
Proceedings Paper:
Bauso, D., Giarré, L. and Pesenti, R. (2004) Neuro-dynamic programming for cooperative inventory control. In: Proceedings of the American Control Conference. Proceeding of the 2004 American Control Conference , June 30 -July 2,2004 , Boston, MA, USA. IEEE , 5527 - 5532. ISBN 0780383354

Neuro-Dynamic Programming for Cooperative Inventory Control

Dario Bauso, Laura Giarré and Raffaele Pesenti

***Abstract*— In Multi-Retailer Inventory Control the possibility of sharing set up costs motivates communication and coordination among the retailers. We solve the problem of finding suboptimal distributed reordering policies which minimize set up, ordering, storage and shortage costs, incurred by the retailers over a finite horizon. Neuro-Dynamic Programming (NDP) reduces the computational complexity of the solution algorithm from exponential to polynomial on the number of retailers.**

I. Introduction

We consider a two echelon, one-warehouse multi-retailers inventory system. Each day, a stochastic demand materializes at each node. Unfulfilled demand is backlogged. Retailers observe their own inventory level, communicate and make decisions whether to reorder or not from warehouse to fulfill the expected demand. Ordered quantities plus inventory at hand may not exceed storage capacity at each store. Reordering occurs by means of a single track which serves all the retailers. Set up costs are shared among all retailers who reorders, also called *active retailers*. This motivates a certain coordination of reordering policies. The system under concern is depicted in Fig. 1.

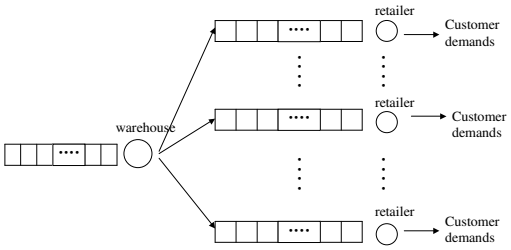

Fig. 1. One-warehouse multi-retailer inventory system

Decentralization of policies under partial information is the main focus in [6]. In [10] the authors analyze the benefits of the information sharing on the performance of the entire chain. In [1] issues are discussed, regarding the use of different kinds of penalties, transfer prices and cost sharing schemes to improve the coordination of policies optimized on a local basis.

In a static context, i.e., for fixed day and fixed inventory levels, we introduced in [3], a distributed consensus protocol [7] for estimating the number of active retailers and coordinating the reordering policies. Each retailer is assumed to choose a fixed *threshold policy*, with threshold l_i on the number of active retailers. In other words one defines its intention to reorder only if at least other $l_i - 1$ retailers are willing to do the same. We proved that consensus on the number of active retailers is asymptotically globally reached and coordination is the same that if the decision making process would be centralized, namely, any retailer has access to the thresholds of all other retailers and chooses whether to reorder or not. The proposed distributed protocol has the advantage that the retailers do not communicate their threshold policy to reach consensus on the number of active retailers.

This paper extends the aforementioned results to a dynamic inventory control context, i.e., where inventory levels change each day. We show that the threshold policies assumed in [3], are strictly connected to the well known (s, S) policies [9], [8]. In some cases, we prove that a optimal policy, for each ith retailer, is to order only in conjunction with at least other $l_i - 1$ retailers. We prove also that the threshold l_i can be computed locally by the ith retailer depending on the current inventory level and expected demand. This is possible by implementing a distributed Neuro-Dynamic Programming (NDP) algorithm polynomial on the number of retailers, which avoid the curse of dimensionality and reduces errors due to model uncertainties.

This paper is organized as follows. In Section II we develop a hybrid model for the cooperative inventory control problem. In Section III we prove that the cost function is K-convex and hence can be efficiently computed in a reduced number of points. We show also that threshold policies on the number of active retailers are optimal. In Section IV we presents the NDP algorithm. In Section V we provide conclusions.

II. Hybrid Model

In this section we present a novel hybrid model for the multi-retailer inventory system (see, e.g., Fig. 2). In particular, in Subsection II-A, we model the n decoupled inventory subsystems. In Subsection II-B, we model the information flow among the subsystems. In Subsection II-C, we introduce the structure of the local controllers and formally state the problem.

