can be met. In [7], products are priced using a dynamic pricing algorithm which considers an estimated distribution of the buyer reservation price for products of a seller. Reversing the cumulative form of this distribution yields a function expressing the proportion of buyers willing to pay the seller a specified price, which can also be interpreted as the probability that a customer accepts an offered price. This function can subsequently be used to determine the price expected to yield a specified sales quota. In [36], estimated distributions of buyer's private values are used in a similar way.

Another way of modeling acceptance probabilities is by using linear regression on data points representing recent prices offered, along with the resulting acceptance rate [28]. Acceptance probability distributions could also be trained off-line [2]. Another option is to try to model the decision function of the accepting entities, based on their decision histories, e.g., using Chebychev polynomials [29].

Related work suggests some other aspects besides acceptance probability estimations to be taken into account as well when pricing products. For instance, current and future offers of competitors (outside options) could be considered [19, 20]. In [32], outside options are considered as well. Here, a dynamic product pricing model is proposed in which the price change of the product itself as well as the relative price of competing products is quantified in a price elasticity. Using scenario analysis for distinguishing between various situations of price elasticity (i.e., market conditions), the optimal pricing policy is selected.

Market conditions can also be accounted for using economic regimes [15, 16, 18]. A regime can be considered to be a set of conditions, characterizing the state of a system or process. Regimes provide an intuitive way of conditioning behavior in different scenarios. In literature, several approaches to regime identification and prediction have been proposed in different contexts [1, 11, 12, 26]. In an economic context, traders could take into account economic regimes because the ability of decision makers to correctly identify the current regime and predict the onset of a new regime is crucial in order to prevent over- or underreaction to market conditions [22]. Economic regimes can guide tactical (e.g., product pricing) and strategic sales decisions (e.g., product mix and production planning) [15, 16, 18].

Finally, in case of known demand and uncertain supply, a responsive pricing policy, in which the retail price is determined after observing the realized supply, results in a higher expected profit than a pricing policy in which the realized supply is not taken into account [33]. Consequently, modeling expected or observed supply-side behavior in the product pricing process could contribute to profit maximization.

3. A PRODUCT PRICING MODEL BASED ON OFFER PRICE DISTRIBUTIONS

Given the considerations presented in Section 2, we propose to approximate customer offer acceptance probabilities in a scenario as simulated in the TAC SCM game by taking into account the distribution of all offered prices. By considering a full price distribution, we model the decision making processes of all traders and obtain a complete estimation of customer acceptance probabilities, as opposed to for instance the linear regression on recent offer prices proposed in [28]. Incorporating offer price distributions rather than individual offers into the framework can compensate for a drawback encountered in [19], where offer prices of individual competitors are predicted even though these competitors may not actually bid, which structurally causes the trader's offers to be relatively low (i.e., lower than necessary in order to win the bidding process). When reasoning in terms of offer price distributions rather than individual offers, the phenomenon of taking into account non-existing offers is better accounted for, as the offer price distributions are formed by all offers actually done rather than offers all competitors would make, should they actually bid.

3.1 Dynamic Pricing

Assuming that customers only consider bids at or below their reservation price and that they always select the trader offering the requested product for the lowest price, the distribution of realized order prices can be derived from the distribution of valid offer prices. Modeling order price distributions using offer price distributions is intuitive, as this captures the market dynamics and facilitates the representation of relevant information on supply-side behavior. Now, let for product g on game day d a mean number of $\overline{n}_{dg}$ randomly sampled valid offer prices p_{dg} for each out of m_{dg} RFQs be identically and independently distributed in accordance with a distribution $f\left(p_{dg};\theta\right)$ and a cumulative distribution $F\left(p_{dg};\theta\right)$, with $0 < p_{dg} < u$ (i.e., prices are nonnegative and have an upperbound u) and θ a vector of unknown parameters. For such a distribution, the cumulative distribution of the minimum valid offer prices (and thus the order prices) $\underline{p}_{dg}$ over all m_{dg} RFQs can be derived as [14]

$$F_{\underline{p}}\left(\underline{p}_{dg};\theta\right) = \left(1-\left(1-F\left(\underline{p}_{dg};\theta\right)\right)^{\overline{n}_{dg}}\right)^{m_{dg}}, \quad 0 < \underline{p}_{dg} < u, \quad \overline{n}_{dg}, m_{dg} > 0. \tag{1}$$

The cumulative density of order prices, $F_{\underline{p}}\left(\underline{p}_{dg};\theta\right)$, yields the fraction of order prices realized at or below a specific value, which is similar to the probability that an order is placed with another trader offering a similar or better deal. Consequently, the reverse of this cumulative density approximates the probability for an agent to offer a better deal than other competitors. Hence, the reverse cumulative density of order prices associated with product g on game day d is an estimation of the probability that a customer will place an order o with an agent, given its offer price p_{dg}, $P\left(o|p_{dg}\right)$. Acceptance probabilities can therefore be estimated as

$$\begin{aligned} P\left(o|p_{dg}\right) &= 1 - F_{\underline{p}}\left(p_{dg};\theta\right), \quad 0 < p_{dg} < u \\ &= 1-\left(1-\left(1-F\left(p_{dg};\theta\right)\right)^{\overline{n}_{dg}}\right)^{m_{dg}}, \\ & \quad 0 < p_{dg} < u, \quad \overline{n}_{dg}, m_{dg} > 0. \end{aligned} \tag{2}$$

Equation (2) can be used to estimate the share of received orders with respect to the total number of RFQs for product g on game day d, generated by a price offered on all these RFQs. Now, let q^*_{dg} be the sales quota (i.e., desired acceptance probability [17]) for product g on day d, with m_{dg} associated RFQs, for each of which $\overline{n}_{dg}$ prices are offered. This implies that $P\left(o|p^*_{dg}\right)$ is required to be q^*_{dg}. Solving the equation to p^*_{dg} yields the optimal offer price p^*_{dg} expected to yield the desired quota. This way, products can be priced using estimations of offer price distributions.

