Adaptive Tactical Pricing in Multi-Agent Supply Chain Markets using Economic Regimes

Alexander Hogenboom

Econometric Institute, Erasmus University Rotterdam, P.O. Box 1738, NL-3000 DR Rotterdam, The Netherlands, e-mail: hogenboom@ese.eur.nl

Wolfgang Ketter and Jan van Dalen

Rotterdam School of Management, Erasmus University Rotterdam, P.O. Box 1738, NL-3000 DR Rotterdam, The Netherlands, e-mail: wketter@rsm.nl, jdalen@rsm.nl

Uzay Kaymak

School of Industrial Engineering, Eindhoven University of Technology, P.O. Box 513, NL-5600 MB Eindhoven, The Netherlands, e-mail: u.kaymak@ieee.org

John Collins

Computer Science and Engineering, University of Minnesota, Minneapolis, Minnesota, United States, e-mail: jcollins@cs.umn.edu

Alok Gupta

Carlson School of Management, University of Minnesota, Minneapolis, Minnesota, United States, e-mail: agupta@csom.umn.edu

Adaptive Tactical Pricing in Multi-Agent Supply Chain Markets using Economic Regimes

(Authors' names blinded for peer review)

ABSTRACT

In today's complex and dynamic supply chain markets, information systems are essential for effective supply chain management. Complex decision making processes on strategic, tactical, and operational levels require substantial timely support in order to contribute to organizations' agility. Consequently, there is a need for sophisticated dynamic product pricing mechanisms that can adapt quickly to changing market conditions and competitors' strategies. We propose a two-layered machine learning approach to compute tactical pricing decisions in real-time. The first layer estimates prevailing economic conditions – economic regimes – identifying and predicting current and future market conditions. In the second layer, we train a neural network for each regime to estimate price distributions in real-time using available information. The neural networks compute offer acceptance probabilities from a tactical perspective to meet desired sales quotas. We validate our approach in the Trading Agent Competition for Supply Chain Management. When competing against the world's leading agents, the performance of our system significantly improves compared to using only economic regimes to predict prices. Profits increase significantly even though the prices and sales volume do not change significantly. Instead, tactical pricing results in a more efficient sales strategy by reducing both finished goods and components inventory costs.

INTRODUCTION

In today's supply chains for durable goods, decision processes on strategic, tactical, and operational levels are complicated by the complexity of logistics systems, which can encompass many interrelated entities. The relations between these entities tend to get more and more flexible and dynamic. Hence, effective Supply Chain Management (SCM), focused on flexible and dynamic relationships among supply chain entities, is vital to the competitiveness of manufacturers within this supply chain. Effective SCM enables manufacturers to respond to changing market demands in a timely and cost effective manner (Chopra & Meindl 2004) and can thus improve their agility. Real-time decision making technologies are becoming increasingly crucial due to the prevalence of e-business (Swaminathan & Tayur 2003).

Decision making technologies may be developed to support supply chain coordination activities (Aviv 2003). A major challenge is dynamic product pricing under uncertain demand for a single product (Araman & Caldentey 2009, Besbes & Zeevi 2009) or multiple products (Adida & Perakis 2010, van Ryzin & Vulcano 2004). When flexible and dynamic relationships between supply chain entities stimulate manufacturers of durable goods to compete for customer orders, optimal product prices must account for factors other than customer demand, such as competitors' strategies or market conditions (Dasgupta & Hashimoto 2004, Ketter et al. 2009, 2012, Saha et al. 2005, Sohn et al. 2009). This requires sophisticated dynamic product pricing, facilitated by intelligent automated decision support systems (Bichler et al. 2010).

We consider a dynamic supply-chain environment where customer demand arrives in the form of Request for Quotes (RFQs) containing product specifications, due date and reserve price. Manufacturers operate in an oligopolistic environment and competitively bid for customer orders. This market is characterized by high volatility of price and demand, as well as low market transparency. Therefore, manufacturers need to make strategic and tactical decisions about procurement, production, and pricing. Prior research (Ketter et al. 2009, 2012) has shown that economic regimes provide a way of intuitively and meaningfully characterizing and modeling market conditions, without a need for explicit modeling of individual aspects of the market. The original economic regime model quantifies market characteristics by means of a latent variable, which is assumed to be directly related to distributions of realized prices. This model successfully supports product pricing, as it enables manufacturers to estimate present and future distributions of realized prices and adjust their strategic pricing strategies accordingly.

However, some characteristics of the single-layered economic regime model impede tactical applicability of this model in highly competitive settings. First, the model as proposed by Ketter et al. (2009, 2012) is more concerned with predictions of changes in economic regimes over time to make strategic decisions, such as sales quotas over a longer planning horizon. A focus on predicting short-term prices to meet these sales quotas, could improve the supply-chain throughput resulting in better profits, as compared to only using strategic information provided by the regimes. Second, the pricing approach using one-layered economic regimes uses a single pricing model. In the present study, we use different pricing models for different regimes by training a separate neural network for each regime. The individual neural networks estimate the parameters for different log-logistic distributions associated with price distributions for each regime. We update each neural network in real-time by using available market information.

Our overall objective is to develop a dynamic product pricing model with the ability to adapt to real-time information and to investigate whether and how such a model contributes to profit maximization. With the help of the original economic regime model, we partition the data into separate regime-specific training sets, capturing the relation between real-time available information and price distributions under different market circumstances. This enables us to use the original economic regime model in order to adapt our price distribution estimations to identified market circumstances. The proposed model captures the pricing behavior of competitors at the most detailed level and individual requests for quotes (RFQs) by individual customers. Besides observed order prices, we consider real-time available information such as RFQ characteristics, e.g., requested quantities, lead times, and reservation prices, in this process.

We validate our approach in a highly competitive setting, the Trading Agent Competition for Supply Chain Management (TAC SCM) (Collins et al. 2005, 2010). TAC SCM has been organized since 2003 to promote high quality research into trading agents in supply chain environments. In TAC SCM, several manufacturers, modeled by means of autonomous software trading agents, compete in a component procurement market and in a sales market where assembled products are sold through reverse auctions in response to RFQs. Both markets are only partially observable in terms of auction outcomes and pricing.

The contribution of this paper is to propose a novel two-layered machine learning model for tactical sales management in supply chain markets for durable goods with limited information and high uncertainty. It achieves this by focusing on predicting economic conditions using economic regimes, estimating price distributions for each regime using fast neural networks, and using these price distributions to develop efficient sales strategies.

The paper is organized as follows. First, we discuss related work on dynamic product pricing. Our own product pricing approach is introduced in the subsequent two sections. Then, we evaluate the novel approach. In the last section, we conclude and provide directions for future research.

RELATED WORK

Previous research has revealed a number of important aspects of dynamic product pricing: the role of acceptance probabilities, the need for market responsive pricing policies, and the dependence of different market conditions. Acceptance probabilities indicate the likelihood that customers actually accept an offer and proceed to buy the product at the offered price. Estimated acceptance probabilities are useful for sellers in developing pricing policies to meet sales targets. Recent research supports the viability of this approach. Dasgupta & Hashimoto (2004), for instance, model product prices using a dynamic pricing algorithm based on an estimated distribution of buyer reservation prices. Reversing the cumulative reservation price distribution gives the proportion of customers willing to buy the product as a function of the offered price. This function can subsequently be used to determine the price that is expected to yield a specified sales quota. Likewise, Walsh et al. (2008) use estimated distributions of buyer's private values to model acceptance probabilities. Other ways of modeling acceptance probabilities include linear regression of acceptance rates on offer prices (Pardoe & Stone 2006b), off-line trained acceptance probability distributions, and modeling the decision function of the accepting entities, based on decision histories, e.g., using Chebychev polynomials (Saha et al. 2005).

In addition, dynamic product pricing should be responsive to market situations. Tang & Yin (2007), for instance, show for a situation with known demand and uncertain supply that 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 that discards realized supply. Consequently, modeling expected or observed supply-side behavior in the product pricing process contributes to profit maximization. Furthermore, Kovalchuk & Fasli (2008) and Li et al. (2004) consider current and future offers of competitors, defined as outside options. Sohn et al. (2009) analyze such outside options by quantifying a product's own price change together with the relative prices of competing products in a price elasticity. Optimal pricing policies are then selected based on scenario analysis for various outcomes of this price elasticity.