A. System Dynamics

Consider a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$; each retailer is a node $v_i \in \mathcal{V}$, where $i \in \Gamma := \{1, 2, ..., n\}$, and each communication link is an edge $e = (v_i, v_j) \in \mathcal{E}$; $i, j = 1, 2, ..., n$. Let $n = |\mathcal{V}|$, where $|S|$ indicates the cardinality

D. Bauso is with DINFO, Università di Palermo, 90128 Palermo, Italy bauso@ias.unipa.it

L. Giarré is with DIAS, Università di Palermo, 90128 Palermo, Italy giarre@unipa.it

R. Pesenti is with DINFO, Università di Palermo, 90128 Palermo, Italy pesenti@unipa.it

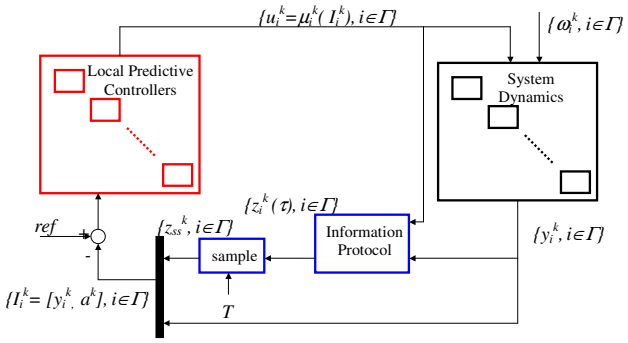

Fig. 2. Block Diagram of the closed loop inventory system.

of the set S. The model input u_i^k is the quantity of inventory ordered by the ith retailer at each stage $k = 0, 1, ..., N-1$. We model with ω_i^k the stochastic demand faced by the ith retailer.

The ith inventory subsystem is a finite-state discrete-time model, that for all $i \in \Gamma$ takes on the form

$$x_i^{k+1} = x_i^k + u_i^k - \omega_i^k.$$

The inventory at hand plus inventory ordered may not exceed storage capacity as is expressed in the following equation

$$x_i^k + u_i^k \leq C_{store}.$$

The ith output y_i^k, referred to as sensed information, is

$$y_i^k = x_i^k,$$

i.e., each retailer observes only his inventory level.

B. Consensus Protocols

The information flow is managed through a *distributed* protocol $\Pi = \{(f_i, h_i, \phi_i) : \text{for all } i \in \mathcal{V}\}$

$$\dot{z}_i^k(\tau) = f_i(z_j^k(\tau), \text{ for all } j \in N_i),\ 0 \leq \tau \leq T, \quad (1a)$$
$$z_i^k(0) = h_i(y_i^k), \quad (1b)$$
$$a^k = \phi_i(z_{ss}^k) \quad (1c)$$

where:

- $f_i : \Re^n \to \Re$ describes the dynamics of the transmitted information of the ith node as a function of the information both available at the node itself and transmitted by the other nodes, as expressed in (1a);
- $h_i : \mathcal{Z} \to \Re$ generates a new transmitted information vector given his output at the stage k, as described in (1b);
- $\phi_i : \Re \to \mathcal{Z}$ estimates, based on current information, the aggregate info (1c).

Here N_i is the neighborhood of the ith retailer, $N_i = \{j \in \Gamma : (v_i, v_j) \in \mathcal{E}\} \cup \{i\}$, i.e., the set of all the retailers j that are connected to i and i itself and

$$z_{ss}^k = \lim_{\tau \to T^-} z_i(kT + \tau), \quad \text{for all } i \in \Gamma, \quad (2)$$

represents the steady state value assumed by $z_i^k(\tau)$ within the interval $[kT,\ (k+1)T]$.

We refer the reader to [7] for studies on the convergence of consensus protocols. For given scenario, defined by the full state vector, $x^k = \{x_i^k,\ \text{for all } i \in \Gamma\}$, the converging value of the transmitted information, a_i^k, plus the sensed information, y_i^k constitute the *partial information* vector, $I_i^k = [y_i^k,\ a_i^k]$ available to the ith retailer.

C. Local Predictive Controllers

The local controllers compute the following cost over a finite horizon

$$J_i(\hat{I}_i^k,\ u_i^k) = \mathrm{E}\left\{ g_i(\hat{I}_i^N) + \sum_{\hat{k}=k}^{N-1} (\alpha^{\hat{k}} g_i(\hat{I}_i^{\hat{k}},\ u_i^{\hat{k}})) \right\} \quad (3)$$

where $\hat{I}_i^k$ is the predicted information and α^k is the discount factor at stage k. The stage cost $g_i(\hat{I}_i^k,\ u_i^k, k)$ is defined as

$$\begin{aligned} g_i(\hat{I}_i^k,\ u_i^k, k) \quad &= \tfrac{K}{a^k}\delta(u_i^k) + cu_i^k + p\mathrm{E}\{\max(0, -\hat{y}_i^{k+1})\} \\ &\quad + h\mathrm{E}\{\max(0, \hat{y}_i^{k+1})\} \end{aligned}, \quad (4)$$

where K represents the set up cost, c is the purchase cost per unit stock, p is the penalty on storage, h the penalty on shortage, and $\delta(u_i(k))$ is zero if the ith retailer does not reorder, and one if he reorders.

