Inventory Management and the Impact of Anticipation in Evolutionary Stochastic Online Dynamic Optimization

P. A. N. Bosman and H. La Poutré

Abstract—Inventory management (IM) is an important area in logistics. The goal is to manage the inventory of a vendor as efficiently as possible. Its practical relevance also makes it an important real–world application for research in optimization. Because inventory must be managed over time, IM optimization problems are dynamic and online (i.e. they must be solved as time goes by). Dynamic optimization is typically harder than non–dynamic optimization. Much research in IM is devoted to finding specific algorithms that solve specific abstractions. For each new aspect to be taken into account, a new algorithm must be designed. In this paper, we aim at a more general approach. We employ general insights into online dynamic problem solving. A recently proposed framework is also employed. We point out how IM problems can be solved in a much more general fashion using evolutionary algorithms (EAs). Here, time–dependence (i.e. decisions taken now have consequences in the future) is an important practical type of problem difficulty that is characteristic of practical dynamic optimization problems. Time–dependence is usually not taken into account in the literature and myopic (i.e. blind to future events) algorithms are often designed. We show that time–dependence is automatically tackled by our novel approach. We extend the common definition of IM problems with time–dependence by introducing customer satisfaction. We show that customer satisfaction for IM problems with superior solutions can be achieved when this form of time–dependence is properly taken into account. This also demonstrates our conclusion that taking into account the existence of time–dependence in practical online dynamic optimization problems such as IM is very important.

I. INTRODUCTION

Logistics, i.e. the area of planning, managing and moving goods, is important to the world in general. Optimization problems or subareas that are characteristic of logistics are therefore important real–world applications of optimization algorithms. Inventory Management [15], [16] (IM) is one of such important subareas in logistics and is often seen as an important part in supply chain optimization [1], [10], [12]. The goal in IM is to manage the inventory of a vendor as efficiently as possible. The vendor sells goods to customers. To make the most profit, the vendor must ensure that the inventory is filled enough to meet with the demand of the customers without overstocking. As it typically takes some time before a replenishment order arrives from the vendor’s suppliers, the vendor needs to plan ahead when deciding upon when to order new goods in which amounts and from which suppliers.

Like many other optimization problems in logistics, IM is a dynamic optimization problem. Such problems change with time. Since the dynamic changes are typically unknown beforehand the problem generally has to be solved online, i.e. as time goes by [2], [9], [13], [17]. Because the problems to be solved, even for a single point in time, are often hard and there is typically not much time between two subsequent decision times, restarting optimization from scratch is often undesirable. The tracking of (near–) optima, once they have been found, is therefore desirable. To be able to do this, the optimization algorithm needs to have a proper degree of adaptivity. Evolutionary algorithms (EAs) [11] are good candidates as they employ a set of solutions rather than a single solution. Adaptivity is then a virtue of issues such as maintaining diversity around (sub)optima and continuously searching for new regions of interest that may appear over time [7].

The field of online optimization itself is relatively young [17]. Classically, many problems are not modelled as being dynamic. Although the problems to be solved are already very hard to solve without the dynamism and the necessity of solving the problem online, in practice these issues are very important and cannot be disregarded, especially when one realizes the issue of time–dependence (i.e. decisions taken now have consequences in the future). To solve dynamic optimization problems online it is straightforward to take a myopic, i.e. “near-sighted”; approach, which is often indeed taken. The quality of a decision is then taken only to be how good it is in the current situation. This approach however is blind to the important issue of time–dependence: decisions taken now have consequences in the future. Myopic decisions may however lead to a future in which lower profits can be obtained as a result of time–dependence in the problem at hand. A seemingly suboptimal solution for the current situation, which typically corresponds to actions like making investments, may on the other hand create a future in which much higher rewards can be obtained. Integrated over the entire time–span, this suboptimal decision for the current situation may thus very well be the better choice. To properly tackle time–dependence, anticipation of future situations is needed to be able to make well–informed decisions [3], [5], [8], [18].

The difference between the optimum when using the myopic approach compared to using anticipation can be arbitrarily big [3]. This has been demonstrated however only on an artificial moving–peaks problem. In this paper we show how time–dependence plays an important role in practice as well. Here we focus specifically on inventory
management problems. We take into account how the level of customer satisfaction influences future buying behavior of customers. Not only do we point out how this is a source of time–dependence, we also use a novel approach to tackle time–dependence properly, following a framework for solving online dynamic optimization problems with time–dependence that we recently proposed [6]. The results show that if time–dependence is properly taken into account, the profits can be much larger because much more effective inventory–management strategies are selected by the underlying EA. This demonstrates our main conclusion that taking into account the existence of time–dependence in practical online dynamic optimization problems such as IM is very important.

The remainder of this paper is organized as follows. In Section II we formulate IM as an online dynamic optimization problem. We also show how customer satisfaction can be modeled and how it introduces an important source of time–dependence. In Section III we design an online dynamic EA for solving IM problems. We use this EA in Section IV to solve a selection of IM problems. We conclude this paper with a summary and an outlook in Section V.