Further, dynamic pricing policies may benefit from information about systematically changing market conditions, known as economic regimes (Hathaway & Bezdek 1993, Mount et al. 2006). The ability of decision makers to correctly identify the current regime and predict the onset of new regimes is crucial in order to prevent over- or underreaction to market conditions (Massey & Wu 2005). Different economic regimes, such as situations of excess demand or excess supply, may involve different optimal pricing policies. Economic regimes can guide both tactical and strategic decision making. An example of tactical decisions would be product pricing, while the determination of product mix and production planning are examples of strategic sales decisions (Ketter et al. 2009, 2012).

Viable product pricing strategies in dynamic markets would ideally involve the estimation of the probability that customers accept an offered price, the use of expected or observed supplier behavior and competitors' pricing, and the identification of specific, possibly changing economic regimes. However, a dynamic product pricing approach that takes these findings jointly into account, and is thus able to adapt to ever-changing market circumstances such as those in highly competitive complex supply chains, is yet to be proposed. In our current endeavors, we aim to take a step in this direction.

Product Pricing based on Offer Price Distributions

Manufacturers of durable goods typically sell products in markets with differentiated customer demand. Knowledge about competitors' procurement, production, and pricing decisions is generally not available. Such knowledge can at most indirectly be inferred from unexpected increases or decreases of the volume or profitability of one's own sales. Customers may be expected to purchase the lowest priced product from any offer of equally preferred product specifications. Aggregate demand will vary over time with respect to volume and quality. Accurate predictions of the price at which customers are willing to accept offered products are essential to maintain sustainable market positions.

In this section, we model the probability that offers are accepted at a certain offer price. The model is based on the idea that past information about product sales and prices can be used to infer the distribution of offered prices in the market. Real-time application of this model requires flexible updating of the parameters involved, for which we propose Artificial Neural Networks (ANNs), as we explain below. A summary of the mathematical notation used in this paper can be found in the appendix (Table 4).

Modeling Acceptance Probabilities

We model customer offer acceptance probabilities based on the distribution of all offered prices. In this way, we directly consider the unobservable decision making processes of all traders and obtain a complete estimation of customer acceptance probabilities. By contrast, linear regression on recent offer prices, as proposed by Pardoe & Stone (2006b) may be less effective because regression typically only captures price movements in one direction. Furthermore, conflating actual offer price distributions with predicted price distributions should yield a more robust approach than using only predicted individual offer prices from a model (Kovalchuk & Fasli 2008), where point estimates are potentially non-occurring. Using the offer price distribution for a particular RFQ, we derive order price distributions as the distribution of the order statistic of the minimum offer price. The cumulative density of order prices summarizes the probabilities that order prices are at or below certain values. The reverse cumulative density approximates the probability for an agent to offer better deals than other competitors and thus to obtain orders.

Let us consider a single RFQ r on a particular day d for a particular good g. Let the prices p_{dgr} offered by traders in response to this RFQ be distributed in accordance with a marginal distribution $f\left(p_{dgr};\theta\right)$ and a cumulative distribution $F\left(p_{dgr};\theta\right)$, with $0 < p_{dgr} < u$, i.e., prices are non-negative and have a reservation price u as upper bound, and θ a vector of unknown parameters.

Assuming that customers only consider bids at or below their reservation price and always select the bids with the lowest offer price, order prices can be interpreted as realizations of the first order statistic for the minimum of a sample of offer prices. Considering n_{dgr} offer prices per RFQ, the cumulative density of the minimum offer price for that particular RFQ is the statistical order distribution (Kapadia et al. 2005):

$$F_1\left(p_{dgr};\theta\right) = 1 - \left(1 - F\left(p_{dgr};\theta\right)\right)^{n_{dgr}}, \quad 0 < p_{dgr} < u, \quad n_{dgr} > 0. \tag{1}$$

This cumulative distribution of order prices yields the fraction of order prices realized at or below offer price p_{dgr}. Hence, a trader placing an offer of p'_{dgr}, incurs an $F_1\left(p'_{dgr};\theta\right)$ risk of not acquiring the order. The probability $P\left(o|p_{dgr}\right)$ that a customer will place an order o with a trader at an offer price p_{dgr} can therefore be defined as:

$$P\left(o|p_{dgr}\right) = 1 - F_1\left(p_{dgr};\theta\right), \quad 0 < p_{dgr} < u. \tag{2}$$

The customer offer acceptance probability (2) holds for a single RFQ. Suppose that m_{dg} RFQs are presented to the market on day d for good g. Let 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 $0 < p_{dg} < u$ and θ a vector of unknown parameters. For such a distribution, the cumulative distribution of the order prices over all m_{dg} RFQs can be derived as:

$$F_1\left(p_{dg};\theta\right) = \left(1 - \left(1 - F\left(p_{dg};\theta\right)\right)^{n_{dg}}\right)^{m_{dg}}, \quad 0 < p_{dg} < u, \quad n_{dg}, m_{dg} > 0, \tag{3}$$

yielding an offer acceptance probability defined as:

$$P\left(o|p_{dg}\right)=1-\left(1-\left(1-F\left(p_{dg};\theta\right)\right)^{n_{dg}}\right)^{m_{dg}},\quad 0<p_{dg}<u,\quad n_{dg},m_{dg}>0. \tag{4}$$

Equation (4) can be used to estimate the share of received orders with respect to the total number of RFQs for product g on day d, generated by a price offered on these RFQs. When pricing products, one may have a specific target share or sales quota. Now, let q^*_{dg} be a ratio in the interval $[0,1]$, representing the sales quota for product g on day d, with m_{dg} associated RFQs, for each of which n_{dg} prices are offered. The associated offer price p^*_{dg} is obtained by taking the inverse probability function at q^*_{dg}. Solving the equation to p^*_{dg} yields the optimal offer price p^*_{dg} associated with the desired quota. Following this approach, tactical pricing decisions for day d are directly linked to price distributions for day d through modeled acceptance probabilities.

Utilization of the price distributions in the bidding process requires insight into θ and n_{dg} in (4). When all data is available, m_{dg} and n_{dg} can be determined by a counting process and θ can be estimated by maximum likelihood as follows. When all prices in the sample of offers p_{dg} for all m_{dg} RFQs issued for product g on day d are 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 equals

$$f\left(p_{dg};\theta\right)=\prod_{r=1}^{m_{dg}}\prod_{i=1}^{n_{dg}}f\left(p_{dgri};\theta\right),\quad 0<p_{dgri}<u, \tag{5}$$

where p_{dgri} is the ith of n_{dg} prices offered on day d for RFQ r for product g. Parameters θ can be estimated by minimizing their negative log-likelihood function for observed offer prices p_{dg}, e.g., using the Newton-Raphson method (Coleman & Li 1996):

$$L\left(\theta;p_{dg}\right)=-\sum_{r=1}^{m_{dg}}\sum_{i=1}^{n_{dg}}\ln\left(f\left(p_{dgri};\theta\right)\right),\quad 0<p_{dgri}<u. \tag{6}$$

The optimization of (6) is straightforward when all relevant price information is available. In real-time, however, offer prices from competitors may not be available for estimating the joint distribution of offer prices (5). One solution to dealing with this problem might be to use a general probability density estimator for finding a mapping between observable quantities and the target distributions. Examples are kernel density estimation (Jones 1989), probabilistic fuzzy systems (van den Berg et al. 2004) and artificial neural networks (Marchette & Priebe 1990). However, identifying this mapping may be hampered by the large number of parameters to estimate. Furthermore, the convenient multiplicative form of (5) would not hold in general. A more viable alternative is to select a general parametric distribution function and to use a general function approximator to determine the parameter values of the target distribution from the information available real-time. In the next section, we further elaborate on this idea.

Real-Time Parameter Estimation

Real-time estimation of the distribution parameters θ and n_{dg} requires the specification of a relation between these parameters and a vector of real-time available information, $\vec{x}$, possibly including information on RFQs or one's own sales. Approximations of these relations, $\hat{h}_{\theta}\left(\vec{x}\right)$ and $\hat{h}_{n_{dg}}\left(\vec{x}\right)$, can be found, for instance, by linear regression (Pardoe & Stone 2006b). However, the functional specification of the relation between the parameters and real-time available information may not always be known a priori. We therefore model these relations with artificial neural networks (ANNs), which have been proven to be useful for economic forecasts in various domains (Kovalchuk & Fasli 2008).