As will be clear later on, the idea of the solution algorithm is to use a simulation-based tunable predictor of the form

$$\hat{I}_i^{k+1} = \begin{bmatrix} \hat{y}_i^{k+1} \\ \hat{a}_i^{k+1} \end{bmatrix} = \begin{bmatrix} x_i^k + u_i^k - \hat{\omega}_i^k \\ \psi_i(a_i^k,\ u_i^k) \end{bmatrix} \quad (5)$$

In (4) we assume that the set up cost is equally shared among the active retailers.

We report hereafter the formalization of the problem under concern. Given a set of retailers reviewed as dynamic agents of a network with topology $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.

Problem (Local Controllers Synthesis) *For each ith retailer, determine the reordering policy $u_i^k = \mu(I_i^k)$, that minimizes the N-stage individual payoff defined in (3).*

Subproblem (Protocols Design) *Determine a distributed protocol Π which maximizes the set of active retailers A_Π.*

III. Dynamic Programming Approach

In this section, we prove that the inventory must be ordered in quantity thus to fulfill exactly the expected demand for the upcoming days, as summarized in Theorem 3.1. We provide an intuitive explanation of such a result.

Let $K^k = \frac{K}{a^k}$ the set up cost charged to each retailer that reorders at stage k, and $d_i^k = x_i^k + u_i^k$, the instantaneous inventory position, i.e., the inventory level just after the order has been issued. Then we claim as follows.

- If the setup cost K^k decreases with time (in the future more and more retailers are interested in reordering) retailers place short term orders. Optimal policies are multiperiod policies $(s^k,\ S^k)$, with a unique lower and upper threshold, (see, e.g., Fig. 3).

- On the contrary, if the setup cost K^k increases with time (in the future less and less retailers are interested in reordering), retailers place long-term orders. Optimal policies are multiperiod policies $(s^k,\, S^k)$ with multiple thresholds at different inventory levels (see, e.g., Fig. 4).

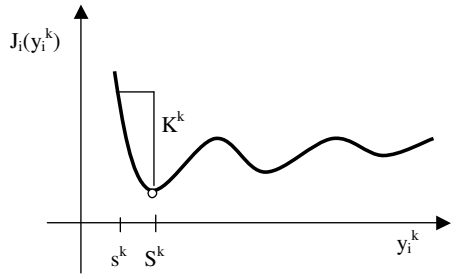

Fig. 3. Intuitive plot of the cost when the set up cost decreases with time: single thresholds $(s^k,\, S^k)$.

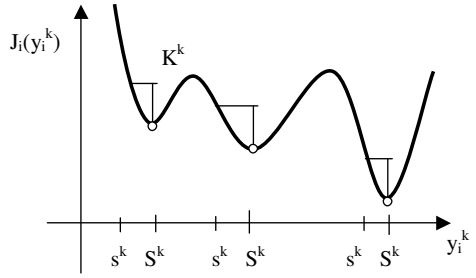

Fig. 4. Intuitive plot of the cost when the set up cost increases with time: multiple thresholds $(s^k,\, S^k)$.

A. Searching for Structure: K-convex analysis

To show that the individual objective functions, $J_i;\, i \in \Gamma$, have at most N local minima, we, first, apply the DP algorithm (6)-(7) to minimize the cost (3). The Bellman's equation is then rearranged by defining a new function $H(\cdot)$ as in which verifies the $\overline{K}_i$-convexity property, where $\overline{K}_i$ is the maximum set up cost incurred by the ith retailer over the horizon. Exploiting the definition of the inventory position d_i and of the set up cost K^k, we rewrite the stage cost (4) as

$$\begin{aligned} g_i(d_i^k,\, a^k) \;\; = \;\; & K^k\delta(u_i^k) + cu_i^k + p\mathrm{E}\{\max(0, -(d_i^k - \omega_i^k))\} \\ & + h\mathrm{E}\{\max(0, d_i^k - \omega_i^k))\}. \end{aligned}$$