3.2 Model Parameter Estimation

When pricing products, the unknown parameters θ and $\overline{n}_{dg}$ in the acceptance probability approximation detailed in (2) must be estimated for a product g on game day d. When all data is available, $\overline{n}_{dg}$ can be determined by a counting process. Furthermore, θ can be estimated by maximum likelihood as follows. Assuming all prices in the sample of prices $\vec{p}_{dg}$ offered for all m_{dg} RFQs issued for product g on day d to be identically and independently distributed in accordance with the offer price distribution $f\left(p_{dgri};\theta\right)$, the joint distribution of all valid offer prices can be derived as

$$f\left(\vec{p}_{dg};\theta\right) = \prod_{r=1}^{m_{dg}} \prod_{i=1}^{n_{dgr}} f\left(p_{dgri};\theta\right), \quad 0 < p_{dgri} < u, \tag{3}$$

where p_{dgri} is the ith of n_{dgr} prices offered on game day d for RFQ r for product g and parameters θ can be estimated by minimizing the negative log-likelihood function of these parameters for a sample of observed offer prices $\vec{p}_{dg}$ (e.g., using the Newton-Raphson method described in [4]):

$$L\left(\theta;\vec{p}_{dg}\right) = \sum_{r=1}^{m_{dg}} \sum_{i=1}^{n_{dgr}} -\ln\left(f\left(p_{dgri};\theta\right)\right), \quad 0 < p_{dgri} < u. \tag{4}$$

However, data on offer prices is not available in TAC SCM, as usually in the real business world, due to limited visibility of the real-time environment. Hence, the distribution parameters θ and $\overline{n}_{dg}$ can be estimated using a vector of real-time available information, $\vec{x}$. The relations between available information and distribution parameters, $h_\theta\left(\vec{x}\right)$ and $h_{\overline{n}_{dg}}\left(\vec{x}\right)$, can be modeled using Artificial Neural Networks (ANNs), yielding approximations of these relations: $\hat{h}_\theta\left(\vec{x}\right)$ and $\hat{h}_{\overline{n}_{dg}}\left(\vec{x}\right)$. An ANN is a mathematical model inspired by biological neural networks, which provides a general, practical method for learning real-valued, discrete-valued, and vector-valued functions over continuous and discrete-valued attributes from examples in order to facilitate regression or classification [24]. The model consists of interconnecting artificial neurons (nodes), ordered into an input layer, hidden layers, and an output layer.

Due to the ability of an ANN of capturing complex nonlinear relations, which is a useful feature in case of learning functions whose general form is unknown in advance, parameter estimation using (4) can be replaced with such a model, albeit with different inputs (i.e., real-time available data). Representing the unknown relations between distribution parameters and real-time available data using ANNs also brings the attractive feature of fast evaluation of these (modeled) functions, which is crucial in case of real-time product pricing. Other advantages include robustness to noise in the training data [24], the possibility to introduce adaptivity by adjusting the weights of each node's inputs on-the-fly using newly obtained examples (if any), and the fact that ANNs have proven to be useful for economic forecasts in various domains [19]. Moreover, our experimental results regarding TAC SCM showed that ANNs better captured the relation between data and distribution parameters than for instance a linear regression model.

We propose to use a specific type of ANN for parameter estimation: a radial basis function network (RBFN). An RBFN can be considered as a two-layer ANN consisting of a hidden layer and an output layer. The activation function in each hidden unit h is a kernel function $K_h\left(d\left(x_h,\vec{x}\right)\right)$, the output of which approximates 0 as $d\left(x_h,\vec{x}\right)$ – the (typically Euclidian) distance between an instance characterized by a vector of features $\vec{x}$ and the kernel center x_h – increases. The kernel functions in the hidden layer typically are Gaussians, centered at x_h with variance σ^2_h. The number of Gaussians H is subject to optimization and their centers can be determined by clustering the data, using for example the k-means algorithm [21]. The network's output for an instance $\vec{x}$, $\hat{h}\left(\vec{x}\right)$, is a linear combination of the activation units, weighted for their weights w_h, and a bias w_0 [24]:

$$\hat{h}\left(\vec{x}\right) = w_0 + \sum_{h=1}^{H} w_h K_h\left(d\left(x_h,\vec{x}\right)\right), \tag{5}$$

$$K_h\left(d\left(x_h,\vec{x}\right)\right) = e^{\frac{1}{2\sigma^2_h} d^2\left(x_h,\vec{x}\right)}. \tag{6}$$

Hence, an RBFN is a global approximation $\hat{h}\left(\vec{x}\right)$ of a target function $h\left(\vec{x}\right)$, represented as a linear combination of local approximations of this target function, as the contribution of each kernel is localized to a region around its center. Because RBFNs can be designed and trained in a fraction of the time it takes to train standard feed-forward back-propagation neural networks [24], an RBFN would be a good approximator for distribution parameters.

4. ADAPTIVE PRODUCT PRICING USING ECONOMIC REGIMES

The real-time applicable product pricing model proposed in Section 3 is static and does not adapt the modeled relations between price distributions and observable data to changing market conditions and market responses. In [36], an English auction scenario is considered, in which bidders have independent private values, all originating from the same distribution. These private values result in bids up to the private values. The best (highest) bid wins. The distribution of the private values of the bidders is estimated using averaging and binary search techniques, combined with simulations. Adaptivity to market disruptions is realized by assuming changes in bidding (and thus market disruptions) to actually be a shift in the underlying value distribution. The estimated private value distribution is shifted accordingly.

The product pricing scenario we target in our research is somewhat similar to the scenario described in [36]. In our case, traders bid on an RFQ. The best (lowest) bid wins. However, contrary to the English auction scenario, our RFQ bidding process much more resembles (reverse) sealed bid, first price auctions, as we assume traders not to be aware of bids of their competitors and the best (lowest) bid wins (as is the case in the TAC SCM game). Therefore, changes in bidding behavior of the competitors cannot be observed.

However, regime information might help here, as realized prices and hence order probabilities tend to vary, depending on the economic regime [15, 16, 18]. Hence, changes in pricing behavior can be accounted for by incorporating regime information into the process of estimating order price distributions and the associated customer offer acceptance probabilities [17]. Therefore, in order to facilitate a truly adaptive, real-time applicable product pricing approach, we propose to dynamically model the relations between available data and price distributions using economic regimes, such that product pricing strategies can be differentiated depending on market characteristics. To this end, per dominant regime k, price distribution parameters θ_k and $\overline{n}_{dgk}$ for product g on day d can be estimated using RBFNs. In case of M considered dominant regimes, this yields M separate price distribution estimations. The acceptance probabilities $P(o_k|p_{dgk})$ derived from these distributions can subsequently be weighted with the associated regime probabilities for regime R_{dgk}, $P(R_{dgk})$.

The weights in the RBFNs could be updated on-line, based on new data. However, when new training samples cannot be presented to the networks (due to limited visibility of market characteristics), daily approximations of customer offer acceptance probabilities – which use the parameters estimated by the RBFNs – can be adjusted by multiplying the acceptance probabilities by a factor representing the ratio between the number of actually received orders and the number of predicted orders, as proposed in [27]. If more orders have been received than predicted, the acceptance probability is larger than expected, to an extent equal to the ratio between received and predicted number of orders. If less orders have been received than predicted, the acceptance probability should be adjusted downwards. This ratio, which can also be referred to as a residual error term ϵ, enables market responses to be fed back to the model, as this ratio can be updated in real-time. A smoothed error term $\widetilde{\epsilon}$ can be used in order to prevent over- or undercompensation.

