Safety Stock Placement in Non-cooperative Supply Chains

Péter Egri[1]

Abstract. The paper studies the safety stock placement problem in decentralised supply chains consisting of autonomous stages. For the inventory optimisation problem we apply the guaranteed-service model, while the non-cooperative attitude is handled with mechanism design theory. We propose and investigate four different mechanisms based on the Vickrey – Clarke – Groves scheme, and their distributed implementation. We illustrate on numerical examples how the mechanisms achieve the globally optimal solution in different ways.

1 INTRODUCTION

In order to provide high service levels for the customers, companies have to maintain inventories, and these are accumulated at the most expensive point of the supply chain as end-products [6]. For example, in the U.S. automotive sector recently so much finished cars have been kept in inventories, that they would have been enough for satisfying average demand for 60 days [1]. Japanese auto manufacturers perform significantly better, e.g., Toyota keeps finished goods to cover demand for a 30 days shorter period than General Motors.

European automotive companies face similar problems. Customers expect to have their orders fulfilled in a couple of days, and they demand very high service levels [6]. Recently, several efforts have been made in order to cope with this challenge by innovatively applying modularity, flexibility, lead-time reduction [12] and collaborative planning [3]. These solutions deal with technologies, short-, medium-, and long-term planning, lean production, but the global supply chain design optimization is often missing.

Inventory positioning is such a strategic issue in complex supply networks—like the one indicated on Fig. 1—that aims at minimising overall inventory cost, while guaranteeing a given service level for the customers. There are examples from the automotive industry for 30% reduction in inventory levels after repositioning of the inventories, while at the same time, preserving the high standards of the service [16]. In [11] it is mentioned that usually 25-50% reduction in holding cost is achievable, and the inventory positioning is illustrated on some large-scale industrial examples.

However, the applicability of such global optimisation approaches in distributed environments requires cooperative attitude, i.e., that the participants agree on minimising the total costs. This may be—although not necessarily—true in the supply network of a single company, but almost inconceivable in a network consisting of different companies.

Global optimisation problems involving agents with different goals can be successfully handled by *mechanism design* theory, which facilitates the alignment of conflicting goals with the global objective. In this paper we combine an inventory positioning model with mechanism design analysis in order to extend the applicability of strategic supply chain design methods across companies.

[1] Fraunhofer Project Center for Production Management and Informatics, Computer and Automation Research Institute, Hungarian Academy of Sciences, Kende u. 13-17, 1111 Budapest, Hungary, email: egri@sztaki.hu

Figure 1. A part of an automotive supply network.

The remainder of the paper is organised as follows. In Section 2 we overview the related literature. Next we present the optimisation model for serial chains, and investigate four different mechanisms that can achieve the optimal solution in Section 3. We demonstrate the differences of the mechanisms using a numerical study in Section 4. Finally, in Section 5, we conclude the paper and enumerate some possible future research directions.

2 LITERATURE REVIEW

Mechanism design theory deals with the problem of constructing the rules of a game with incomplete information in order to achieve some preferred outcome. It assumes an independent, benevolent decision maker, who collects the private information from the agents, decides about the outcome, and pays to the agents for disclosing the private knowledge.

One of the main achievements in this field is the Vickrey – Clarke – Groves (VCG) mechanism, which is the only one in the general model that can provide *efficient* (globally optimal) and *truthful* (agents are not interested in lying about their private information) behaviour. Nisan and Ronen combined the classic mechanism design theory with computer science considerations in their seminal paper, where they also illustrated the application of the VCG mechanism on the shortest path problem [13]. It was later proved that despite the advantageous truthfulness and efficiency properties of the presented mechanism, it tends to overpay the agents, and the overpayment can be arbitrary large [5]. Recently, algorithmic mechanism design has been extensively used in multiagent optimization problems, such as multiagent planning [20] and resource allocation [2].

Implementing a mechanism without an independent decision maker is the field of *distributed mechanism design* [14, 15]. In [7] a general method called *replication* is presented for implementing VCG mechanisms in a distributed way. However, it does not solve the problem of *budget-balance*: an independent source for the agents' payments is still required.

Applying mechanism design for supply chain optimization is not completely new in the literature. Both [8] and [9] present different decision problems in decentralised supply chains, and they apply the VCG mechanism for solving them. However, both models assume an independent decision maker, and do not investigate the possibilities of distributed implementation. In [4] a two-stage supply network is considered with a single supplier and multiple retailers, and a combined mechanism design and information elicitation model is developed. In this special case, a non-VCG mechanism can be applied, which is truthful, efficient, budget-balanced, and can be implemented in a distributed way, resulting in a theoretical model for the Vendor Managed Inventory (VMI) business practice.