By applying the dynamic programming algorithm, we have

$$J_i^N(I_i^N) = 0, \tag{6}$$

$$J_i^k(I_i^k) = \min_{u_i^k \in U}[g_i(d_i^k,\, a^k) + \alpha^{k+1}\mathrm{E}\{J_i^{k+1}(I_i^{k+1})\}]. \tag{7}$$

Let us define the new function

$$\begin{aligned} G_i^k(d_i^k,\, a^{k+1}) \;\; & = cd_i^k + \mathrm{E}\{p\max(0, -(d_i^k - \omega_i^k)) \\ & + h\max(0, d_i^k - \omega_i^k) + J_i^{k+1}(I_i^{k+1})\}, \end{aligned}$$

and rewrite the Bellman's equation (7) as follows

$$J_i^k(I_i^k) = -c_i x_i^k + \min_{d_i^k \geq x_i^k}[K^k + G_i^k(d_i^k,\, a^{k+1}),\, G_i^k(x_i^k,\, a^{k+1})]. \tag{8}$$

Note that if we can show that J_i^{k+1} is K^k-convex then G_i^k is also K^k-convex and the Bellman's equation (8) has a unique minimizer.

Indeed, it has been proved in [4] that K^k-convexity of $G_i^k(d_i,\, a^{k+1})$ implies K^k-convexity of $J_i^k(I_i^k)$.

This represents a sufficient condition that guarantees optimality of multiperiod $(s_i^k,\, S_i^k)$ order-up-to policies.

We recall that s_i^k represents the minimum threshold on inventory level below which retailers reorders to restore level S_i^k.

Let us remind that S_i^k minimizes $G_i^k(\cdot,\, a^{k+1})$ and threshold s_i^k verifies

$$G_i^k(s_i^k,\, a^{k+1}) = G_i^k(S_i^k,\, a^{k+1}) + K^k.$$

Now, let us call $\underline{s}_i^k$, the threshold which corresponds to the assumption that the ith retailer is charged the whole set up cost; namely we have $K_i^k = K;\, i \in \Gamma$. At the same way, let us define with $\overline{s}_i$ the threshold computed as if all retailers would share equally the setup costs; thus, each retailer is charged a set up cost $K_i^k = \frac{K}{n}$, namely one nth of the entire cost K. We now explicit dependence of threshold s_i^k on set up cost K_i^k by defining the function $s_i^k(\frac{K}{a^k})$ for which it holds $\underline{s}_i^k \leq s_i^k(\cdot) \leq \overline{s}_i^k$.

Now, let us call

$$H_i^k(d_i^k,\, a^k) = \min_{y_i^k \geq x_i^k}[K^k + G_i^k(d_i^k,\, a^{k+1}),\, G_i^k(x_i^k,\, a^{k+1})].$$

In the following, we consider a^k a parameter and show that the individual objective function, $J_i^k(x_i);\, i \in \Gamma$, which is generically non convex, has all local minima coincide with the demand summed over one or more days.

Theorem 3.1: Solutions of the Bellman's equation (7) are at most $N - k$ different multi period policies $(s_i^k,\, S_i^k)$, where $S_i^k \in \{\sum_{j=k}^{M}\omega_i^j;\, M = k,\, k+1,\, ...,\, N\}$ and threshold s_i^k verifies $G_i^k(s_i^k,\, a^{k+1}) = G_i^k(S_i^k,\, a^{k+1}) + K^k$. Policy are associated to different intervals of inventory levels.

Proof: The essential idea is that the cost is piecewise linear. This is evident in the Bellman's equation where the cost J_i^k is the summation of a piecewise linear stage cost g_i^k (with unique global minimum at ω_i^k) and a piecewise linear future cost (with potential local minima at $\omega_i^k + S_i^{k+1}$). ■

An immediate consequence of the above theorem is that the set of feasible decisions is finite and each element represents the exact ordered quantity to fulfill the expected demand for the upcoming 1, 2, ..., N days.

B. Threshold Reordering Policies

The aim is now to show that Nash equilibrium reordering policies have a threshold structure on the number of retailers interested in reordering. To see this, we first introduce a

preliminary lemma on single-stage inventory control and reinterpret the concept of threshold $(s,\, S)$ in a way more suitable for a multi-retailer scenario. In particular we change a threshold on inventory level s into a threshold l on "how many retailers are interested in reordering".