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 samples in order to facilitate regression or classification (Mitchell 1997). The model consists of interconnecting artificial neurons (nodes), ordered into an input layer, hidden layers, and an output layer. Due to its ability to capture complex nonlinear relations, an ANN model is used for real-time parameter estimation with real-time available inputs. Our use of ANNs brings the attractive feature of fast evaluation of these functions, which is essential for real-time product pricing. Other advantages include robustness to noise in the training data (Mitchell 1997) and support for adaptivity by adjusting the weights of each node's inputs on-the-fly using newly obtained samples.

A drawback of standard feed-forward back-propagation ANNs is the time needed for design and training. An alternative could be a Probabilistic Neural Network (PNN) or a Radial Basis

Function Network (RBFN). These two types of ANNs are essentially quite similar, with their main difference lying in the fact that PNNs model each data point in the training set with a neuron, whereas RBFNs typically need a number of neurons that is significantly smaller than the number of data points in the training set. This gives RBFNs an attractive advantage over PNNs for our purpose, with our data typically consisting of a vast amount of data points.

In this light, we use an RBFN, a specific type of ANN which can be designed and trained in a fraction of the time it takes to design and train standard feed-forward back-propagation ANNs (Mitchell 1997). An RBFN is a two-layer ANN consisting of a hidden layer and an output layer. The activation function in each hidden unit h is a local function $K_h\left(d\left(\vec{x}_h, \vec{x}\right)\right)$, where $\vec{x}$ is a vector of features, $\vec{x}_h$ is called the center of the local function, and $d\left(\vec{x}_h, \vec{x}\right)$ is the (Euclidean) distance between $\vec{x}_h$ and $\vec{x}$. $K_h(z)$ is a so-called kernel function whose value is at a maximum when $z = 0$, and approaches zero for large values of z. The local functions in the hidden layer are typically Gaussians (Mitchell 1997), centered at $\vec{x}_h$ with variance σ_h^2. The number of Gaussians H is subject to optimization and their centers can be determined by clustering the data in various ways, for example using the k-means algorithm (MacQueen 1967). The variance σ_h^2 of each Gaussian may for example be estimated as the largest squared distance between the centers. 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 (Mitchell 1997):

$$\hat{h}\left(\vec{x}\right) = w_0 + \sum_{h=1}^{H} w_h K_h\left(d\left(x_h, \vec{x}\right)\right), \tag{7}$$

$$K_h\left(d\left(x_h, \vec{x}\right)\right) = e^{\frac{1}{2\sigma_h^2} d^2\left(x_h, \vec{x}\right)}. \tag{8}$$

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 functions around their centers. It has a relatively easy design and training stage, it is robust, has the possibility to introduce adaptivity, and has fast real-time evaluation capabilities. By using an RBFN to map real-time available information to distribution parameter values, our product pricing model is responsive to observable information in a predefined way. However, this mapping as such is essentially static and should be updated at runtime in order to adapt to changing market conditions and market responses. Therefore, we introduce adaptivity to the model by using economic regimes (Ketter et al. 2009, 2012).

ADAPTIVE PRODUCT PRICING USING ECONOMIC REGIMES

Adaptive, real-time product pricing requires dynamic modeling of the relations between available data and price distributions based on market conditions. Additionally, we incorporate market feedback by means of error terms that reflect the market's response to our pricing decisions.

In real-world scenarios, market conditions offer only limited visibility. Previous research (Ketter et al. 2009, 2012) identified distinct market conditions through economic principles and machine learning techniques. Historical data were used to compute price distributions that, in turn, enabled the estimation of sales prices and the identification of economic regimes (e.g., scarcity, a balanced situation, or oversupply). Here, we add real-time adaptivity using observable market information.

Realized order prices and order probabilities tend to vary with the distribution of economic regimes (Ketter et al. 2009, 2012), which affects the relations between available data and price distributions. Hence, we differentiate product pricing strategies with respect to economic regimes. Suppose that market conditions can be meaningfully summarized by means of M regimes. None of these regimes can be known to occur with certainty. Instead we assume that each day d, each of these regimes, R_{dgk}, occurs with probability $P\left(R_{dgk}\right)$ for product g, as calculated by applying the regime model proposed by Ketter et al. (2009, 2012). The regime with the highest probability of occurrence is labeled as dominant regime on day d. For each dominant regime $k = 1, \ldots, M$, price distribution parameters θ_k and n_{dgk} (and m_{dg}, in case of predictions) for product g on day d are estimated using RBFNs. Based on these M estimated distributions, acceptance probabilities $P\left(o_k|p_{dgk}\right)$ are derived, which are subsequently weighted with the associated regime probabilities $P\left(R_{dgk}\right)$ for regime R_{dgk}, to have the resulting acceptance probability estimations reflect the market conditions.

Updating the weights in the RBFNs based on real-time information is often not feasible due to limited visibility of, for instance, the realized sales price distributions. Instead, daily estimated acceptance probabilities may be adjusted directly by multiplying the probabilities with of a correction factor ϵ defined as the ratio of actually received orders and predicted orders, as proposed by Pardoe & Stone (2006a) and Ketter et al. (2012). This ratio feeds market responses back to the model in real-time. Using performance information up until day $d-1$, the error term can be smoothed, yielding $\widetilde{\epsilon}_{(d-1)gk}$ for each regime k, to prevent over- or undercompensation.

Obviously, acceptance probabilities $P\left(o_k|p_{dgk}\right)$ range from 0 to 1. But if we straightforwardly adopt the suggested correction, adjusted acceptance probabilities $P\left(o_k|p_{dgk}\right)'$ will be in the range $\left[0, \widetilde{\epsilon}_{(d-1)gk}\right]$. This implies that for $q^*_{dg} \geq \widetilde{\epsilon}_{(d-1)gk}$ no suitable price can be found, which is clearly undesirable in case $\widetilde{\epsilon}_{(d-1)gk} < 1$. Therefore, we introduce the feedback correction as an exponent of the offer acceptance probability as follows:

$$\begin{aligned} P\left(o_k|p_{dgk}\right)' &= P\left(o_k|p_{dgk}\right)^{\widetilde{\epsilon}_{(d-1)gk}}, \quad 0 < p_{dgk} < u, \quad \widetilde{\epsilon}_{(d-1)gk} > 0, \\ &= \left(1 - \left(1 - \left(1 - F\left(p_{dgk}; \theta_k\right)\right)^{n_{dgk}}\right)^{m_{dg}}\right)^{\widetilde{\epsilon}_{(d-1)gk}}, \\ &\quad 0 < p_{dgk} < u, \quad n_{dgk}, m_{dg}, \widetilde{\epsilon}_{(d-1)gk} > 0, \end{aligned} \tag{9}$$

The adjusted acceptance probabilities range from 0 to 1 for $0 < p_{dgk} < u$ after correction for the smoothed correction term $\widetilde{\epsilon}_{(d-1)gk}$. For each regime k, expression (9) is used to determine the optimal corrected offer price $p^{*'}_{dg}$ corresponding with the desired sales quota q^*_{dg} for product g on day d by solving the equation $P\left(o_k|p^{*'}_{dgk}\right)' = q^*_{dg}$ to $p^{*'}_{dgk}$. An overall corrected offer price $p^{*'}_{dg}$ can then be obtained by averaging the regime-based prices using the associated regime probabilities $P\left(R_{dgk}\right)$ as weights.

The correction term considered in (9) 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 offer price $p^{*'}_{(d-1)g}$ are corrected by the unsmoothed correction terms $\epsilon_{(d-1)gk}$ to the proportion of actually received number of orders $q_{(d-1)g}$. The correction terms can subsequently be smoothed. Hence, offer price and customer response should be related as follows:

$$q_{(d-1)g} = P\left(o_k|p^{*'}_{(d-1)g}\right)^{\epsilon_{(d-1)gk}}, \quad 0 < q_{(d-1)g} < 1, \quad 0 < p^{*'}_{(d-1)g} < u, \quad \epsilon_{(d-1)gk} > 0. \tag{10}$$

In an attempt to track the trend in the correction terms and thus to provide an accurate estimate of the error of the model on day d, given information up until day $d-1$, the correction terms can be smoothed using exponential smoothing. Here, the smoothing factor β is taken to be proportional to the associated regime probabilities in order to attribute errors only to the models responsible for these errors. Smoothing is done by linearly combining the latest correction term (11) and the previous smoothed correction:

$$\epsilon_{(d-1)gk} = \frac{\ln\left(q_{(d-1)g}\right)}{\ln\left(P\left(o_k | p^{*'}_{(d-1)g}\right)\right)}, \quad 0 < q_{(d-1)g} < 1, \quad 0 < P\left(o_k | p^{*'}_{(d-1)g}\right) < 1, \tag{11}$$

$$\widetilde{\epsilon}_{(d-1)gk} = \beta P\left(R_{(d-1)gk}\right) \epsilon_{(d-1)gk} + \left(1 - \beta P\left(R_{(d-1)gk}\right)\right) \widetilde{\epsilon}_{(d-2)gk}. \tag{12}$$

Summing up, the model assumes some distribution to be underlying offer prices, the parameters of which can be estimated real-time using RBFNs. The relation between the available data and the parameters is dynamically modeled using economic regimes (characterizing market conditions) and error terms (accounting for customer feedback), thus enabling adaptive product pricing. For now, predicted effects of the resulting dynamic pricing behavior on market conditions are not directly taken into account when pricing products using the proposed model, yet the model can always adapt to newly instigated market conditions immediately after the fact. The proposed framework can now be evaluated in a highly competitive setting.