A recent review of general inventory control models in supply chain management can be found in [18]. The problem of safety stock placement in supply chains is discussed in [11], where two different approaches, the stochastic- and the guaranteed-service models are presented. In this paper, we adopt the latter one, and repeat its solution method in the simplest case, considering a serial supply chain in Section 3.1.

3 MODEL

In this section we investigate a strategic supply chain design problem, the safety stock placement, in a non-cooperative setting with rational agents. We consider a serial supply chain with n stages, where the nodes represent manufacturing or transportation operations as shown in Fig. 2. Inventory can be held after each node with different h_i unit holding costs. The market demand is stochastic, but the T_i processing lead-times at the nodes are deterministic. We assume that there is no fixed ordering or setup cost, and the nodes apply a *base-stock policy*: an order for stage i immediately generates an order with the same quantity towards stage $i+1$ in order to maintain the base-stock level. We also assume the *guaranteed-service model*: guaranteed service time S_i means that if stage $i-1$ places an order in period t, it receives the goods in period $t+S_i$, and the service time for the final customers is given as a boundary condition ($S_1 = s_1$).

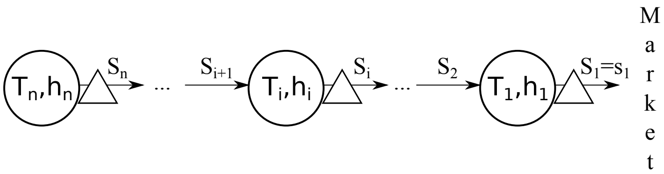

Figure 2. Supply chain setting.

3.1 Centralised approach

The demand in each period is assumed to be independent, normally distributed random variable with mean μ and standard deviation σ. Thus the total demand of t consecutive periods is normally distributed with mean μt and standard deviation $\sigma\sqrt{t}$. The required inventory for satisfying the demand of t periods is therefore $\mu t + k\sigma\sqrt{t}$, where μt is the expected demand and $k\sigma\sqrt{t}$ is the safety stock. The k safety factor should be determined depending on the allowed probability of stock-out, $1-\alpha$, where α denotes the required *service level*. Table 1 shows the appropriate safety factors for some α values (based on [17]). It is assumed that the demand over t periods cannot exceed $\mu t + k\sigma\sqrt{t}$ (or else it is lost, backlogged, served from an other source or with extraordinary production).

Table 1. Service level and safety factors.

α	90%	91%	92%	93%	94%	95%	96%	97%	98%	99%	99.9%
k	1.28	1.34	1.41	1.48	1.56	1.65	1.75	1.88	2.05	2.33	3.09

Guaranteed service time $S_i \geq 0$ means that if stage $i-1$ places an order in period t, it receives it in period $t + S_i$. With the processing lead-time, the *replenishment time* at stage i will be $S_{i+1} + T_i$, where it is assumed that $S_{n+1} = 0$. If stage i wants to provide service time S_i, it therefore needs to hold inventory for $S_{i+1} + T_i - S_i$ periods, which is called the *net replenishment time*.

Since negative net replenishment time is meaningless, we have the constraints $S_i \leq S_{i+1} + T_i$. From this limitation also follows, that if S_1 should equal to s_1, the minimum service time at stage i is s_1 minus the total lead-times in the chain $(i-1, \ldots, 1)$. Let us define the minimum service time as

$$\underline{S}_i = \max\left\{0, s_1 - \sum_{j=1}^{i-1} T_j\right\}, \quad (1)$$

thus we have the constraint $\underline{S}_i \leq S_i \leq S_{i+1} + T_i$.

Using the net replenishment time, the base-stock level at stage i can be calculated as

$$B_i = \mu(S_{i+1} + T_i - S_i) + k\sigma\sqrt{S_{i+1} + T_i - S_i}, \quad (2)$$

and the expected inventory in period t becomes

$$\mathbb{E}[I_i(t)] = B_i - \sum_{j=0}^{t-S_i} \mu + \sum_{j=0}^{t-S_{i+1}-T_i} \mu = k\sigma\sqrt{S_{i+1} + T_i - S_i}. \quad (3)$$

The total expected inventory holding cost for the supply chain is

$$\sum_{i=1}^{n} h_i k\sigma\sqrt{S_{i+1} + T_i - S_i}, \quad (4)$$

therefore the optimal service times can be determined with the following non-linear program:

$$\min \sum_{i=1}^{n} h_i \sqrt{S_{i+1} + T_i - S_i} \quad (5)$$

s.t.