Lemma 3.2: (Single-Stage Optimization) For each inventory level x_i there exists a threshold $l_i \in \{1,\, 2,\, ...,\, n\}$, such that the reordering policy

$$\mu_i(I_i) = \begin{cases} S_i - x_i & \text{if} \quad a \geq l_i \\ 0 & \text{if} \quad a < l_i \end{cases} \tag{9}$$

is a Nash equilibrium for the single-stage formulation of the Multi-retailer Inventory Control Problem.

Proof: From Theorem 3.1, if $N = 1$, we have a unique multi period policy $(s_i,\, S_i)$. This means that retailers make decisions according to

$$\mu_i(I_i) = \begin{cases} S_i - x_i & \text{if } x_i < s_i \\ 0 & \text{if } x_i \geq s_i. \end{cases} \tag{10}$$

For given x_i, the idea is to find the minimum value of l_i that verifies the condition $x_i < s_i$. This is straightforward for the two limit cases of "low" and "high" inventory level, namely $x_i < \underline{s}_i$, and $x_i \geq \overline{s}_i$ respectively. It is left to prove (10) for the intermediate case $\underline{s}_i \leq x_i \leq \overline{s}_i$ (see proof in [2]). ■

As evident from (9) the single-stage formulation of the Multi-retailer Inventory Control Problem leads to reordering policies with a threshold structure. Results from Lemma 3.2 can be extended to the multi-stage formulation.

Theorem 3.3: (Multi-Stage Optimization) For each inventory level x_i^k there exists a threshold $l_i^k \in \{1,\, 2,\, ...,\, n\}$, such that the reordering policy is

$$\mu(I_i^k) = \begin{cases} S_i^k - x_i^k & \text{if} \quad a^k \geq l_i^k \\ 0 & \text{if} \quad a^k < l_i^k \end{cases} \tag{11}$$

is a Nash equilibrium for the multi-stage formulation of the Multi-retailer Inventory Control Problem.

Proof: The structure of the proof is the same as for the single-stage inventory problem, in Lemma 3.2. Only, note that from Theorem 3.1, we now have at most $N - k$ different multi period policy $(s_i^k,\, S_i^k)$, each one associated to a different interval of inventory levels. The trick of the prove is to repeat the argument above for each interval. ■

We then conclude that optimizing the multi-retailer inventory control problem over a multi-stage horizon leads to Nash equilibrium reordering policies with threshold structure on the number of active retailers.

C. Local Estimation via Consensus Protocols

In this subsection, we discuss the solution of the subproblem on protocol design. The focus is on consensus protocols to estimate the number of active retailers a^k. Indeed, given the vector $l = \{l_i\}$, collecting the optimal thresholds, each retailer makes the decision "do not reorder" if his local estimation is lower than his threshold, as expressed in Eq. (11). We assume that the transmitted information is the current estimate of the percentage of retailers who are interested in reordering. The current estimate $z_i(\cdot)$ is re-initialized to $\{0,\, 1\}$ at the beginning of each time interval $[kT,\, kT + 1]$ based on the current inventory level x_i^k. In particular, if the ith inventory level is "low", i.e., the corresponding threshold l_i does not exceed the network size n, then the retailer is willing to reorder; he has got no information yet except his observed inventory level; thus, he assumes that all other retailers are in the same circumstances (spatially invariant assumption) and set $z_i^k = 1$, indicating that everyone is today interested in reordering. On the contrary, if the inventory level is "high" (l_i exceeds n), he is not willing to join the group to order and set $z_i^k = 0$, indicating that no one is in need to reorder. Thus we can write

$$z_i^k = \begin{cases} 0 & l_i(x_i^k) > n \\ 1 & \text{otherwise.} \end{cases} \tag{12}$$

Then, each retailer updates the estimate on-line on the basis of new estimates data received from neighbors. At any time, t_i, whenever the number of retailers interested in reordering, a^k, goes below his threshold, l_i, the ith retailer communicates his decision to "give up" to reorder by activating an exogenous impulse signal, $\delta_i(t - t_i)$. This exogenous impulse can be activated only one time (once you exit the group you are no longer allowed to rejoin it) and only when the all local estimates have reached consensus on a final value. This occurs every t_f, where t_f is an estimate of the worst case possible settling time of the protocol dynamics.