Implementation in a Software Agent

The model as proposed in the previous sections assumes a general offer price distribution, which is yet to be specified. We propose to use 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, according to the Kolmogorov-Smirnov test with a significance level of 0.05, this distribution appears to adequately describe the data in TAC SCM in over 60% of the analyzed price samples from historical data, i.e., TAC SCM 2007 Semi-Finals and Finals (9323–9327tac5 and 7308–7312tac3) (Swedish Institute for Computer Science 2004–2012) and TAC SCM 2008 Semi-Finals and Finals (763–768tac02 and 794–799tac01) (University of Minnesota 2003–2012) game data. The log-logistic distribution $f(p; \alpha, \gamma)$ and its cumulative form $F(p; \alpha, \gamma)$ (Mood et al. 1974) are truncated such that the distribution is defined on the domain $0 < p < u$ and parameterized such that α represents the median and γ quantifies the distribution tightness:

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

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

When $F(p_{dgk};\theta_k)$ in (9) is substituted for (14), the corrected price $p^{*'}_{dg}$ expected to yield a required sales quota (i.e., acceptance probability (Ketter et al. 2012)) q^*_{dg} can be obtained as

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

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

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

with $u = 1.25$ (as normalized prices in TAC SCM are never above 1.25 of nominal cost) and the number of considered regimes $M = 5$, as extensively discussed by Ketter et al. (2009, 2012). In TAC SCM, the number of offers per RFQ n_{dg} is approximated by the mean number of offers per RFQ for product g on day d, which is bounded by the number of participating agents.

Dynamic product pricing in TAC SCM can now be done using (15) through (17). Due to limited visibility of market conditions, the α_k, γ_k, and n_{dgk} parameters for product g on day d for dominant regime k are to be estimated using RBFNs. Further adaptivity of the framework can be realized by utilizing (11) and (12) to account for customer feedback. We dynamically scale the learning rate β to range between 0.15 and 0.85 (as determined using a hill-climbing procedure), depending on the distance of the model's median with respect to the price-following median of observed minimum and maximum prices.

Implementation in the MinneTAC Agent

The sales decisions made by the MinneTAC agent originate from price predictions based on market observations, and are characterized separately for each of the three market segments.

These economic regimes have been characterized as extreme scarcity, scarcity, a balanced situation, oversupply, and extreme oversupply (Ketter et al. 2009, 2012). On day d, the (dominant) regime for good g is identified as the regime having the highest likelihood of occurrence, given the estimated normalized mean price for that day. We estimate the normalized mid-range price, $\widetilde{\mathrm{np}}_{dg}$, as the average of the double exponentially smoothed normalized minimum and maximum price observed in the market. The normalized price np_{dg} for good g on day d is the ratio of actual price to nominal production cost, which is the sum of the nominal costs of the parts required to build a product. Normalized prices rarely fall below half of nominal cost, since this is the lowest cost available from suppliers, and never rise above 1.25 of nominal cost, the upper bound on customer reservation prices.

A price density function has been modeled on historical normalized order price data by using a Gaussian Mixture Model (GMM) (Titterington et al. 1985) with 25 Gaussians, which number reflects a balance between prediction accuracy and computational overhead. Clustering these price distributions over time periods (by using the k-means algorithm) yields distinguishable statistical patterns that can be interpreted as economic regimes (Ketter et al. 2009, 2012). In TAC SCM, price information is available up until the preceding day. Hence, the MinneTAC agent estimates the mean price of the current day d using an exponentially smoothed prediction of $\widetilde{\mathrm{np}}_{dg}$ and then returns the regime probabilities for day d. When this estimate is supplied to the model, each individual Gaussian in the model is activated to its own distinct extent, thus generating an expected price distribution. Subsequently normalizing all clusters' price densities enables determination of regime probabilities. Price distributions and regime probabilities for future days are determined by using Markov prediction and Markov correction-prediction processes for short-term and long-term decision making, respectively (Ketter et al. 2012). After determining current and future regime probabilities, products are priced using offer acceptance probabilities, such that sales quotas are fulfilled.

In our benchmark configuration, the sales process is fully driven by the economic regime (ER) model (Ketter et al. 2009, 2012). Median product prices and future price trends are derived from order price distributions estimated by this model. The trends and medians, along with cost estimates and resource constraints, are inputs to a linear program that computes sales quotas to maximize profits over a medium-term time horizon. Given these daily sales quotas, optimal offer prices are computed as the price that will sell a quota against a probability that is specified by an order acceptance probability curve. This curve is approximated using the order price distributions estimated by the ER model. In order to compensate for uncertainty in generated predictions, interval randomization is applied to offer prices, which adds a slight variability to these prices. The ER model underlying the sales process in this benchmark configuration does not adapt to real-time available information or customer feedback.

The price distributions estimated by the ER model are updated on-line based on observed minimum and maximum prices, but they do not account for other factors like RFQ characteristics and lack full adaptivity. Previous research has shown the effectiveness of economic regimes on predominantly a strategic level of decision making (Ketter et al. 2009, 2012), yet the lack of adaptivity of the model renders it less useful for decision making on a tactical level, such as product pricing. In an attempt to improve the product pricing process by combining regime information with real-time available data, we replace the ER model with a Dynamic Economic Regime (DynER) model. We implement the proposed adaptivity for daily product pricing decisions on a tactical level. The algorithm (Algorithm 1) involves parameter estimation using RBFNs and subsequent product pricing based on (15) through (17). A correction term is used, following (11) through (12).

[Figure 1 about here.]

In order to introduce adaptivity into the ER model, we built an adapter for the outputs of the ER model (see Fig. 1). The adapter uses RBFNs to transform order price distributions estimated by the ER model into parameterized order price distributions per dominant regime and assigns weights to these distributions, equal to their associated regime probabilities. These adaptations are based on RFQ characteristics and predicted demand. The adapted distributions can then be used for product pricing. A parameterized acceptance probability function can be derived per dominant regime. Given a quota for a product (specified by the agent's allocation component), the product pricing component uses the adapter to compute the price that is expected to yield this quota per dominant regime and weights the suggested prices with their associated regime probabilities. Then, offers are made on all selected RFQs for the considered product, in an interval around the optimized price. The market responses to these offers are fed back to the adapter.

[Algorithm 1 about here.]

Radial Basis Function Network Training

In order for the DynER approach to work in the MinneTAC agent for TAC SCM, we need to train RBFNs for real-time estimation of unobservable price distribution parameters. To this end, we utilize TAC SCM 2007 Semi-Finals and Finals (9321–9328tac5 and 7306–7313tac3) (Swedish Institute for Computer Science 2004–2012) and TAC SCM 2008 Semi-Finals and Finals (761–769tac02 and 792–800tac01) (University of Minnesota 2003–2012) game data. The first two games and the last game per server form the test set, the rest forms the training set. For each dominant regime k, an RBFN is trained to estimate α_k, γ_k, and n_{dgk} for product g on day d. Training and test datasets are split into datasets per dominant regime, identified by the current ER model.

We 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. Hence, these regime-based price distributions are used as RBFN inputs. Other RBFN inputs include the product type, the day, the RFQ characteristics of the product for that day, and observable historical prices. We characterize the regime-based price distributions by their 10th, 50th, and 90th percentiles. The RFQ characteristics of the product for that day include the number of RFQs, the mean and standard deviation of their reservation prices, the mean and standard deviation of the requested quantities, and the mean and standard deviation of the lead times. The observable historical prices are characterized by the minimum, maximum, and mid-range prices of the preceding day, both nominal and exponentially smoothed.