$$\underline{S}_i = \max\left\{0, s_1 - \sum_{j=1}^{i-1} T_j\right\} \quad i \in \{1, \ldots, n\} \quad (6)$$

$$\underline{S}_i \leq S_i \leq S_{i+1} + T_i \quad i \in \{1, \ldots, n\} \quad (7)$$

$$S_1 = s_1 \quad (8)$$

$$S_{n+1} = 0 \quad (9)$$

Let $P(T, h, s_1)$ denote the above program with the lead-time and cost vectors T and h. Note that $P(T, h, s_1)$ is independent from μ, σ and k. This means, that whether a stage should hold inventory or not is independent from the specific mean and variance of the demand, and can be determined only with the knowledge of the lead-times and holding costs.

Simpson proved that the minimum of the objective function occurs at a vertex of the convex polyhedron defined by the constraints of the program [19]. This means that in an optimal S^*, each S_i^* equals either $S_{i+1}^* + T_i$ (when the stage does not hold any inventory) or $\underline{S}_i$ (where the stage offers immediate—or minimal—service time). Based on this result, Graves and Willems showed that the problem can be solved efficiently using the following dynamic programming recursion [10]:

$$f_{n+1} = 0 \tag{10}$$

$$f_i = \min_{j=i+1\ldots n+1} \left\{ f_j + h_i \sqrt{\underline{S}_j + \sum_{l=i}^{j-1} T_l - \underline{S}_i} \right\} \quad (i \leq n) \tag{11}$$

where f_i is the optimal cost in the $(n, n-1, \ldots, i)$ chain if stage i holds safety stock for providing $\underline{S}_i$ service time.

3.2 Mechanism design for safety stock placement

Let us consider the case, when the stages of the supply chain are independent, rational entities with private information, i.e., h_i and T_i are only known at stage i. Instead of minimising the total cost, each stage intends to minimise its own cost, which can be done simply by not holding stock at all, except at stage 1, which has to keep an enormous end-product stock in order to guarantee service time s_1.

The solution for this situation provided by the mechanism design theory is to assume a central decision maker, who collects the private information from the stages, determines the service times and provides some payment t_i for each stage, see Fig. 3. Since the stages might distort their disclosed information, we denote the inventory holding costs and lead-times collected by the mechanism as $\hat{h}_i$ and $\hat{T}_i$. The utility function of the stages becomes

$$u_i = t_i - v_i(S) \tag{12}$$

where t_i is the payment received, and $v_i(S) = h_i k\sigma\sqrt{S_{i+1} + T_i - S_i}$ is the expected inventory holding cost at stage i. (When $S_{i+1} + T_i - S_i < 0$ then $v_i(S) = 0$.)

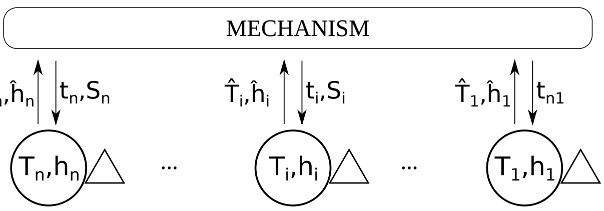

Figure 3. Mechanism design setting and information flow.

A VCG mechanism applied to the safety stock placement problem determines S^* as the solution of $P(\hat{T}, \hat{h}, s_1)$, and defines the payments in the form of

$$t_i = g_i(\hat{h}_{-i}, \hat{T}_{-i}) - \sum_{j \neq i} \hat{v}_j(S^*), \tag{13}$$

where $\hat{h}_{-i} = (\hat{h}_1, \ldots, \hat{h}_{i-1}, \hat{h}_{i+1}, \ldots, \hat{h}_n)$, $\hat{T}_{-i} = (\hat{T}_1, \ldots, \hat{T}_{i-1}, \hat{T}_{i+1}, \ldots, \hat{T}_n)$, g_i is an arbitrary function independent from $\hat{h}_i$ and $\hat{T}_i$, and $\hat{v}_i(S) = \hat{h}_i k\sigma\sqrt{S_{i+1} + \hat{T}_i - S_i}$.

It is well known, that VCG mechanisms are *truthful*, consequently, the stages can optimise their utility by disclosing $\hat{h}_i = h_i$ and $\hat{T}_i = T_i$. Furthermore it is *efficient*, viz., it minimises the total holding cost in the chain. The functions g_i give the freedom for constructing different mechanisms, e.g., $g_i \equiv 0$ results in a situation, where each stage must pay the total cost of the chain minus its own. In the next subsections we examine different VCG mechanisms and their properties.