For dominant regime k, the probability that a customer accepts an offer and hence places an order o_k, given price p_{dgk} for product g on game day d, $P(o_k|p_{dgk})$, ranges from 0 to 1. Multiplying this probability with the suggested ratio $\widetilde{\epsilon}_{(d-1)gk}$ (which depends on regime k and has been updated using performance information up until day $d-1$) yields a corrected probability $P(o_k|p_{dgk})'$ in the range $\left[0, \widetilde{\epsilon}_{(d-1)gk}\right]$. This implies that no suitable price can be found for $q^*_{dg} \geq \widetilde{\epsilon}_{(d-1)gk}$, which is an undesirable feature in case $\widetilde{\epsilon}_{(d-1)gk} < 1$. However, when the corrected customer offer acceptance probability $P(o_k|p_{dgk})'$ is defined as

$$\begin{aligned} & P(o_k|p_{dgk})' \\ &= P(o_k|p_{dgk})^{\widetilde{\epsilon}_{(d-1)gk}}, \quad 0 < p_{dgk} < u, \quad \widetilde{\epsilon}_{(d-1)gk} > 0, \\ &= \left(1 - \left(1 - (1 - F(p_{dgk};\theta_k))^{\overline{n}_{dgk}}\right)^{m_{dg}}\right)^{\widetilde{\epsilon}_{(d-1)gk}}, \\ & 0 < p_{dgk} < u, \quad \overline{n}_{dgk}, m_{dg}, \widetilde{\epsilon}_{(d-1)gk} > 0, \end{aligned} \tag{7}$$

offer acceptance probabilities continue to range from 0 to 1 for $0 < p_{dgk} < u$ after correction.

Using (7), the corrected offer price $p^{*'}_{dg}$ expected to yield the desired sales quota q^*_{dg} for product g on day d can be defined for each dominant regime k, by requiring $P\left(o_k|p^{*'}_{dgk}\right)'$ for that product on that day to be q^*_{dg}. Solving the equation to $p^{*'}_{dgk}$ yields the optimal corrected offer price $p^{*'}_{dgk}$ expected to yield the desired quota q^*_{dg} under dominant regime k. When these corrected prices are then weighted for their associated regime probabilities $P(R_{dgk})$, the corrected price $p^{*'}_{dg}$ expected to yield the required quota can be obtained.

The error term considered in (7) should be assigned values such that under each dominant regime k, the expected customer offer acceptance probabilities $P\left(o_k|p^{*'}_{(d-1)g}\right)$ associated with an offer price $p^{*'}_{(d-1)g}$ are corrected by the unsmoothed exponential error terms to the proportion of actually received number of orders $q_{(d-1)g}$. The found error terms can subsequently be smoothed. Hence, offer price and customer response should be related as follows:

$$\begin{aligned} q_{(d-1)g} = P\left(o_k|p^{*'}_{(d-1)g}\right)^{\epsilon_{(d-1)gk}}, \quad 0 < q_{(d-1)g} < 1, \\ 0 < p^{*'}_{(d-1)g} < u, \quad \epsilon_{(d-1)gk} > 0. \end{aligned} \tag{8}$$

The error terms can be smoothed using double exponential smoothing (also referred to as Brown linear smoothing) [3], where the smoothing factor β is weighted for the associated regime probabilities in order for errors only to be attributed to the models responsible for these errors. Smoothing is done by linearly combining two components (see (12)), the first of which (defined in (10)) is a linear combination of the latest error (see (9)) and the previous first component. The second component (defined in (11)) is a linear combination of the first component and the previous second component.

$$\begin{aligned} \epsilon_{(d-1)gk} &= \frac{\ln\left(q_{(d-1)g}\right)}{\ln\left(P\left(o_k|p^{*'}_{(d-1)g}\right)\right)}, \\ & 0 < q_{(d-1)g} < 1, \quad 0 < P\left(o_k|p^{*'}_{(d-1)g}\right) < 1, \end{aligned} \tag{9}$$

$$\begin{aligned} \widetilde{\epsilon}'_{(d-1)gk} &= \beta P\left(R_{(d-1)gk}\right)\epsilon_{(d-1)gk} + \\ & \left(1 - \left(\beta P\left(R_{(d-1)gk}\right)\right)\right)\widetilde{\epsilon}'_{(d-2)gk}, \end{aligned} \tag{10}$$

$$\begin{aligned} \widetilde{\epsilon}''_{(d-1)gk} &= \beta P\left(R_{(d-1)gk}\right)\widetilde{\epsilon}'_{(d-1)gk} + \\ & \left(1 - \left(\beta P\left(R_{(d-1)gk}\right)\right)\right)\widetilde{\epsilon}''_{(d-2)gk}, \end{aligned} \tag{11}$$

$$\widetilde{\epsilon}_{(d-1)gk} = 2\widetilde{\epsilon}'_{(d-1)gk} - \epsilon''_{(d-1)gk}. \tag{12}$$

So far, the proposed framework assumes the offer prices to be distributed in accordance with a distribution the type and parameters of which have not been defined yet. For this purpose, we propose a log-logistic distribution, as this distribution covers a variety of shapes depending on the parameters α and γ and has the attractive feature that an analytical closed form expression exists for the cumulative density function. Moreover, this distribution appears to sufficiently describe the data in the TAC SCM game in over 60% of the analyzed price samples from historical game data[1], according to the Kolmogorov-Smirnov test [23] (when requiring the p-value to be over 0.05). The remaining samples could not sufficiently be described using a simple parametric distribution such as the log-logistic distribution due to the complex form of their true densities. The log-logistic distribution $f(p;\alpha,\gamma)$ and its cumulative form $F(p;\alpha,\gamma)$ [25], truncated such that the distribution is defined on the domain $0 < p < u$ and reparameterized such that α represents the median and γ quantifies the distribution tightness, can be described using (13) and (14).