When adapting regime-based price distributions with our RBFNs, we use data on product type and day because of the heterogeneity of products and the volatility of their associated prices in TAC SCM. RFQ characteristics are incorporated, since competitors' pricing strategies are likely to depend on these characteristics. Reservation prices, for example, give an indication of the maximum price manufacturers will bid. Also, the number of simultaneous similar bidding processes affects the associated revenues due to their (partial) substitutivity (Walsh et al. 2008). Observable (historical) prices can also provide more insight into competitors' pricing strategies (Ketter et al. 2009, 2012, Kovalchuk & Fasli 2008).

Using our historical TAC SCM data, target parameter values can be determined by counting (for n_{dgk}) and by fitting distributions using (6) and (13) (for α_k and γ_k). A typical training dataset thus generated contains over 15,000 observations, a typical test dataset over 8,000. The performance of RBFNs trained on the training set can be evaluated on the test set, which is sufficiently large and representative (Mitchell 1997). For the resulting optimal values for γ_k, the increment in γ_k needed to tighten the distribution increases as the distribution gets tighter. As the required accuracy decreases for an increasing γ_k, the networks are trained to predict the natural logarithm of γ_k.

For each regime-specific RBFN, the architecture is optimized keeping the numbers of inputs and outputs fixed. The number of hidden nodes is determined using a hill-climbing procedure, that minimizes the root mean squared deviation (RMSD) of the outputs on the test set. The RMSD equals

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

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

In order to train the RBFNs, data are normalized to zero mean and unit variance and subsequently clustered using the k-means algorithm, with the number of clusters equal to the number of hidden nodes (Nabney 2002). Using the expectation maximization algorithm (Dempster et al. 1977), Gaussians are fit by optimizing the centers (initialized as the cluster centers). Gaussian widths equal the maximum squared Euclidian distance between the function centers. Finally, internal weights of the RBFNs are optimized, such that the sum of squared errors is minimized.

The results in Table 1 indicate that in general, parameter estimation based on real-time available data leaves some room for improvement, albeit to a lesser extent for α_k. Nevertheless, the proposed feedback mechanism should (partially) compensate for this.

[Table 1 about here.]

Performance Evaluation

The performance of the DynER system is assessed in real TAC SCM games, as specified by Collins et al. (2005). The random settings of TAC SCM, like in customer demand, introduce a problem when using this testing environment to compare two different approaches. This randomness is partly controlled by using the TAC SCM server (Sodomka et al. 2007), in which random seeds used for generating market conditions can be re-used. However, the stochastic behavior of participating agents cannot be controlled, as this is not generated on the server. Research shows that under equal market conditions, such stochastic behavior does not have a significant impact on the agent profit levels and that profitability differences between agents can be reliably detected in approximately 40 games (Sodomka et al. 2007).

We therefore evaluate the performance of the DynER model in 40-game experiments on a controlled server. These 40 games are used for validation purposes and are not part of the training and test sets for our model. Each experiment set consists of a paired evaluation of the performance of the benchmark (Ketter et al. 2009, 2012) and the DynER system under equal market definitions. The configurations can be tested in a highly competitive setting, in which MinneTAC competes against world's leading TAC SCM trading agents: TacTex (Pardoe & Stone 2006b) (TT), DeepMaize (Kiekintveld et al. 2006, 2009) (DM), PhantAgent (Stan et al. 2006) (PA), the CrocodileAgent (Podobnik et al. 2006) (CA), and Mertacor (Chatzidimitriou et al. 2008) (MC). TT predicts demand using a Bayesian approach and estimates offer acceptance probabilities using linear regression. It adapts these offer acceptance probability estimations to its opponents' behavior. DM uses a gradient descent algorithm to find offer prices that optimize the expected value of the resulting orders. PA and CA use simple heuristics for determining what to sell for what price. MC predicts the winning bid per RFQ using a regression model, complemented with a price-following correction mechanism. The acceptance probabilities are subsequently estimated using a k-nearest neighbors algorithm (MacQueen 1967).

Performance differences are assessed with respect to their statistical relevance using a paired, two-sided Wilcoxon signed-rank test (Gibbons 1986, Hollander & Wolfe 2000), which evaluates the hypothesis that the differences between paired observations are symmetrically distributed around a median equal to zero. If this null hypothesis is rejected, at a significance level of 0.05, the compared samples are labeled significantly different.

Tables 2 and 3 illustrate the observed performance of the benchmark and the DynER configurations of MinneTAC (MT), with respect to the performance of MinneTAC's competitors. The number of obtained orders obtained by MinneTAC decreases by about 3% when introducing adaptivity into the product pricing process, while MinneTAC's profits significantly increase by 478%, thus improving its competitive position. The mean difference between the bank account balance of MinneTAC and its competitors significantly improves by 136% (in absolute terms) from -1.270 million (standard deviation 2.742 million) to 0.452 million (standard deviation 1.803 million). Additionally, the difference between MinneTAC's score and the top score in a game significantly improves by 27% from -6.879 million (standard deviation 2.987 million) to -5.044 million (standard deviation 2.365 million). The decreased standard deviations suggest that the performance of MinneTAC is more stable over all experiments when utilizing the DynER approach instead of using the single-layered economic regime approach. As the DynER model has been designed to facilitate tactical product pricing, an intuitive explanation for MinneTAC's improved performance may be improvements in better meeting target sales quotas. Later we will discuss this in more detail.

[Table 2 about here.]

[Table 3 about here.]

Improvements in MinneTAC's bidding behavior can be observed in its bid efficiency. Bid efficiency is the revenue of a bid on an arbitrary RFQ, expressed in terms of the maximum revenue that could have been obtained from that RFQ, given its reservation price and competitor's offers (Jordan et al. 2007). The optimal bid efficiency is equal to 1. A bid efficiency less than 1 implies underpricing, whereas a bid efficiency greater than 1 implies overpricing. For the orders that MinneTAC does obtain, the mean bid efficiency increases slightly, yet significantly by 0.21% from 0.954 (standard deviation 0.005) to 0.956 (standard deviation 0.007) over all experiments. This implies that the prices of acquired orders tend to be closer to second-lowest prices yielding a reduced margin between customers' reservation prices and realized order prices. In cases of overpricing, the mean bid efficiency significantly improves by 1.41% from 1.058 (standard deviation 0.028) to 1.043 (standard deviation 0.008), indicating that MinneTAC's lost bids are closer to the winning bid. The decreased standard deviation indicates a more stable performance of our acceptance probability estimation process in the case of non-winning offers. These observations suggest that DynER's acceptance probability estimations are better than those in the benchmark configuration. The order price distribution estimations (underlying acceptance probability estimations) in Fig. 2 support our claim. Clearly, in the typical scenario depicted in Fig. 2, the distribution of realized offer prices estimated by using DynER, given any of five considered dominant regimes, is closer to the real distribution than those estimated by the one-layered benchmark.

[Figure 2 about here.]

An intuitive explanation for the increase in profits could be that products are typically sold at higher prices when tactical sales pricing is introduced or the agent is able to sell significantly more products. However, neither of these seem to be the case with DynER. DynER results in a 3% decrease in the number of orders obtained and the average price of products sold decreases by 2%. Therefore, the increase in profits can entirely be attributed to reduction in costs. The cost has multiple components, namely production cost as well as components and finished good inventory costs. Further analysis of our experimental data shows that approximately 65% of the observed increase in profits can be attributed to total material costs, which have significantly decreased by about 5%. This may be the result of a minor decrease in the need for components due to the slightly decreased number of obtained orders. The remaining 35% of the observed increase in profits can be attributed to a significant decrease of both storage costs and interest by 13%. Therefore, the proposed adaptive tactical product pricing approach results in higher profits by bidding better matched prices to achieve target sales quota to optimize its inventory levels, thus decreasing storage costs as well as interest paid.