The general idea that we use for developing specific VCG mechanisms is the following. We change $\hat{h}_i$ and $\hat{T}_i$ values in $\hat{h}$ and $\hat{T}$ for a predetermined $\tilde{h}_i$ and $\tilde{T}_i$, thus the resulting vectors denoted as $\hat{h}^{(i)}$ and $\hat{T}^{(i)}$ will not depend on $\hat{h}_i$ and $\hat{T}_i$. Then we calculate an optimal $S^{(i)}$ solution for the program $P(\hat{T}^{(i)}, \hat{h}^{(i)}, s_1)$, and define the g_i function as

$$g_i(\hat{h}_{-i}, \hat{T}_{-i}) = \sum_{j \neq i} \hat{v}_j(S^{(i)}). \tag{14}$$

Let $\tilde{v}_i(S) = \tilde{h}_i k\sigma\sqrt{S_{i+1} + \tilde{T}_i - S_i}$ denote the expected inventory holding cost function in the modified problem for stage i. The next theorem characterises the payment of any such mechanism.

Theorem 1 $\tilde{v}_i(S^*) - \tilde{v}_i(S^{(i)}) \geq t_i \geq \hat{v}_i(S^*) - \hat{v}_i(S^{(i)})$

Proof Since S^* minimises the objective function of $P(\hat{T}, \hat{h}, s_1)$

$$\hat{v}_i(S^*) + \sum_{j \neq i} \hat{v}_j(S^*) \leq \hat{v}_i(S^{(i)}) + \sum_{j \neq i} \hat{v}_j(S^{(i)}), \tag{15}$$

which can be rearranged resulting $t_i \geq \hat{v}_i(S^*) - \hat{v}_i(S^{(i)})$.

On the other hand, $S^{(i)}$ minimises the objective function of $P(\hat{T}^{(i)}, \hat{h}^{(i)}, s_1)$, therefore

$$\tilde{v}_i(S^{(i)}) + \sum_{j \neq i} \hat{v}_j(S^{(i)}) \leq \tilde{v}_i(S^*) + \sum_{j \neq i} \hat{v}_j(S^*), \tag{16}$$

thus we get $\tilde{v}_i(S^*) - \tilde{v}_i(S^{(i)}) \geq t_i$. □

The theorem provides an upper and a lower bound on the payments, which helps to characterise the expected utility of the stages as well as the budget of the mechanism (the total payment). Note that in some of the mechanisms we apply infinite $\tilde{h}_i$ modified holding cost, in which cases the upper bound also becomes infinite, thus fails to provide any useful information about the possible overpayment. However, due to the definition, the payments are always finite.

3.2.1 *Commonly known lead-times* ($\mathcal{M}_1$)

Firstly, we consider the situation where only the holding cost (h_i) is private information at stage i, and the T_i values are common knowledge. Although this contradicts our original assumptions, we decided to include this case for providing a comparison with the further mechanisms.

We use $\tilde{h}_i = \infty$ and the commonly known T in $P(T, \hat{h}^{(i)}, s_1)$. We further assume that $s_1 \geq T_1$, otherwise the stage 1 would have to keep some safety stock in order to guarantee s_1 service time, and then any feasible solution would be optimal—with infinite total cost. Since in the modified program the holding cost at stage i is infinite, in the optimal $S^{(i)}$ solution stage i will not carry any safety stock, i.e.,

$S_i^{(i)} = S_{i+1}^{(i)} + T_i$. This mechanism is analogous to the shortest path mechanism [13]: stage i receives as large payment as its contribution to the cost decrease of other stages.

The next theorem states, that if a stage does not hold inventory, it also does not receive any payment, otherwise it gets compensation which is not less than its cost.

Theorem 2 *If $S_i^* = S_{i+1}^* + T_i$ then $t_i = 0$. Else $S_i^* = \underline{S}_i$ and then $t_i \geq \hat{v}_i(S^*)$.*

Proof Since $S_i^{(i)} = S_{i+1}^{(i)} + T_i$, Theorem 1 in this case means $\tilde{v}_i(S^*) \geq t_i \geq \hat{v}_i(S^*)$, from which the statement follows. □

An immediate corollary of the theorem is that $\forall i : u_i \geq 0$, which property is called *individual rationality*. Furthermore, the mechanism has deficit, i.e., $\sum_{i=1}^n t_i \geq 0$.

Unfortunately, if T_i is private information this mechanism cannot be applied, since $S^{(i)}$ would then depend on $\hat{T}_i$, which corrupts the properties of the VCG mechanism. Therefore, in the following subsections we construct different mechanisms for the case when T_i is private information.