Given (12), an average-consensus protocol leads all local estimates to converge to the max a^k (see [3]). The continuous-time average-consensus protocol takes on the form

$$\begin{cases} h_i(x_i^k) & = & l_i(x_i^k) \leq n \\ f_i(z^k(\tau)) & = & -L_{i\bullet} z^k(\tau) + \delta_i(t - t_i) \cdot u_i^k \\ \phi(z_i^k(\tau)) & = & n(\lim_{t \to T^-} z_i^k(\tau)). \end{cases}$$

where L is the Laplacian matrix of the communication network topology; t_i is in turn the time instant where the current estimate converges to a value below the threshold; it can be defined by the following logic conditions

$$t_i : s.t.[l_i(x_i^k) > n] \quad \text{OR} \quad [(l_i(x_i^k) \leq n) \text{AND} (n z_i(t_i) < l_i) \\ \text{AND } (t_i = k t_f,\, k \in \mathcal{N})].$$

We refer the reader to [3] for details on the optimality of the protocol above.

IV. NDP Solution Algorithm

In this section, we cast the hybrid model within the framework of neuro-dynamic programming.

A. Consensus on Features a_i^k

To review the features as a compact description of the behavior of the other retailers, we consider i) the NDP architecture based on feature extraction displayed in Fig. 5

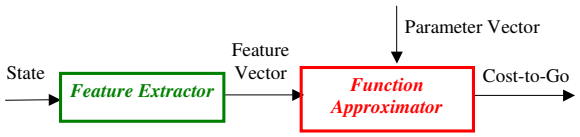

Fig. 5. The information flow management uses consensus protocols to extract the features.

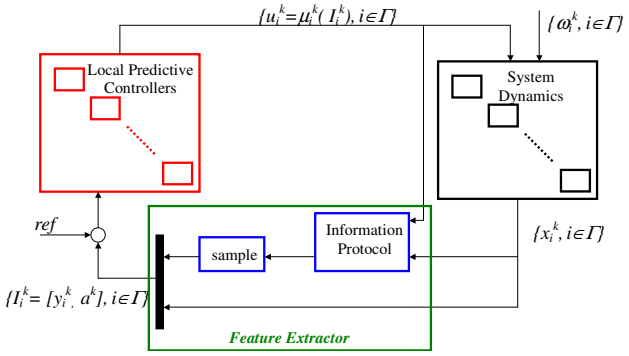

Fig. 6. Block Diagram of the closed loop system.

(see e.g. [5]) and ii) the block diagram of the Hybrid Model displayed in Fig. 6.

The full state vector of the hybrid model, x^k becomes, in the approximation architecture, the input to the feature extractor. The information flow management block can be reviewed as the feature extractor. The full state vector reduces to the partial information vector $I_i^k = [y_i^k, a_i^k]$ available to the ith retailer. Each local controller implement a function approximator, which receives the partial information vector and returns the individual cost-to-go $\widetilde{J}_i^k(I_i^k, r)$ over the horizon.

B. Linear Architecture

We assume that the probability distribution over all potential values assumed by a^k propagates according to the linear dynamics $a^{k+1} = a^k \Psi^k$ where $\Psi^k = \{\psi_{ij}^k, i, j \in \Gamma\}$. In this case we have i) a matrix of weights r that coincides with the transition probability matrix of the predictor, namely, $r = \Psi = \{\Psi^k, k = 1, 2, ..., N\}$, and ii) *basis functions* $\widetilde{J}_i^{k+1}(I_i^{k+1}, a^{k+1})$ representing different future costs associated to different a^{k+1}.

The approximation architecture linearly parameterizes the future costs associated to all possible behaviors of the other retailers over the horizon. This can be described as

$$\sum_{a^{k+1}=1}^{|\mathcal{Z}|} \Psi_{a^k,\, a^{k+1}}^k \widetilde{J}_i^{k+1}(I_i^{k+1}, a^{k+1}) = \psi_{a^k \bullet}^k \widehat{J}_i^{\,k+1}(I_i^{k+1}, \bullet)^T,$$

where $\psi_{a^k \bullet}^k$ is the row of the transition probabilities from a^k to all possible a^{k+1}, and $\widehat{J}_i^{\,k+1}(I_i^{k+1}, \bullet)^T$ is the transposed row of the associated future costs.

ω_1	4	8	6	5	7	8	4	5	6	8
ω_2	0	0	1	7	8	0	6	2	1	4
ω_3	0	3	2	0	3	1	1	3	3	0

TABLE I
EXPECTED DEMAND FOR THE UPCOMING TEN DAYS.

C. The NDP Algorithm

This Algorithm is organized in two parts. In the first part the retailers compute the set of admissible decisions U_i^k and reachable states R_i^k over the horizon. The second part presents three steps.