Interestingly, even though the procurement strategy is identical in both configurations, we see significant differences in component inventory costs. Therefore, it is useful to examine this phenomenon in more detail. Total inventory costs are proportional to the number of stocked items and the time these items are in stock. Hence, an intuitive explanation for the significant drop in inventory costs when introducing adaptivity would be a drop in inventory levels. Indeed, inventory levels are significantly reduced by approximately 7% on average. This suggests that less inventory is built-up due to more efficient pricing. Figure 3(a) shows the finished goods inventory levels, while the average inventory levels are not significantly different in the two configurations, in the single-layered model inventories are typically held for a much longer time period resulting in higher inventory costs. As Figure 3(b) indicates, the component inventory levels are significantly lower (by 25%) with DynER. This reflects selling the right product at the right time to use the production facility more efficiently. DynER's consistent sales strategy is clearly reflected in Figure 3 as opposed to single-layered model where sales are concentrated at the beginning and the end of the sales cycle.

[Figure 3 about here.]

[Figure 4 about here.]

More detailed examination of MinneTAC's inventory levels reveals that, on average, the mean number of consecutive days that MinneTAC's inventory levels remain unchanged decreases significantly by 16%. For MinneTAC's component inventories, the decrease is approximately 6%, whereas for finished goods inventories, the decrease is about 22%. The mean number of consecutive days that MinneTAC's inventories do not decrease due to a lack of successful sales attempts, significantly decreases by 15% on average, where finished goods inventories account for an approximate 18% decrease and component inventories account for about 12% decrease in time in between between inventory decreases. Moreover, the mean number of consecutive days without an increase of inventory levels decreases by about 7%. This significant decrease is about 10% for finished goods inventories and an insignificant 2% decrease for component inventory levels.

To summarize, the introduction of tactical consideration in product pricing causes inventories to move faster. DynER causes MinneTAC to sell more often, resulting in a faster decrease of finished goods inventories and, to replenish these finished goods inventories, the corresponding constituent components are used in production thereby decreasing component inventories. The frequency of replenishment in component inventories does not increase significantly, as the employed procurement strategy – which already provides for very frequent procurement – is not changed in this experimental set-up. The standard deviation of MinneTAC's sales quantities may be a more direct indicator of the change in sales behavior. This standard deviation on average decreases by 36% when DynER is used, which suggests that MinneTAC's sales performance is more stable over time.

The tactical change of the MinneTAC agent introduced by the DynER approach has its effects on MinneTAC's market position as well as on the supply chain as a whole, as illustrated in Fig. 4. First, the number of days when all customer demand is met by the participating agents, on average, increases by 6%. While this is beneficial to customers, it introduces higher competition for participating agents for a longer duration. Interestingly, it is in these highly competitive periods of time, when there is no excess demand, that our tactically focused strategy significantly improves MinneTAC's market share (by 66%). In contrast, MinneTAC's market share in less competitive phases of the game is hardly affected. The dynamic tactical product pricing mechanism allows us to better adjust our prices during well balanced regimes, where demand can be satisfied with existing supply. This enables us to keep inventories balanced and to realize a lean, cost-effective sales process.

CONCLUSIONS AND FUTURE WORK

When product pricing strategies are linked to price distribution estimates and real-time information is taken into account, the relation between data and price distribution parameters can be dynamically modeled using economic regimes (characterizing market conditions) and correction terms (accounting for market feedback). In a constrained environment like TAC SCM, economic regime estimates contribute to profit maximization when they are used to differentiate product pricing strategies. Test results indicate that in highly competitive settings, this novel innovative tactical approach, comprising of a two-layered machine technique significantly improves the performance over a single-layered strategic product pricing approach.

The tactical adaptive product pricing mechanism performs better in estimating the high and low ends of price distributions (bid efficiency increases), thus better tracking the price distributions in the price range over a short period of time. Conversely, these distributions do not capture the full details of the distribution, not taking advantage of long-term market trends. Therefore, we do not see an increase in either sales price nor sales volume. However, profits increase significantly due to better alignment of sales quotas and finished goods inventories, which overall reduces inventory holding costs.

While the performance of our model significantly improves compared to using only economic regimes to predict prices, some aspects warrant future research. We are currently working on an ensemble prediction method, where we use the one-layered regime model for strategic decision-making, and the two-layered regime model for tactical decision-making. Since the methods might work well for different planning horizons, we plan to perform ensemble prediction for all days over the planning horizon. This should result in an overall better sales performance.

REFERENCES

Adida, E., & Perakis, G. (2010). Dynamic pricing and inventory control: uncertainty and competition. *Operations Research*, *58*(2), 289–302.

Araman, V., & Caldentey, R. (2009). Dynamic pricing for nonperishable products with demand learning. *Operations Research*, *57*(5), 1169–1188.

Aviv, Y. (2003). A time-series framework for supply-chain inventory management. *Operations Research*, *51*(2), 210–227.

Besbes, O., & Zeevi, A. (2009). Dynamic pricing without knowing the demand function: risk bounds and near-optimal algorithms. *Operations Research*, *57*(6), 1407–1420.

Bichler, M., Gupta, A., & Ketter, W. (2010). Designing smart markets. *Information Systems Research*, *21*(4), 688–699.

Chatzidimitriou, K., Symeonidis, A., Kontogounis, I., & Mitkas, P. (2008). Agent Mertacor: a robust design for dealing with uncertainty and variation in SCM environments. *Expert Systems with Applications*, *35*(3), 591–603.

Chopra, S., & Meindl, P. (2004). *Supply chain management*. New Jersey, USA: Pearson Prentice Hall.

Coleman, T., & Li, Y. (1996). An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, *6*(2), 418–445.

Collins, J., Arunachalam, R., Sadeh, N., Eriksson, J., Finne, N., & Janson, S. (2005). The supply chain management game for the 2006 trading agent competition. Tech. Rep. CMU-ISRI-05-132, Carnegie Mellon University, Pittsburgh, PA.

Collins, J., Ketter, W., & Sadeh, N. (2010). Pushing the limits of rational agents: the trading agent competition for supply chain management. *AI Magazine*, 63–80.

Dasgupta, P., & Hashimoto, Y. (2004). 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)*. New York, MY, USA: IEEE Computer Society, 277–284.

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, *39*(1), 1–38.

Gibbons, J. (1986). Nonparametric statistical inference. *Technometrics*, *28*(3), 275.

Hathaway, R., & Bezdek, J. (1993). Switching regression models and fuzzy clustering. *IEEE Transactions on Fuzzy Systems*, *1*(3), 195–204.

Hollander, M., & Wolfe, D. (2000). Nonparametric statistical methods. *Journal of the American Statistical Association*, *95*(449), 333.

Jones, M. (1989). Discretized and interpolated kernel density estimates. *Journal of the American Statistical Association*, *84*(407), 733–741.

Jordan, P., Kiekintveld, C., Miller, J., & Wellman, M. (2007). Market efficiency, sales competition, and the bullwhip effect in the TAC SCM tournaments. *Lecture Notes in Computer Science*, *4452*, 62–74.

Kapadia, A., Chan, W., & Moye, L. (2005). *Mathematical statistics with applications*. CRC Press.

Ketter, W., Collins, J., Gini, M., Gupta, A., & Schrater, P. (2009). Detecting and forecasting economic regimes in multi-agent automated exchanges. *Decision Support Systems*, *47*(4), 307–318.

Ketter, W., Collins, J., Gini, M., Gupta, A., & Schrater, P. (2012). Real-time tactical and strategic sales management for intelligent agents guided by economic regimes. *Information Systems Research*, (Online First, DOI: 10.1287/isre.1110.0415).

Kiekintveld, C., Miller, J., Jordan, P., Callender, L., & Wellman, M. (2009). Forecasting market prices in a supply chain game. *Electronic Commerce Research and Applications*, *8*(2), 63–77.

Kiekintveld, C., Miller, J., Jordan, P., & Wellman, M. (2006). Controlling a supply chain agent using value-based decomposition. In *Proceedings of the 7th ACM conference on Electronic commerce (EC'06)*. New York, NY, USA: ACM, 208–217.

Kovalchuk, Y., & Fasli, M. (2008). Adaptive strategies for predicting bidding prices in supply chain management. In *Proceedings of the Tenth International Conference on Electronic Commerce (ICEC 2008)*. Innsbruck, Austria.

Li, C., Giampapa, J., & Sycara, K. (2004). Bilateral negotiation decisions with uncertain dynamic outside options. In *First IEEE International Workshop on Electronic Contracting*. Washington, DC, USA: IEEE Computer Society, 54–61.

MacQueen, J. (1967). Some methods of classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. 281–297.

Marchette, D., & Priebe, C. (1990). The adaptive kernel neural network. *Mathematical and Computer Modelling*, *14*, 328–333.