3.2.2 *Disregarding holding costs ($\mathcal{M}_2$)*

In this mechanism we use $\tilde{h}_i = 0$ and an arbitrary $\tilde{T}_i$. Since in this modified problem the inventory holding is free in stage i, it will keep as much safety stock, as possible ($S_i^{(i)} = \underline{S}_i^{(i)}$). Furthermore, none of the upstream stages holds any stock ($S_j^{(i)} = S_{j+1}^{(i)} + \hat{T}_j$, $(j = i+1, \ldots, n)$), and therefore $S_{i+1}^{(i)} = \sum_{j=i+1}^n \hat{T}_j$. It can be seen that the optimal $S^{(i)}$ is indeed independent from the value of $\tilde{T}_i$. This mechanism corresponds to the *Clarke pivot rule*, and the following theorem characterises its properties. The theorem follows from Theorem 1 using $\tilde{T}_i = 0$.

Theorem 3 $0 \geq t_i \geq \hat{v}_i(S^*) - \hat{h}_i k \sigma \sqrt{\sum_{j=i+1}^n \hat{T}_j - \underline{S}_i^{(i)}}$

This mechanism has surplus, i.e., $\sum_{i=1}^n t_i \leq 0$. This approach can be interpreted as comparing the optimal S^* solution to $S^{(i)}$, where stage i holds maximal inventory. It can be seen that this is unfair to the lower stages, where the possible maximal inventory is larger.

3.2.3 *Disregarding lead-times ($\mathcal{M}_3$)*

The next mechanism is constructed by using $\tilde{h}_i = \infty$ and $\tilde{T}_i = 0$. In the optimal $S^{(i)}$ stage i will not hold any stock, and therefore $S_i^{(i)} = S_{i+1}^{(i)}$. The payment in this case can be characterised by the following theorem (corollary of Theorem 1).

Theorem 4 *If $S_i^* = S_{i+1}^* + \hat{T}_i$ then $0 \geq t_i \geq -\hat{h}_i k \sigma \sqrt{\hat{T}_i}$. Else if $S_i^* = \underline{S}_i$ then $t_i \geq \hat{v}_i(S^*) - \hat{h}_i k \sigma \sqrt{\hat{T}_i}$. In the special case when $S_i^* = \underline{S}_i = 0$ then $t_i \geq 0$.*

This mechanism can work either with surplus or deficit, thus it can be viewed as a transition between the previous two mechanisms.

3.2.4 *Considering average lead-times ($\mathcal{M}_4$)*

In this subsection, we try to approximate the behaviour of the mechanism $\mathcal{M}_1$ by defining $\tilde{h}_i = \infty$ and $\tilde{T}_i = \sum_{j \neq i} \hat{T}_j/(n-1)$, i.e., the mean lead-time of the other stages. If we assume that $s_1 \geq \tilde{T}_1$, then the optimal $S^{(i)}$ solution satisfies $S_i^{(i)} = S_{i+1}^{(i)} + \tilde{T}_i$, wherewith Theorem 1 is reduced to the following form.

Theorem 5 *When the lead-time of stage i is below or equal to the average ($\hat{T}_i \leq \tilde{T}_i$), then $t_i \geq \hat{v}_i(S^*)$.*

Otherwise, when the lead-time is above or equal to the average, and $S_i^* = S_{i+1}^* + \hat{T}_i$ then $0 \geq t_i \geq -\hat{h}_i k \sigma \sqrt{\hat{T}_i - \tilde{T}_i}$, else $t_i \geq \hat{v}_i(S^*) - \hat{h}_i k \sigma \sqrt{\hat{T}_i - \tilde{T}_i}$.

The corollary of the theorem is that the stages are interested in decreasing their lead-times, since decreasing it below the average guarantees non-negative utility.

3.2.5 *Summary of the mechanisms*

In the next table we summarise the construction of the previous four mechanisms.

Table 2. Summary of the mechanisms

	$\mathcal{M}_1$	$\mathcal{M}_2$	$\mathcal{M}_3$	$\mathcal{M}_4$
$\tilde{h}_i$	∞	0	∞	∞
$\tilde{T}_i$	T_i	*	0	$\text{avg}_{j \neq i} T_j$

3.3 Decentralised protocol

In order to implement a mechanism in a decentralised way, i.e., without a trusted centre, two issues should be addressed: (i) the computation of the optimal safety stock placements, and (ii) assuring the payment for each stage. The first issue can be resolved by *replication*, which is a standard technique for implementing a VCG mechanism in a decentralised setting *faithfully*, i.e., in such a way, that the rational stages are not interested in deviating from the proposed protocol [7]. For example, if the stages disclose their private information to all of the other stages (so every stage knows $\hat{h}$ and $\hat{T}$), and then all of them can compute the optimal service times and payments. If they agree on the solution, they adopt it, otherwise they suffer a severe penalty, e.g., by missing the opportunity of serving the market.