[1]TAC SCM 2007 Semi-Finals and Finals (9323–9327tac5 and 7308–7312tac3) [30] and TAC SCM 2008 Semi-Finals and Finals (763–768tac02 and 794–799tac01) [35] game data.

$$f\left(p;\alpha,\gamma\right)=\frac{\left(\alpha^{-\gamma}-u^{-\gamma}\right)\gamma p^{-\gamma-1}}{\left(\alpha^{-\gamma}-2u^{-\gamma}+p^{-\gamma}\right)^{2}},\quad 0<p<u,\quad \alpha,\gamma>0, \tag{13}$$

$$F\left(p;\alpha,\gamma\right)=\frac{\alpha^{-\gamma}-u^{-\gamma}}{\alpha^{-\gamma}-2u^{-\gamma}+p^{-\gamma}},\quad 0<p<u,\quad \alpha,\gamma>0. \tag{14}$$

When $F\left(p_{dgk};\theta_k\right)$ in (7) is substituted for (14), the corrected price $p_{dg}^{*'}$ expected to yield a required sales quota (i.e., acceptance probability [17]) q_{dg}^{*} can be obtained as shown in (15) through (17). Here, let $1 \leq k \leq M$, with the number of considered regimes $M = 5$ [15, 16, 18].

$$q_{dg}^{*'}=\sqrt[\overline{n}_{dgk}]{1-\sqrt[m_{dg}]{1-\sqrt[\widetilde{\epsilon}_{(d-1)gk}]{q_{dg}^{*}}}},\quad 0<q_{dg}^{*}<1,\quad m_{dg},\overline{n}_{dgk},\widetilde{\epsilon}_{(d-1)g}>0, \tag{15}$$

$$p_{dgk}^{*'}=\left(\frac{u^{-\gamma_k}\left(\alpha_k^{-\gamma_k}\left(u^{\gamma_k}-2\alpha_k^{\gamma_k}\right)q_{dg}^{*'}+1\right)}{1-q_{dg}^{*'}}\right)^{-\frac{1}{\gamma_k}},\quad \alpha_k,\gamma_k>0, \tag{16}$$

$$p_{dg}^{*'}=\sum_{k=1}^{5}P\left(R_{dgk}\right)p_{dgk}^{*'},\quad 0<p_{dgk}^{*'}<u. \tag{17}$$

We now have a product pricing approach, which assumes a double-bounded log-logistic distribution to be underlying offer prices, the parameters of which can be estimated real-time using RBFNs, based on available information. This approach is capable of adapting to market disruptions, which are characterized using economic regimes, as product prices are determined per dominant regime using (15) and (16) and subsequently weighted for their associated regime probabilities in (17). The relations between price distributions and available information are thus dynamically modeled, depending on economic regimes.

Structural errors in the product pricing process are accounted for by feeding market responses to placed offers back into the product pricing model. In order for the product pricing approach to remain valid, market responses are fed back using an exponential error term, designed to transform the estimated probability of acceptance function into a function better approximating the true acceptance probability. This error term is corrected using daily observations of expected and observed acceptance probabilities – double exponentially smoothed with a smoothing factor weighted for the associated regime probabilities – using (9) through (12). This feedback process enables the product pricing model to adapt to the true customer offer acceptance probabilities.

5. PERFORMANCE IN TAC SCM

The final framework can be evaluated by implementing the approach in the MinneTAC trading agent [6] for the TAC SCM game. To this end, product pricing should be done using (9) through (12) and (15) through (17). The α_k, γ_k, and $\overline{n}_{dgk}$ parameters for product g on game day d for dominant regime k are to be estimated using RBFNs. This section elaborates on implementation and testing of the proposed approach.

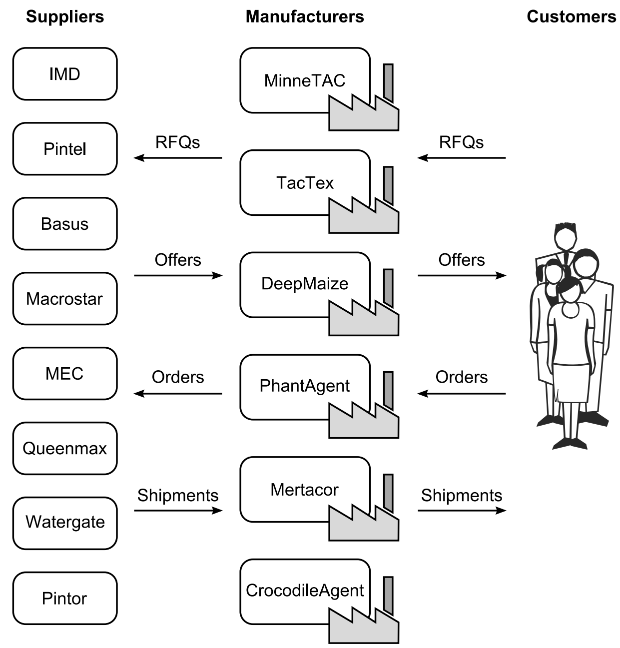

Figure 1: Schematic overview of a typical TAC SCM game scenario.

In the TAC SCM game, a supply chain for PCs is considered in 220 game days of 15 real-time seconds each. This supply chain consists of customers, manufacturers and suppliers (see Figure 1). Every game day, customers issue RFQs for 16 PC types, on which manufacturers can bid. Customers always place an order with the manufacturer offering the requested product for the lowest price (if this price is at or below their reservation price). The requested products are assembled by the manufacturers using components procured from suppliers. These manufacturers are software trading agents (such as MinneTAC) developed by competing teams that all try to maximize their profit over a game. The major challenge is the limited visibility of the market environment. Real-time available data consists of information about received RFQs and agent's own orders, the preceding day's minimum and maximum order price of each PC type, and aggregate market statistics issued every 20 days.

Order prices realized in the TAC SCM game tend to be volatile. Order price distribution characteristics change over time (see Figure 2, where prices are normalized by expressing these prices in terms of their associated production costs). First of all, the mean of these distributions changes over time. Furthermore, the order price distributions sometimes are tight (i.e., minimum and maximum order prices are close together), whereas the spread of prices is larger on other game days. The spikes in order prices are mostly caused by opportunistic or manipulative agents. As price distribution estimation is hard due to this volatility, TAC SCM is a good testbed for our product pricing approach.

5.1 Implementation in the MinneTAC Agent

The sales decisions made by the MinneTAC agent originate from price predictions based on microeconomic conditions, which are characterized for each individual market segment: economic regimes are identified and predicted.

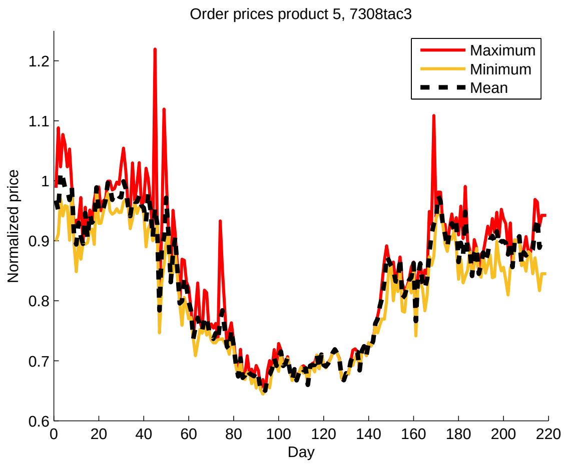

Figure 2: TAC SCM order price sample over time.