1) *Policy improvement*. For given prediction Ψ, we improve the policy via the stochastic Bellman's equation backwards in time
$$\mu_i^k(I_i^k) = \text{argmin}_{u_i^k \in U_i^k(x_i^k)} \left[g_i(I_i^k, u_i^k, k) + \alpha^{k+1} \psi_{a^k \bullet}^k \widehat{J}_i^{\,k+1}(I_i^{k+1}, \bullet) \right].$$
2) *Value iteration*. The improved policy is valued through repeated Quasi-Monte Carlo simulations. Active exploration guarantees that initial states are sufficiently spread over the local minima. During the value iteration we compute and store the number of times a transition Ψ_{ij} occurs during the repeated finite length simulations. At the end of each simulation, the protocol runs over the horizon and returns the training set for the next step.
3) *Temporal Difference*. We use the training set to update the transition probabilities of the predictor.

The tree steps are iteratively repeated until convergence of policies.

Lemma 4.1: Each iteration of the NDP algorithm, for given initial state x^0, has computational complexity polynomial on the number of retailers, i.e., $O(n^2\, N\, R^2)$

Proof: The proof starts from considering that the complexity of the algorithm depends essentially on the complexity of the second part. Here, we write the Bellman's equation considering the set of feasible decisions U_i^k, for each retailer $i \in \Gamma$, for each stage $k = 1, 2, ..., N$ and for each decomposed state $I_i^k \in (R_i^k \times \Gamma)$. Thus, complexity is $O(n^2\, N\, R^2)$. ■

Assuming that convergence is achieved in a finite number of iterations, the Temporal Difference Algorithm returns stochastic Nash equilibrium policies, paths and costs-to-go. Further efforts are still to be made, oriented to investigate the convergence conditions of this algorithm.

Example 1: Let us consider a group of three retailers and parameters $K = 24$, $p = 8$, $h = 1$, and $c = 2$. Retailers face a stochastic poissonian demand with expected values over the horizon of ten days as in Table I.

At the first iteration, no communication has occurred among the retailers and the "policy improvement" returns the uncoordinated reordering policies displayed in Fig. 7.

The "value iteration" consists in 12 simulations of the inventory system under the improved reordering policies.

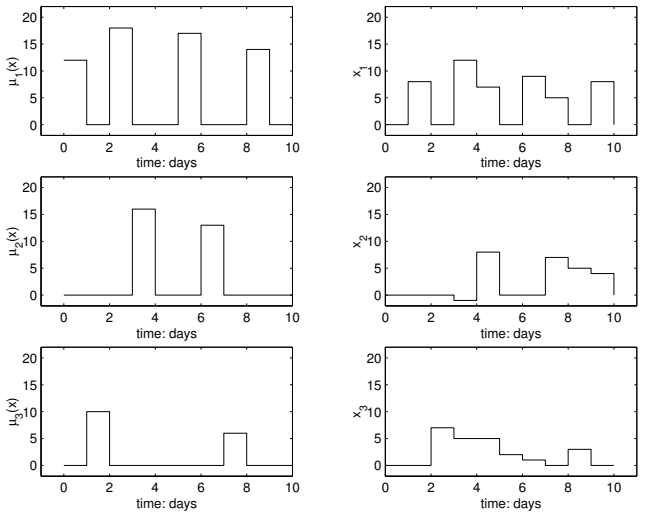

Fig. 7. Uncoordinated reordering policies.

The set of initial states is a stochastic sequence extracted from a poissonian distribution with mean value respectively, equal to 25, 10, and 6 for the 1st, 2nd, and 3rd retailer. Indeed, we know from deterministic simulation results that J_1 has potential local minima at 18, 23, 30, J_2 at 1, 8, 16, and J_3 at 8, 10 as displayed in Figure 9 (solid and dotted lines). Here, the costs associated to the 1st, 2nd, 3rd and 4th policy improvements when demand is deterministic are represented by four lines of different colors (blue, red, magenta, and red). At the end of each simulation the retailers run a consensus protocol returning a^k over the horizon. Based on this new aggregate information, during the "temporal difference" the retailers update the transition probabilities of the predictor and a new iteration starts. In this example, the algorithm eventually converges to a Nash equilibrium in six iterations returning a coordinated distribution of reorders over the horizon as shown in Fig. 8. We see from Fig. 9