Regarding the second issue, we suggest that the payments of the mechanism have to be covered from the market. Let us consider the situation presented on Fig. 4, where predefined unit prices p_i for the product and components are given.

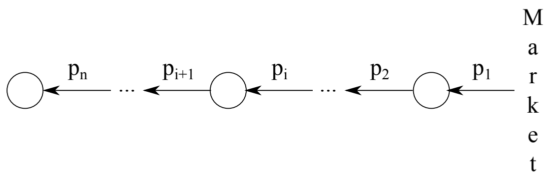

Figure 4. Decentralised implementation of the mechanism.

In order to assure the appropriate payment for the stages, the unit prices have to be modified. Note that we assume that the modification of p_1 does not influence the demand. This can be assumed if the modification is sufficiently small, therefore we prefer mechanisms that imply small change to p_1.

Let us define the new unit prices as $p_i' = p_i + \sum_{j=i}^n t_j/\mu$. With this modification, the expected utility of stage i—disregarding any production cost—in each period becomes

$$u_i = (p'_i - p'_{i+1})\mu - v_i(S^*) = (p_i - p_{i+1})\mu + t_i - v_i(S^*) \quad (17)$$

therefore the proposed decentralised implementation keeps the truthfulness and efficiency of the VCG mechanisms.

4 COMPUTATIONAL STUDY

4.1 Numerical example

Table 3 (page 6) illustrates the results of using the four different mechanisms in a supply chain with 10 stages. The parameters of the problem are $s_1 = 5$, $\mu = 1000$, $\sigma = 100$ and $k = 2.05$. The h_i, T_i and p_i parameters are indicated in the table. In the optimal case, stages 2 and 6 keep safety stock. In accord with the theorems, the mechanism $\mathcal{M}_1$ assigns payment only to those two stages, and the payment is not less than the expected holding cost. Mechanism $\mathcal{M}_2$ determines only negative payments, except for stage 6, which is the uppermost stage holding stock. The third and fourth mechanism assign both positive and negative payments as well; and in the latter case, the non-negative payment for stages with lead-time below the average (3.6) can also be observed. Note that we have disregarded the production costs in the model, which would only cause a constant shift in the utilities, therefore do not influence neither the optimal solution nor the payments. That is the reason of the unexpected increase of the utility at the uppermost stage.

Fig. 5 illustrates the same costs and the payments according to the different mechanisms at each stage graphically.

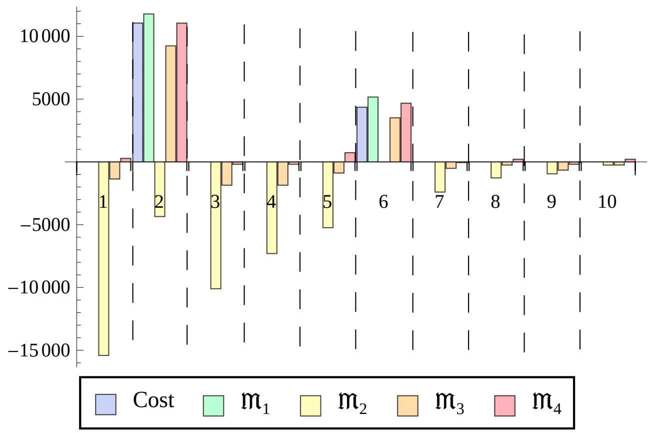

Figure 5. Illustration of the costs and payments at the different stages.

Table 4 shows the total payments which can be compared to the total holding cost of the optimal S^*. It can be seen that $\mathcal{M}_3$ resulted in a total payment closest to zero, and therefore it caused the smallest change in the market price by increasing it with only 2% .

4.2 Simulation

In order to check which mechanism results in the least change of the market price, we have run several experiments with different parameters. Table 5 shows the average results based on 500 simulation runs. The n, s_1, k, σ, μ and p parameters were the same as in the previous example, while the lead-times and holding costs were randomly

Table 4. Total payment and change in the market price

	$\mathcal{M}_1$	$\mathcal{M}_2$	$\mathcal{M}_3$	$\mathcal{M}_4$
$\sum v_i(S^*)$	15410			
$\sum t_i$	16944	-47276	5125	16526
p'_1/p_1	1.075	0.79	1.02	1.075

generated in each run, but using the same dataset for each mechanisms. The h_i values are from a uniform distribution with support $[n-i+1, 3(n-i+1)]$, which is reasoned with the observation that holding cost is likely to be higher downstream the supply chain. The T_i parameters were generated from an uniform distribution over $\{1, \ldots, 5\}$. The generating approach of the lead-times simulate various combinations of long manufacturing and short assembly operations, as well as long transportation times from global (e.g., Far Eastern) suppliers.