These economic regimes can be extreme scarcity, scarcity, a balanced situation, oversupply, and extreme oversupply. On game day d, the regime for good g is identified using regime probabilities; the regime having the highest probability, given the estimated normalized mean price of that day is assumed to be the current dominant regime. This price estimate is a smoothed normalized mid-range price, $\widetilde{\mathrm{np}}_{dg}$, which is the average of the double exponentially smoothed normalized minimum and maximum price. To this end, prices are first of all normalized by expressing these prices in terms of their associated production costs, such that these normalized prices np_{dg} range from 0 to 1.25.

A product-level price density function has been modeled on historical normalized order price data using a Gaussian Mixture Model (GMM) [34] with a sufficient number of Gaussians, reflecting a balance between prediction accuracy and computational overhead (currently in MinneTAC we are using a Gaussian Mixture Model with 25 Gaussians). Clustering these price distributions over time periods (using the k-means algorithm [21]) has yielded distinguishable statistical patterns: economic regimes. In the TAC SCM game, price information is only available up until the preceding game day. Hence, the MinneTAC agent approximates the mean price of day d using exponential smoothing prediction of $\widetilde{\mathrm{np}}_{dg}$ and subsequently returns the regime probabilities for day d. When such an approximation is supplied to the model during a game, the individual Gaussians in the model are activated to a certain extent, thus generating an expected price distribution. Subsequently normalizing all clusters' price densities enables determination of regime probabilities. Future regime probabilities are determined using Markov prediction and Markov correction-prediction processes (for short-term and long-term decision making, respectively) [17].

When current or future regime probabilities have been determined, products are priced based on the likelihood of customer acceptance of these prices, such that a sales quota is fulfilled. These processes of regime identification, regime prediction, and product pricing, as well as other processes in the MinneTAC trading agent, are supported by a configurable chain of evaluators [6]. The software selects the optimal configuration of evaluators.

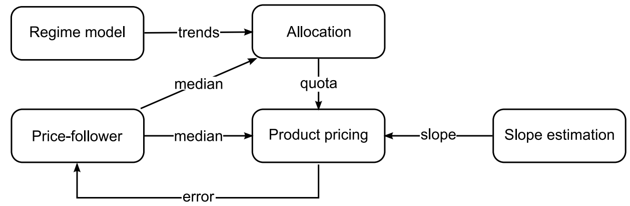

Figure 3: Simplified schematic overview of the benchmark MinneTAC sales process configuration for an arbitrary product on an arbitrary game day.

In the sales process configuration used as benchmark in our research, depicted in Figure 3, price trends are estimated by the regime model. The median price of a product is estimated using a price-following approach implementing a Brown linear exponential smoother. These trends, combined with the estimated median, are used in the allocation process, where sales quotas are generated based on – among other things – these price predictions. The curve representing the probability of acceptance is approximated using the estimated median price and the curve's slope in that median, estimated using exponentially smoothed prices. This customer offer acceptance probability is used for determining the price to be offered in order for the sales agent to sell its desired quota.

In order to compensate for the uncertainty in generated predictions, interval randomization is applied to offer prices, which adds a slight variability to these prices. The estimated median is corrected using feedback derived from the desired acceptance probability and the associated true acceptance probability observed the next day. The error is computed as the difference between the optimized offer price and the actual price. The latter price is derived by solving the acceptance probability estimate to the observed probability. A major drawback here is the assumption that customer feedback is in response to the optimized offer price, whereas this feedback in fact is in response to a price randomized in an interval around this price.

The price distributions estimated by the GMM are updated on-line, but they do not account for factors other than a mean price estimate and lack full adaptivity. In an attempt to improve the product pricing process by combining regime information with real-time available data, we replace the sales model of the benchmark with a system designed for Product Pricing using Adaptive Real-time Regime-based Probability of Acceptance Estimations: PPARRPAE. The algorithm (see Algorithm 1) involves parameter estimation using RBFNs and subsequently pricing products using (15) through (17) (with $u = 1.25$). In this process, an error term is considered, following (9) through (12) (with $\beta = 0.5$, as determined by a hill-climbing procedure). Figure 4 visualizes the relations between the logical components involved in this process.

The main idea is to leave the regime model intact and to build an adapter, which combines the characteristics of the price distribution estimated by the regime model with real-time available information. Using the RBFNs, the adapter transforms this information into a parameterized acceptance probability function per dominant regime and assigns weights to these functions, equal to their associated regime probabilities.

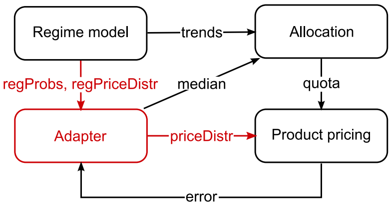

Figure 4: Simplified schematic overview of the proposed PPARRPAE sales process configuration for an arbitrary product on an arbitrary game day.

This adapted distribution can subsequently be used in the product pricing process for an arbitrary product on an arbitrary game day (see Section 4). Given a quota specified by the allocation component, the product pricing component uses the adapter to compute the price expected to yield this quota per dominant regime and weights the suggested prices for their associated regime probabilities. The generated optimal price is then offered on all selected RFQs for the considered product. The market responses to these offers are directly fed back to the adapter, which is able to learn from its errors. Therefore, in order for this information not to be biased, interval randomization is not applied to the generated optimal price, as opposed to the benchmark approach.

The allocation model bases its decisions – among other things – on price predictions, which consist of an estimate of the median price of the considered game day and trends representing expected future deviations from this median. In the benchmark sales model, the trends are estimated using the regime model, whereas the median is estimated using a price-follower approach. This price-following component is also used in the estimation of the daily probability of acceptance function and can thus be updated using market responses. Since in the proposed approach, market responses are not related to the price-following median, but are fed back to the adapter, the prediction of the median price should in this case be provided to the allocation component by the adapter.

5.2 Radial Basis Function Network Training

For each dominant regime k, an RBFN needs to be trained for estimating the α_k, γ_k, and $\overline{n}_{dgk}$ parameters for product g on game day d. Therefore, training and test datasets[2] must be split into datasets per dominant regime. These dominant regimes are identified by the current regime model.