Massey, C., & Wu, G. (2005). Detecting regime shifts: the causes of under- and overreaction. *Management Science*, *51*(6), 932–947.

Mitchell, T. (1997). *Machine learning*. McGraw-Hill Series in Computer Science. McGraw-Hill.

Mood, A., Graybill, F., & Boes, D. (1974). *Introduction to the theory of statistics*. McGraw-Hill.

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

Nabney, I. (2002). *NETLAB: algorithms for pattern recognition*. Springer.

Pardoe, D., & Stone, P. (2006a). Predictive planning for supply chain management. In *Proceedings of the Sixteenth International Conference on Automated Planning and Scheduling*. 21–30.

Pardoe, D., & Stone, P. (2006b). TacTex-05: a champion supply chain management agent. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*. 1489–94.

Podobnik, V., Petric, A., & Jezic, G. (2006). *The CrocodileAgent: research for efficient agent-based cross-enterprise processes*, Springer, vol. 4277 of *Lecture Notes in Computer Science*. 752–762.

Saha, S., Biswas, A., & Sen, S. (2005). Modeling opponent decision in repeated one-shot negotiations. In

Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems. New York, NY, USA: ACM, 397–403.

Sodomka, E., Collins, J., & Gini, M. (2007). Efficient statistical methods for evaluating trading agent performance. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence*. Vancouver, British Columbia, Canada: AAAI Press, 770–775.

Sohn, S., Moon, T., & Seok, K. (2009). Optimal pricing for mobile manufacturers in competitive market using genetic algorithm. *Expert Systems with Applications*, *36*(2), 3448–3453.

Stan, M., Stan, B., & Florea, A. (2006). A Dynamic strategy agent for supply chain management. In *Proceedings of the Eighth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*. Washington, DC, USA: IEEE Computer Society, 227–232.

Swaminathan, J., & Tayur, S. (2003). Models for supply chains in e-business. *Management Science*, *49*(10), 1387–1406.

Swedish Institute for Computer Science (2004–2012). Trading agent competition. Website. Available online, http://www.sics.se/tac.

Tang, C., & Yin, R. (2007). Responsive pricing under supply uncertainty. *European Journal of Operational Research*, *182*(1), 239–255.

Titterington, D., Smith, A., & Makov, U. (1985). *Statistical analysis of finite mixture distributions*. New York, NY, USA: John Wiley and Sons.

University of Minnesota (2003–2012). Trading agent competition. Website. Available online, http://tac.cs.umn.edu/.

van den Berg, J., Kaymak, U., & van den Bergh, W. (2004). Financial markets analysis by using a probabilistic fuzzy modelling approach. *International Journal of Approximate Reasoning*, *35*(3), 291–305.

van Ryzin, G., & Vulcano, G. (2004). Optimal auctioning and ordering in an infinite horizon inventory-pricing system. *Operations Research*, *52*(3), 346–367.

Walsh, W., Parkes, D., Sandholm, T., & Boutilier, C. (2008). 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*. Chicago, IL, USA, 1499–1502.

APPENDIX

A summary of the mathematical notation used in this paper can be found in Table 4.

[Table 4 about here.]

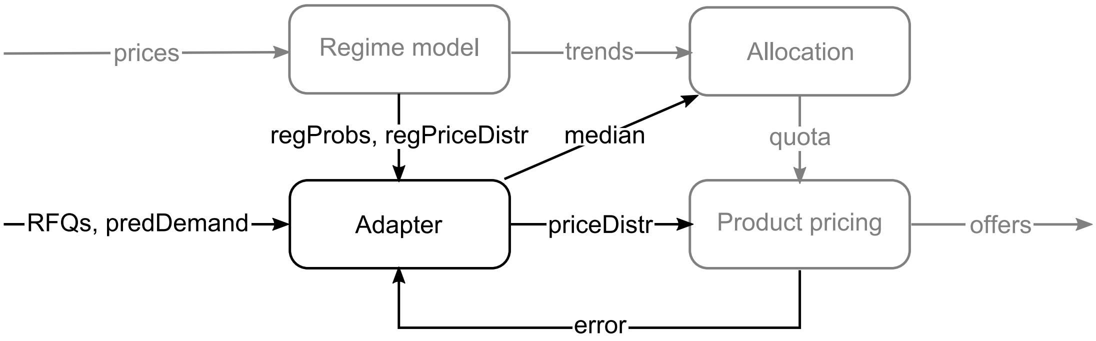


Figure 1: Schematic overview of the DynER sales process for an arbitrary product.

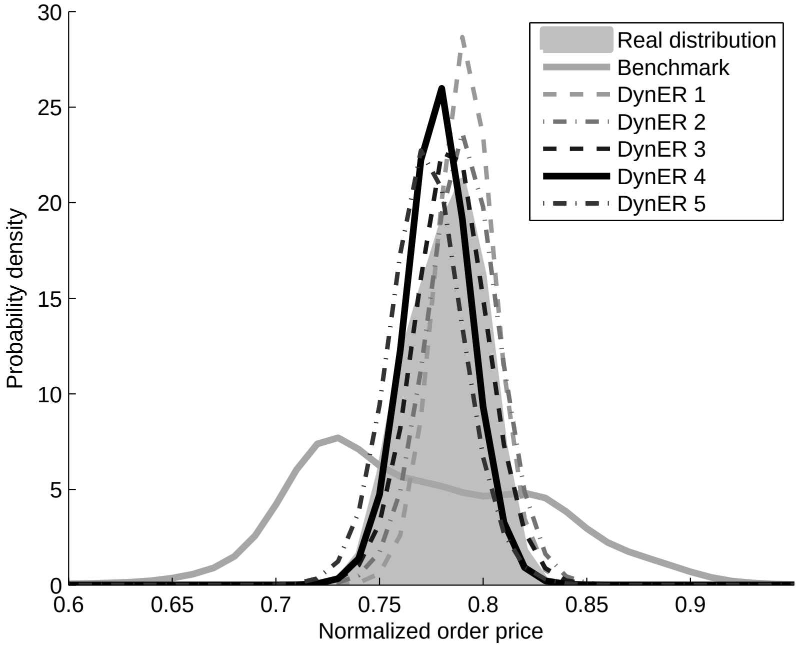
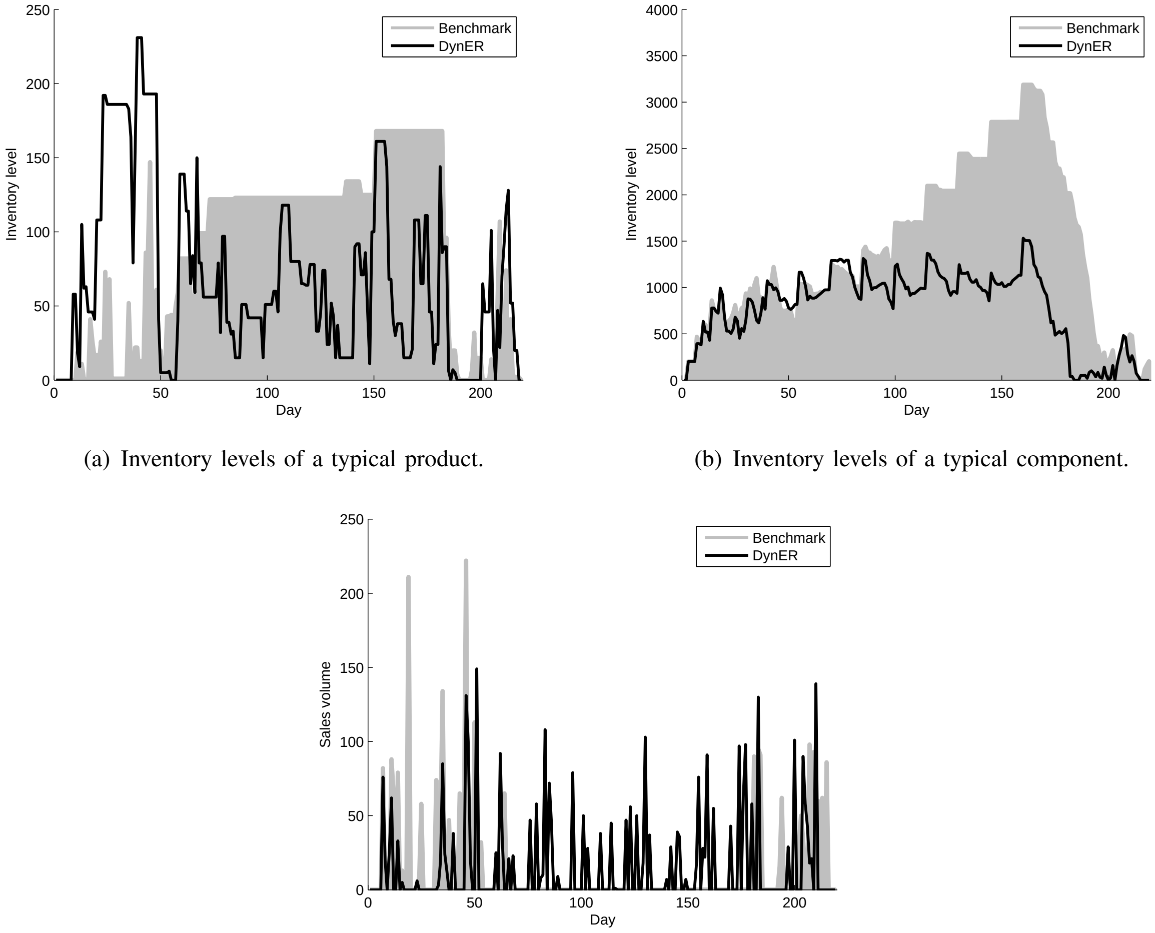