Table 5. Average performance of the mechanisms based on 500 runs.

	$\mathcal{M}_1$	$\mathcal{M}_2$	$\mathcal{M}_3$	$\mathcal{M}_4$
$\sum v_i(S^*)$	13535			
Avg $\sum t_i$	17773	-48628	2416	18133
Avg p'_1/p_1	1.036	0.901	1.005	1.037

It can be seen that mechanism $\mathcal{M}_1$ results in relatively high total payment. A corollary of Theorem 2 is that the total payment can not be less than the total cost. However, no upper bound was given for the payment, and thus the problem of overpayment may occur, similarly to the case of the shortest path mechanism [5]. The mechanism $\mathcal{M}_4$ approximates the first mechanism by using an average lead-time instead of the real one in the payment calculations, and therefore they resulted in similar behaviour. The $\mathcal{M}_2$, which allows only non-positive payments, results in an enormous negative payment, which is more than three times bigger than the inventory holding cost itself. The decentralised protocol in this case results in approximately 10% decrease in the market price, which—assuming price-independent demand—is clearly not desirable for the supply chain. Finally, $\mathcal{M}_3$ resulted in a fairly low total payment, and its indicated increase in the market price was only 0.5%.

5 CONCLUSIONS AND FUTURE WORK

We investigated the safety stock placement problem in non-cooperative serial supply chains, motivated mainly by the inventory management problems of global automotive supply networks. We applied mechanism design theory combined with the appropriate operation research models for minimising the overall inventory holding cost. We presented and compared four specific mechanisms based on the VCG scheme, and examined their distributed implementation.

There are several possible extensions of this work. Besides the presented mechanisms several others are possible, including randomised ones that may have more desirable properties. Providing upper bounds on the total payment, proving *approximate budget-balance*, is also an important research direction. Considering price-dependent demand leads to a more complex, but more realistic inventory holding and pricing problem. *Group-strategyproofness*, i.e., preventing collusions in possible coalitions, would also worth further investigations. A distributed implementation with partial information sharing, where the agents do not share complete private information with every other agents would make the model much more practical.

Table 3. Numerical example.

	i	1	2	3	4	5	6	7	8	9	10
	h_i	16	15	24	10	10	5	8	4	2	1
	T_i	3	5	4	4	2	5	4	2	5	2
	S_i^*	5	2	10	6	2	0	13	9	7	2
	$v_i(S^*)$	0	11056	0	0	0	4354	0	0	0	0
	p_i	223	159	114	81	58	42	30	21	15	11
$\mathcal{M}_1$	t_i	0	11778	0	0	0	5166	0	0	0	0
	$t_i - v_i(S^*)$	0	722	0	0	0	812	0	0	0	0
	p_i'	240	176	119	87	63	47	30	21	15	11
	u_i	63780	46279	32541	23243	16602	12671	8471	6050	4322	10804
$\mathcal{M}_2$	t_i	-15410	-4354	-10099	-7297	-5240	0	-2401	-1275	-950	-249
	$t_i - v_i(S^*)$	-15410	-15410	-10099	-7297	-5240	-4354	-2401	-1275	-950	-249
	p_i'	176	128	86	64	48	37	25	19	14	11
	u_i	48370	30147	22442	15946	11362	7505	6070	4775	3371	10555
$\mathcal{M}_3$	t_i	-1359	9239	-1857	-1857	-886	3510	-514	-249	-654	-249
	$t_i - v_i(S^*)$	-1359	-1816	-1857	-1857	-886	-844	-514	-249	-654	-249
	p_i'	228	166	111	80	59	43	28	20	14	11
	u_i	62421	43741	30684	21387	15716	11015	7956	5801	3668	10555
$\mathcal{M}_4$	t_i	280	11051	-191	-191	732	4671	-54	210	-192	210
	$t_i - v_i(S^*)$	280	-5	-191	-191	732	317	-54	210	-192	210
	p_i'	240	176	119	87	64	46	30	21	15	11
	u_i	64060	45552	32350	23053	17334	12176	8417	6260	4129	11014

We emphasise that combining planning models with the results of the algorithmic mechanism design can be applied to different logistic problems; the model presented in the paper is only one example. Therefore considering more complex planning problems is also a possible future working field.