As we attempt to adapt regime-based price distributions done using the GMM implemented in the MinneTAC agent in order for them to be useful in the daily product pricing process, these regime-based price distributions should be used as artificial neural network inputs. For now, let these distributions be described by their 10th, 50th, and 90th percentile, as well as the spread of these percentiles.

In the TAC SCM game, price distributions tend to differ per product type. Hence, the product type itself might be indicative of the characteristics of the order price distributions and hence the associated offer price distributions. Furthermore, offered prices might also be related to the game day, as for example in the first phase of the game, prices are more likely to be relatively high due to scarcity of products, which is caused by the fact that agents start with zero inventory.

[2]TAC SCM 2007 Semi-Finals and Finals (9321–9328tac5 and 7306–7313tac3) [30] and TAC SCM 2008 Semi-Finals and Finals (761–769tac02 and 792–800tac01) [35] game data. The first two games and the last game per server form the test set, the rest forms the training set.

```
foreach d in days do
    foreach g in products do
        // Update error using last feedback,
        // following (9) through (12)
        error = updateError(getFeedback(d − 1, g));
        // Retrieve data from regime model
        regProbs = getRegProbs(d, g);
        regPriceDistr = getRegPriceDistr(d, g);
        trends = getTrends(d, g);
        // Estimate parameters using RBFNs
        priceDistr = estParams(regPriceDistr,
        getData(d, g));
        // Determine median price using (15)
        // through (17)
        median = priceForProb(0.5, priceDistr, error,
        regProbs);
        // Retrieve allocated quota
        quota = getQuota(d, g, median, trends);
        // Determine optimal price expected to
        // yield quota using (15) through (17)
        price = priceForProb(quota, priceDistr, error,
        regProbs);
        // Bid optimized price on selected RFQs
        priceProduct(d, g, price);
    end
end
```

Algorithm 1: The PPARRPAE approach.

Another indicator for a product's offer price distribution could be the number of RFQs for that product, as the number of simultaneously run similar auctions affects the revenue generated from these auctions due to their (partial) substitutivity [36]. This might hold for TAC SCM too, as each bidding agent is restricted by its limited product capacity. Even more, RFQs for the same product type could be considered to be (partial) substitutes to some of the bidders (depending on their product pricing and RFQ selection strategy). Not only the number of RFQs, but the characteristics of these RFQs as well could be indicators of the pricing behavior they generate. Hence, the mean and standard deviation of requested quantities, requested leadtimes, and reservation prices could be taken into consideration.

Prices realized on the preceding day could also be useful [19]. In-game, as in real-world scenarios, only the minimum and maximum order prices realized on the preceding game day are available for each product. We can only estimate the mid-range price, and therefore an accurate estimate of the price density is of paramount importance. These prices and their associated mid-range and spread could be double exponentially smoothed as well, as this provides a good approximation of the mean price of the preceding game day [15, 16, 18].

Hence, the RBFNs should be trained to adapt regime-based GMM price distribution estimations using available data on product type, game day, RFQ characteristics, and observable prices as predictors. Using historical data in the training and test sets, the target values for $\overline{n}_{dgk}$ can be determined by a counting process and the α_k and γ_k parameters can be determined by fitting distributions using (4) and (13). The performance of models trained on the training set can be evaluated on the games in the test set, as the latter set is sufficiently large and representative [24]. An average training dataset thus generated contains over 15,000 samples, an average test dataset over 8,000.

The thus found optimal values for the γ_k parameter tend to be distributed on an exponential scale; the increment in γ_k needed to tighten the distribution increases as the distribution gets tighter. E.g., a distribution with a γ_k value of 2 is much more different from one with a γ_k value of 5 than a distribution with a γ_k value of 200 is from one with a γ_k value of 500, when all other parameters are fixed. Hence, as the required accuracy decreases for an increasing γ_k, the networks are trained to predict the natural logarithm of γ_k.

Using Weka [37], the RBFNs can be trained relatively easily. The results can subsequently be saved as serialized Java objects, which enables them to be used in Java software like the MinneTAC agent. One drawback of using Weka is that the Weka implementation of an RBFN, `RBFNetwork`, can have only one output. Hence, a network is to be trained per dominant regime per parameter.

Some parameters can be adjusted in the `RBFNetwork` implementation. First of all, the random seed used in the clustering process used to determine the centers of the Gaussians in the networks can be defined. Let this cluster seed be 0 for all networks. One can also specify a so-called ridge value, which indicates how much the regression error in estimating model parameters may diverge from the least squares measure. For all networks, this value is left at its default value, 1E-08. Other parameters are the number of clusters and the minimum standard deviation of these clusters.

The configurations of the latter two parameters can be determined by systematically evaluating all combinations of different values. The configurations yielding the lowest root mean squared deviation (RMSD) on the test set are selected [24]. The RMSD can be defined as

$$\text{RMSD} = \sqrt{\frac{\sum_{\omega=1}^{\Omega} \left(\hat{x}_\omega - x_\omega\right)^2}{\Omega}}, \tag{18}$$

where $\hat{x}_\omega$ is an observation in a set of Ω observations (in this case a value predicted by the model), the associated benchmark value in the test set of which is x_ω.

The optimal number of clusters could be anything between relatively small and rather large. Using too many clusters would cause the model to not generalize very well. Hence, taking into account the size of the dataset, the set of number of clusters considered is $\{25, 50, 100, 150, 200, 300\}$ and standard deviations in the set $\{1, 2, 5, 10, 15\}$ are considered. Apparently, α_k (ranging between 0 and 1.25) can be estimated relatively well, whereas $\overline{n}_{dgk}$ (ranging between 0 and 6) and $\ln(\gamma_k)$ (roughly ranging between -6 and 6) cannot (see Table 1).

5.3 Adaptive Product Pricing in the TAC SCM Game based on Economic Regimes

By running and analyzing a number of games, the performance of the PPARRPAE system proposed in Section 5.1 can be compared with the benchmark. In this experimental setup, games are in accordance with the 2006 TAC SCM game specifications [5]. The randomness incorporated in the game (e.g., in customer demand) is an inconvenient characteristic for a testing environment in which two approaches are to be compared, as this randomness in market conditions implies that many experiments should be run in order to obtain results with any statistical significance.