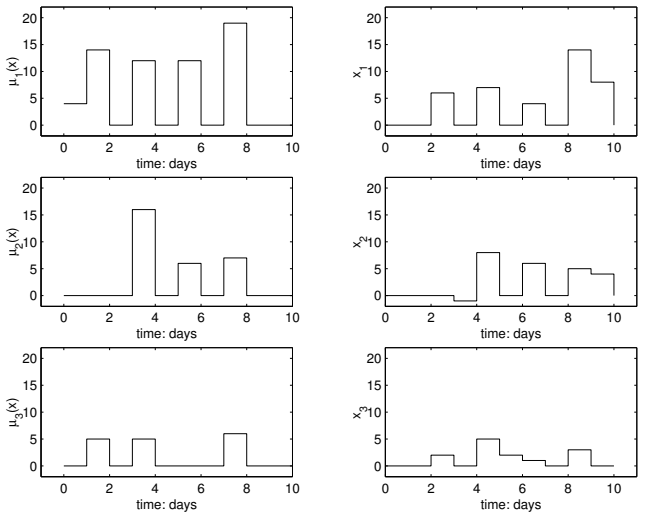

Fig. 8. Coordinated reordering policies.

that the costs-to-go at the 4th and 5th iteration (green and red crosses) draw much near to the cost-to-go of the deterministic problem. We may conclude that the NDP algorithm possesses satisfying learning capabilities.

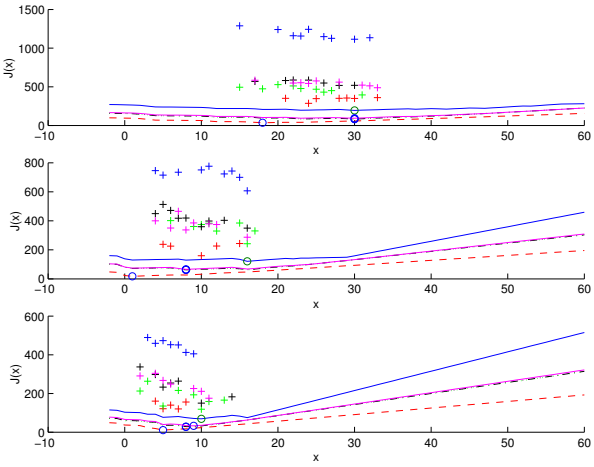

Fig. 9. Costs vs inventory: deterministic (colored lines) and stochastic demand (colored crosses).

V. Conclusion

In this paper we propose an NDP approach to coordinate the reordering policies of a group of retailers. Coordination is motivated by the possibility of sharing set up cost when orders are placed in conjunction. We develop a hybrid model to describe the inventory subsystems and the information flow. We designed consensus protocols for the information flow. Finally we presented a scalable and suboptimal NDP algorithm.

References

[1] S. Axsäter, "A framework for decentralized multi-echelon inventory control", *IIE Transactions*, vol. 33, no. 1, 2001, pp. 91-97.

[2] D. Bauso, "Cooperative Control and Optimization: a Neuro-Dynamic Programming Approach", *Ph. D. Thesis* Università di Palermo, Dipartimento di Ingegneria dell'Automazione e dei Sistemi, Dec. 2003.

[3] D. Bauso and L. Giarrè and R. Pesenti, "Distributed Consensus Protocols for Coordinating Buyers", *Proc. of the IEEE Conference on Decision and Control*, Maui, Hawaii, Dec. 2003.

[4] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 2nd ed. Bellmont, MA: Athena, 1995.

[5] D. P. Bertsekas and J. N. Tsitsiklis,"Neuro-Dynamic Programming", *Athena Scientific*, Bellmont, MA, 1996.

[6] J. C. Fransoo, M. J. F. Wouters and T. G. de Kok, "Multi-echelon multi-company inventory planning with limited information exchange", *Journal of the Operational Research Society*, vol. 52, no. 7, Jul. 2001, pp. 830-838.

[7] R. Olfati Saber and R. M. Murray, "Consensus Protocols for Networks of Dynamic Agents", *Proc. of American Control Conference*, Denver, Colorado, Jun. 2003.

[8] H. E. Scarf, "Inventory Theory", *Operations Research*, vol.50, no.1, Jan-Feb 2002, pp.189-191.

[9] H. E. Scarf, "The Optimality of (s, S) Policies in the Dynamic Inventory Problem", *Mathematical Methods in the Social Sciences*, Stanford University Press, Stanford, CA, 1995.

[10] Z. Yu, H. Yan and T. C. E. Cheng, "Modelling the benefits of information sharing-based partnerships in a two-level supply chain", *Journal of the Operational Research Society*, vol. 53, no. 4, Apr. 2002, pp. 436-446.