ACKNOWLEDGEMENTS

This work has been supported by the OMFB No. 01638/2009 grant and the János Bolyai scholarship No. BO/00659/11/6. The author also thank József Váncza for his help and support.

REFERENCES

[1] G.P. Cachon and M. Olivares, 'Drivers of finished-goods inventory in the U.S. automobile industry', *Management Science*, **56**(1), 202–216, (2010).

[2] M. de Weerdt, Y. Zhang, and T. Klos, 'Multiagent task allocation in social networks', *Journal of Autonomous Agents and Multi-Agent Systems*, **25**(1), 46–86, (2012).

[3] P. Egri, A. Döring, T. Timm, and J. Váncza, 'Collaborative planning with benefit balancing in dynamic supply loops', *CIRP Journal of Manufacturing Science and Technology*, **4**(3), 226–233, (2011).

[4] P. Egri and J. Váncza, 'Supply network coordination by vendor managed inventory – a mechanism design approach', in *Proc. of the 2nd Workshop on Artificial Intelligence and Logistics (AILog-2011)*, pp. 19–24, (2011).

[5] E. Elkind, A. Sahai, and K. Steiglitz, 'Frugality in path auctions', in *Proc. of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 701–709, (2004).

[6] *From build-to-order to customize-to-order: Advancing the automotive industry by collaboration and modularity*, eds., R. Ericsson, R. Becker, A. Döring, H. Eckstein, Th. Kopp, I. Poslu, and J. Váncza, The AC/DC Consortium, ISBN 978-91-633-6973-5, 2010. Code of Practice Findings of the EU-FP6 Project AC/DC – Automotive Chassis Development for 5-Days Cars.

[7] J. Feigenbaum, M. Schapira, and S. Shenker, 'Distributed algorithmic mechanism design', in *Algorithmic game theory*, eds., N. Nisan, T. Roughgarden, E. Tardos, and V.V. Vazirani, Cambridge University Press, (2007).

[8] M. Feldmann and S. Müller, 'An incentive scheme for true information providing in supply chains', *Omega*, **31**, 63–73, (2003).

[9] D. Garg, Y. Narahari, E. Foster, D.M. Kulkarni, and J.D. Tew, 'A Groves mechanism approach to decentralized design of supply chains', in *Proc. of the IEEE Conference on Electronic Commerce*, pp. 330–337, (2005).

[10] S.C. Graves and S.P. Willems, 'Strategic safety stock placement in supply chains', in *Proc. of the 1996 MSOM Conference*, Hanover, NH.

[11] S.C. Graves and S.P. Willems, 'Supply chain design: Safety stock placement and supply chain configuration', in *Supply chain management: Design, coordination and cooperation*, eds., A.G. de Kok and S.C. Graves, volume 11 of *Handbooks in Operations Research and Management Science*, 229–339, Elsevier, (2003).

[12] W. Menzel, J. Lentes, A. Döring, R. Ericsson, and L. Siljemyr, 'Dynamic supply loops – a concept for flexible and faster automotive supply network management', in *Advanced Manufacturing and Sustainable Logistics*, volume 46 of *Lecture Notes in Business Information Processing*, 130–140, (2010).

[13] N. Nisan and A. Ronen, 'Algorithmic mechanism design', *Games and Economic Behavior*, **35**(1-2), 166–196, (2001).

[14] D.C. Parkes and J. Shneidman, 'Distributed implementations of Vickrey-Clarke-Groves mechanisms', in *Proc. of 3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 261–268, (2004).

[15] J. Shneidman and D.C. Parkes, 'Specification faithfulness in networks with rational nodes', in *Proc. of the 23rd Annual ACM Symposium on Principles of Distributed Computing*, pp. 88–97, (2004).

[16] D. Simchi-Levi, *Operation Rules – Delivering Customer Value through Flexible Operations*, MIT Press, 2010.

[17] D. Simchi-Levi, P. Kaminsky, and E. Simchi-Levi, *Designing and managing the supply chain: Concepts, strategies, and cases*, McGraw – Hill, New York, 2000.

[18] D. Simchi-Levi and Y. Zhao, 'Performance evaluation of stochastic multi-echelon inventory systems: A survey', *Advances in Operations Research*, **2012**, 34 pages, (2012). Article ID 126254, doi:10.1155/2012/126254.

[19] K.F. Simpson, 'In-process inventories', *Operations Research*, **6**, 863–873, (1958).

[20] R. van der Krogt, M. de Weerdt, and Y. Zhang, 'Of mechanism design and multiagent planning', in *Proceeding of the 18th European Conference on Artificial Intelligence (ECAI 2008)*, pp. 423–427, (2008).