Parameter	Regime	Clusters	MinStdev	RMSD
α_k	1	25	15	0.0448
α_k	2	50	10	0.0346
α_k	3	100	5	0.0366
α_k	4	50	5	0.0386
α_k	5	300	5	0.0400
$\ln(\gamma_k)$	1	100	15	0.7713
$\ln(\gamma_k)$	2	150	5	0.6903
$\ln(\gamma_k)$	3	150	5	0.6481
$\ln(\gamma_k)$	4	200	5	0.6370
$\ln(\gamma_k)$	5	150	2	0.6732
$\overline{n}_{dgk}$	1	50	15	1.0036
$\overline{n}_{dgk}$	2	25	5	1.0773
$\overline{n}_{dgk}$	3	200	5	0.9974
$\overline{n}_{dgk}$	4	300	2	0.9395
$\overline{n}_{dgk}$	5	100	5	0.8090

Table 1: Optimized configuration of number of clusters and minimum standard deviation of clusters for RBFNs estimating distribution parameters, along with the RMSD of parameter values predicted by the models from their target values.

The issue of randomness in the testing environment is tackled by a controlled TAC SCM game server, in which random seeds used for generating market conditions can be controlled. Random elements in decision processes of competing agents cannot be controlled. Hence, multiple runs with the same random seeds for market conditions could still yield different results. However, under controlled market conditions, such uncontrolled stochastic behavior of participating trading agents does not have a significant impact on the agent profit levels [31]. The results presented in [31] also indicate that most significant profit differences between agents can already be detected in approximately 40 games.

The performance of the PPARRPAE system can hence be evaluated in 40 experiment sets on a controlled server. Each experiment set consists of a paired evaluation of the performance of the benchmark and the PPARRPAE system under equal market characteristics. For now, let the competitors be the default competitors that come with the TAC SCM game. These competitors use a make-to-order strategy.

In each evaluation, the final bank account balance can be considered, as well as the sales performance. To this end, the mean and standard deviation of account balances over all games can be computed. The number of obtained orders should be considered in the analysis as well. The number of times the agent proceeds to actually bidding on RFQs, given an acceptance probability estimate, can also be analyzed.

Performance differences should also be assessed with respect to their statistical relevance. This can be done with a paired Student's t-test [10], which tests whether pairs in two samples are identically distributed (this null hypothesis is rejected at a significance level below 0.05). However, this statistic assumes the observations to be distributed in accordance with a normal distribution and we do not know whether this is a realistic assumption. Therefore, we can also assess statistical relevance of observed performance differences using paired, two-sided Wilcoxon signed-rank test [9, 13]. This is a non-parametric test, which tests the hypothesis that the differences between paired observations are symmetrically distributed around a median equal to 0. If

	Benchmark		PPARRPAE	
Agent	Mean	Stdev	Mean	Stdev
MinneTAC	19.2614	12.4207	49.3933	2.7053
Make-to-order-1	12.9194	3.2799	14.0436	2.9310
Make-to-order-2	13.0250	3.3152	14.1313	3.1668
Make-to-order-3	12.7687	3.3184	14.1034	2.7711
Make-to-order-4	12.8552	3.4148	14.3529	2.9034
Make-to-order-5	13.0803	3.2224	14.2307	2.9874

Table 2: Mean and standard deviation of final bank account balance per agent, calculated over all experiments. Values are expressed in millions.

	Benchmark		PPARRPAE	
Agent	Mean	Stdev	Mean	Stdev
MinneTAC	3.0865	1.0178	4.6474	0.4507
Make-to-order-1	3.2615	0.3367	3.0498	0.3129
Make-to-order-2	3.2615	0.3291	3.0571	0.3448
Make-to-order-3	3.2452	0.3386	3.0364	0.3267
Make-to-order-4	3.2198	0.3292	3.0387	0.3172
Make-to-order-5	3.2504	0.3480	3.0437	0.3301

Table 3: Mean and standard deviation of number of obtained orders per agent, calculated over all experiments. Values are expressed in thousands.

this null hypothesis is rejected (at a significance level below 0.05), the compared sets of samples can be assumed to be significantly different. This test would be suitable in this experimental setup, as the distribution of the values to be compared is unknown.

Over all experiments, PPARRPAE turns out to outperform the benchmark with respect to final bank account balance and the number of obtained orders (see Tables 2 and 3). Generally, when using PPARRPAE, final balances significantly increase with about 160% (with a p-value of 0.0000 for both the paired t-test and the Wilcoxon test) and the number of orders significantly increases with over 50% (with a p-value of 0.0000 for the paired t-test as well as for the Wilcoxon test) with respect to the benchmark.

The increase in number of obtained orders can be explained by the significant increase of the number of usable acceptance probability estimations with approximately 80% (with a p-value of 0.0000 for both the paired t-test and the Wilcoxon test). However, final account balance increase does not appear to be fully explained by an increase in obtained orders. In some experiments, a small increase (or even a decrease) in the number of obtained orders still results in doubled profits. This indicates that orders are better priced. This could be caused by prices of obtained orders to be closer to second-lowest prices, instead of being significantly lower, which results in a reduced margin between customers' reserve prices and realized order prices. Hence, using the PPARRPAE approach improves the quality of acceptance probability estimations and consequently results in better bid efficiency.

6. CONCLUSIONS AND FUTURE WORK

When product pricing strategies are linked to price distribution estimations, taking into account real-time available information, the relation between data and price distribution parameters can be dynamically modeled using economic regimes (characterizing market conditions) and error terms (accounting for customer feedback). Thus, in a constrained environment like TAC SCM, economic regime estimations turn out to contribute to profit maximization when they are used to differentiate product pricing strategies. Test results indicate that this novel approach significantly improves the performance of a price-following product pricing approach; more orders are obtained against higher prices and profits more than double. Much gain is obtained from using proper statistical methods, combined with effective real-time parameter estimation.

Even though the performance of the proposed model already is very promising, some aspects still require more research. First of all, the type and parameterization of models for real-time price distribution and acceptance probability approximation could be further improved. Other possible predictors for acceptance probabilities could be considered as well. Procurement information might be a good candidate here, as costs associated with specific orders could influence the price, depending on the cost allocation applied in the participating trading agents. Another option for future research is trying to use the improved acceptance probability estimations in the allocation or RFQ selection process.

Finally, our approach of product pricing using adaptive regime-based acceptance probability estimations could be challenged in a situation with very tough competition. Currently, we are testing our approach against world's leading TAC SCM agents. If the MinneTAC agent could deal with those agents as with the agents considered in this research, MinneTAC would be more competitive than ever.

7. REFERENCES

[1] R. Becker, S. Hurn, and V. Pavlov. Modelling Spikes in Electricity Prices. *The Economic Record*, 82(263):371–382, 2007.

[2] M. Benisch, A. Sardinha, J. Andrews, and N. Sadeh. CMieux: Adaptive Strategies for Competitive Supply Chain Trading. *ACM SIGecom Exchanges*, 6(1):1–10, 2006.