Figure 2: Typical distribution of realized offer prices for a particular good on an arbitrary day, estimated by benchmark and DynER approaches for any of five dominant regimes.

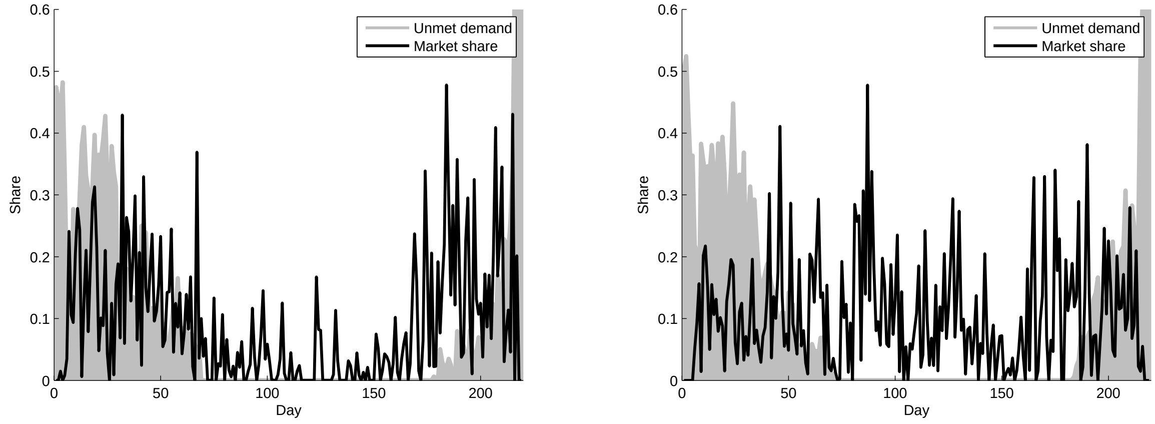


(a) Inventory levels of a typical product.

(b) Inventory levels of a typical component.

(c) Sales volume of a typical product.

Figure 3: Typical sales and inventory levels of finished goods and components over time.



(a) Benchmark.

(b) DynER.

Figure 4: Typical overall MinneTAC market shares and unmet demand levels over time.

Table 1: Optimized number of clusters H for parameter estimating RBFNs and the RMSD of predicted parameter values from their target values per dominant regime k.

k	H	RMSD α_k	RMSD $\ln(\gamma_k)$	RMSD n_{dgk}
1	15	0.045	0.769	0.998
2	15	0.035	0.698	1.073
3	19	0.037	0.651	1.015
4	30	0.039	0.641	0.944
5	23	0.041	0.681	0.809

Table 2: Mean and standard deviation of final bank account balance (in millions) per agent, calculated over all experiments for both the benchmark and the DynER configuration. The performance difference Δ between both configurations and the associated statistical significance (sig.) are reported as well.

	Benchmark		DynER		Δ	
Agent	Mean	Stdev	Mean	Stdev	%	Sig.
MT	-0.355	5.451	1.345	4.109	478.4	0.000
TT	5.144	5.569	5.295	5.501	2.9	0.528
DM	3.907	4.000	4.249	3.845	8.8	0.125
PA	5.627	5.198	5.391	5.143	-4.2	0.096
MC	-2.536	3.845	-2.487	4.273	1.9	0.757
CA	-7.568	5.794	-7.986	5.432	-5.5	0.166

Table 3: Mean and standard deviation of order volume (in thousands) per agent, calculated over all experiments for both the benchmark and the DynER configuration. The performance difference Δ between both configurations and the associated statistical significance (sig.) are reported as well.

	Benchmark		DynER		Δ	
Agent	Mean	Stdev	Mean	Stdev	%	Sig.
MT	4.131	0.569	4.016	0.370	-2.8	0.040
TT	6.646	0.485	6.510	0.533	-2.0	0.000
DM	5.279	0.589	5.424	0.606	2.8	0.000
PA	6.007	0.489	6.021	0.499	0.2	0.323
MC	6.493	0.404	6.525	0.425	0.5	0.175
CA	5.617	0.540	5.579	0.581	-0.7	0.605

Table 4: Summary of notation.

Symbol	Definition
α	Median offer price representation in the truncated log-logistic distribution
β	Smoothing factor for exponential smoothing
γ	Quantification of the tightness of the truncated log-logistic distribution
ϵ	Correction term accounting for the ratio of received to predicted orders
$\widetilde{\epsilon}$	Smoothed correction term accounting for the ratio of received to predicted orders
σ_h^2	Variance of hidden unit h
$d\left(\vec{x}_h, \vec{x}\right)$	Euclidean distance between local function center $\vec{x}_h$ and a a vector of features $\vec{x}$ for hidden unit h
$f\left(p; \theta\right)$	Distribution of normalized valid offer price p, with θ a vector of parameters
$F\left(p; \theta\right)$	Cumulative distribution of normalized valid offer price p, with θ a vector of parameters
$F_1\left(p; \theta\right)$	Cumulative distribution of minimum normalized valid offer price p, with θ a vector of parameters
$h\left(\vec{x}\right)$	Relation between a vector of real-time available information $\vec{x}$ and a value to be estimated (e.g., distribution parameter)
$\hat{h}\left(\vec{x}\right)$	Approximated relation between a vector of real-time available information $\vec{x}$ and a value to be estimated (e.g., distribution parameter)
H	Number of hidden units
$K_h\left(d\left(\vec{x}_h, \vec{x}\right)\right)$	Kernel function of hidden unit h, depending on the distance between the local function center $\vec{x}_h$ and a vector of features $\vec{x}$
$L\left(\theta; p\right)$	Negated log-likelihood function of parameters in vector θ for a sample of normalized valid offer prices p
m	Number of RFQs issued
M	Number of economic regimes
n	Number of normalized valid offer prices
np	Normalized order price
$\widetilde{np}$	Smoothed normalized mid-range order price realized over all RFQs of a product on a game day
p	Normalized valid offer price
p^*	Optimal normalized valid offer price, expected to yield sales quota
$p^{*'}$	Corrected optimal normalized valid offer price, expected to yield sales quota
$P\left(o\|p\right)$	Probability that a customer will place an order o with an agent, given its normalized valid offer price p
$P\left(o\|p\right)'$	Corrected probability that a customer will place an order o with an agent, given its normalized valid offer price p
$P\left(R_k\right)$	Probability of regime R_k
q	Proportion of actually received orders
q^*	Sales quota (desired obtained proportion of orders)
$q^{*'}$	Corrected sales quota (desired obtained proportion of orders)
u	Upperbound of the double-bounded log-logistic distribution
w	Weight of an activation unit
x	Benchmark value
$\hat{x}$	Approximation of x

Algorithm 1 The DynER approach.

```
for each $d$ in $days$ do
    for each $g$ in $products$ do
        // Update error using last feedback, following (11) through (12)
        $error$ = updError(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 price expected to yield quota using (15) through (17)
        $price$ = priceForProb($quota$, $priceDistr$, $error$, $regProbs$);
        // Randomize optimized price and bid on selected RFQs
        priceProduct($d$, $g$, $price$);
    end
end
```