[3] R. G. Brown, R. F. Meyer, and D. A. D'Esopo. The Fundamental Theorem of Exponential Smoothing. *Operations Research*, 9(5):673–687, 1961.

[4] T. Coleman and Y. Li. An Interior Trust Region Approach for Nonlinear Minimization Subject to Bounds. *SIAM Journal on Optimization*, 6(2):418–445, 1996.

[5] J. Collins, R. Arunachalam, N. Sadeh, J. Eriksson, N. Finne, and S. Janson. The Supply Chain Management Game for the 2006 Trading Agent Competition. Technical report, Carnegie Mellon University, November 2005.

[6] J. Collins, W. Ketter, and M. Gini. Flexible decision control in an autonomous trading agent. *Electronic Commerce Research and Applications*, Forthcoming 2009.

[7] P. Dasgupta and Y. Hashimoto. Multi-Attribute Dynamic Pricing for Online Markets Using Intelligent Agents. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2004)*, pages 277–284, Washington, DC, USA, 2004. IEEE Computer Society.

[8] G. Ghiani, G. Laporte, and R. Musmanno. *Introduction to Logistics Systems Planning and Control*. John Wiley & Sons, Ltd, 2004.

[9] J. D. Gibbons. Nonparametric Statistical Inference. *Technometrics*, 28(3):275, 1986.

[10] C. H. Goulden. *Methods of Statistical Analysis*. John Wiley & Sons, Ltd, 2nd edition, 1956.

[11] J. D. Hamilton. A New Approach to the Economic Analysis of Nonstationary Time Series and the Business Cycle. *Econometrica*, 57(2):357–384, March 1989.

[12] R. Hathaway and J. Bezdek. Switching Regression Models and Fuzzy Clustering. *IEEE Transactions on Fuzzy Systems*, 1(3):195–204, 1993.

[13] M. Hollander and D. A. Wolfe. Nonparametric Statistical Methods. *Journal of the American Statistical Association*, 95(449):333, 2000.

[14] A. S. Kapadia, W. Chan, and L. A. Moyé. *Mathematical Statistics with Applications*. CRC Press, 2005.

[15] W. Ketter, J. Collins, M. Gini, A. Gupta, and P. Schrater. Identifying and Forecasting Economic Regimes in TAC SCM. In H. L. Poutré, N. Sadeh, and S. Janson, editors, *AMEC and TADA 2005, LNAI 3937*, pages 113–125. Springer Verlag Berlin Heidelberg, 2006.

[16] W. Ketter, J. Collins, M. Gini, A. Gupta, and P. Schrater. A Predictive Empirical Model for Pricing and Resource Allocation Decisions. In *Proceedings of the Ninth International Conference on Electronic Commerce (ICEC 2007)*, pages 449–458, Minneapolis, MN, USA, 2007.

[17] W. Ketter, J. Collins, M. Gini, A. Gupta, and P. Schrater. Tactical and Strategic Sales Management for Intelligent Agents Guided By Economic Regimes. ERIM Working paper, reference number ERS-2008-061-LIS, 2008.

[18] W. Ketter, J. Collins, M. Gini, A. Gupta, and P. Schrater. Detecting and Forecasting Economic Regimes in Multi-Agent Automated Exchanges. *Decision Support Systems*, Forthcoming 2009.

[19] Y. Kovalchuk and M. Fasli. Adaptive Strategies for Predicting Bidding Prices in Supply Chain Management. In *Proceedings of the Tenth International Conference on Electronic Commerce (ICEC 2008)*, Innsbruck, Austria, 2008.

[20] C. Li, J. Giampapa, and K. Sycara. Bilateral Negotiation Decisions with Uncertain Dynamic Outside Options. In *First IEEE International Workshop on Electronic Contracting*, pages 54–61, Washington, DC, USA, July 2004. IEEE Computer Society.

[21] J. MacQueen. Some Methods of Classification and Analysis of Multivariate Observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

[22] C. Massey and G. Wu. Detecting Regime Shifts: The Causes of Under- and Overreaction. *Management Science*, 51(6):932–947, June 2005.

[23] F. Massey. The Kolmogorov-Smirnov Test for Goodness of Fit. *Journal of the American Statistical Association*, 46(253):68–78, 1951.

[24] T. M. Mitchell. *Machine Learning*. McGraw-Hill Series in Computer Science. McGraw-Hill, 1997.

[25] A. Mood, F. Graybill, and D. Boes. *Introduction to the theory of statistics*. McGraw-Hill, 1974.

[26] T. D. Mount, Y. Ning, and X. Cai. Predicting price spikes in electricity markets using a regime-switching model with time-varying parameters. *Energy Economics*, 28(1):62–80, January 2006.

[27] D. Pardoe and P. Stone. Predictive Planning for Supply Chain Management. In *Proceedings of the Sixteenth International Conference on Automated Planning and Scheduling*, pages 21–30, June 2006.

[28] D. Pardoe and P. Stone. TacTex-05: A Champion Supply Chain Management Agent. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, pages 1489–94, July 2006.

[29] S. Saha, A. Biswas, and S. Sen. Modeling Opponent Decision in Repeated One-Shot Negotiations. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 397–403, New York, USA, 2005. ACM.

[30] SICS AB. Trading Agent Competition (SICS). Website, 2004–2009. Available online, `http://www.sics.se/tac`; last visited February 23, 2009.

[31] E. Sodomka, J. Collins, and M. L. Gini. Efficient Statistical Methods for Evaluating Trading Agent Performance. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence*, pages 770–775, Vancouver, British Columbia, Canada, 2007. AAAI Press.

[32] S. Y. Sohn, T. H. Moon, and K. J. Seok. Optimal Pricing for Mobile Manufacturers in Competitive Market Using Genetic Algorithm. *Expert Systems with Applications*, 36(2):3448–3453, 2009.

[33] C. S. Tang and R. Yin. Responsive Pricing under Supply Uncertainty. *European Journal of Operational Research*, 182(1):239–255, October 2007.

[34] D. Titterington, A. Smith, and U. Makov. *Statistical Analysis of Finite Mixture Distributions*. John Wiley and Sons, New York, 1985.

[35] University of Minnesota. Trading Agent Competition (UMN). Website, 2003–2009. Available online, `http://tac.cs.umn.edu/`; last visited February 23, 2009.

[36] W. E. Walsh, D. C. Parkes, T. Sandholm, and C. Boutilier. Computing Reserve Prices and Identifying the Value Distribution in Real-world Auctions with Market Disruptions. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*, pages 1499–1502, 2008.

[37] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Fransisco, 2nd edition, 2005.