II. INVENTORY MANAGEMENT

A. General description

In this section we describe a commonly used definition of IM problems [16]. In Figure 1 a schematic overview is given of IM problems. A general description of IM is the following. Buyers, also called customers and denoted $C_i$, buy goods from a vendor. The number of buyers is denoted $n_c$. The number of goods and the frequency of buying is called the demand and is denoted $D_i$ for customer $C_i$. To prevent going out of stock, the store keeps an inventory. This inventory must however be replenished from time to time. Because the delivery of new stock from the store’s suppliers also takes time (called the lead time, denoted $L_j$ for supplier $S_j$), the replenishment–order must be placed before going out of stock. The number of suppliers is denoted $n_s$.

$$\max_{\mathbf{z}(t)} M(0) + \left\{ \sum_{t=0}^{t_{end}} \Delta M(t) \right\}$$

where $\Delta M(t)$ is the change in the money of the vendor at time $t$, $M(t)$ depends on the decisions $\mathbf{z}(t)$ and $t_{end}$ is the end of the planning horizon. The expenses of the vendor are in the costs of holding the inventory and in ordering new supplies. The income of the vendor is based on sales made:

$$\Delta M(t) = Sales(t) - HoldingCost(t) - OrderingCost(t)$$

The holding cost can be computed per time–unit and depends on the size of the inventory $I(t)$. Typically, holding costs are a fixed price $p^H$ per unit of inventory per time–unit:

$$HoldingCost(t) = p^H I(t)$$

Typically, the cost of an order that is placed at some supplier is paid for when the order is delivered. The ordering cost at time $t$ therefore depends on earlier decisions $x(t')$, $t' < t$. The cost of a replenishment–order at supplier $S_i$ typically consists of two parts: a fixed part $c^O_i$ and a part that increases with the size of the order. Typically, a fixed price $p^O_i$ per unit of ordered goods is charged. Let $x^O_i(t')$ be the quantity of goods ordered from supplier $i$ at time $t$ and let $L_i(x^O_i(t'))$ be the time it takes supplier $S_i$ to deliver that order, then we have:

$$OrderingCost(t) = \sum_{i=0}^{n_s-1} \sum_{t' \leq t} o(i,t',t)$$

where

$$o(i,t',t) = \begin{cases} c^O_i + p^O_i x^O_i(t') & \text{if } x^O_i(t') > 0 \\ \text{and } t' + L_i(x^O_i(t')) = t & \text{otherwise} \\ 0 & \text{otherwise} \end{cases}$$

Note that in an implementation, it is not efficient to use equation 4 directly. Instead, an event queue can be used to check whether a new supplier order has arrived at time $t$ and hence whether payment is due.

The income of sales at time $t$ depends on whether the demand of a buyer at time $t$ can be met. Only if there are enough goods in inventory, a sale is made. Partial sales are typically excluded [16]. Typically, the goods are sold at a fixed unit price $p^S_i$. Let $D(t)$ be the demand of the buyers at time $t$, then we have:

$$Sales(t) = \begin{cases} p^S D(t) & \text{if } D(t) \leq I(t) \\ 0 & \text{otherwise} \end{cases}$$

We note that there are many other ways in which the above costs and gains can be computed, but most are just minor variations of the equations presented above. Also, other costs and gains besides the ones mentioned above may be taken.
into account such as the directly computable cost of having more demand than inventory (called excess demand cost) and the subsequent possible loss of sales (called lost-sales cost). Such functions are similar to the ones above. Although they indeed make the model more involved, it doesn’t increase the level of problem difficulty, especially for a general problem solving approach that we shall use for this type of problem (see Section III).

Although now $\Delta M(t)$ has been defined, there are still functions in the above definitions that are left undefined, namely $I(t)$, $L_i$ and $D(t)$. The inventory (or stock) level $I(t)$ usually is assumed to have the property $I(0) = 0$, but this is not a requirement. In addition, the inventory level rises with $x_i^{\text{OrderQuantity}}(t')$ at time $t' + L_i(x_i^{\text{OrderQuantity}}(t'))$ (i.e. when the supplier order arrives). The inventory level falls with $D(t)$ when a sale is made at time $t$. The formulations for the rise and fall of $I(t)$ are very similar to those of $\Delta M(t)$. Function $L_i$ for the delivery time for an order placed at supplier $i$ typically follows a certain probability distribution. The same holds for the demand function $D(t)$. The demand function has two sources of stochasticity however: both the quantity $Dq$ of the demand and the frequency of the demand $Df$. In this paper we shall assume all these functions to be normally distributed at any time $t$ with means of respectively $\mu^t(t)$, $\mu^{\text{IM}}(t)$ and $\mu^{Df}(t)$ and variances of respectively $\sigma^{2,L}(t)$, $\sigma^{2,Dq}(t)$ and $\sigma^{2,Df}(t)$ where $\mu^{Df}(t)$ and $\sigma^{2,Df}(t)$ describe the distribution of the time between two subsequent customer visits. Note that all these distributions are restricted to $\mathbb{R}^+$. Time–dependence plays an important role even in the base definition of IM. The decision of whether or not to place a replenishment–order at a certain point in time has great future consequences because it determines future inventory levels. Also, a decision to place an order at a supplier leads to a future event (delivery of goods) that is a response to placing the order. Still, although time–dependence is already important here, it is of a rather trivial nature because the only function that is affected by decisions made earlier is the level of the inventory. Given a fixed demand, a single supplier, and a fixed lead time for that supplier, the best strategy for placing replenishment orders is straightforward to compute [16]. Although this is already no longer possible if there is more than one supplier [15], an extension of the model that defines a second, but also very practically relevant, level of time–dependence, makes the problem even harder.

C. Extended Definition: Customer Satisfaction

Customer satisfaction is important in doing business with customers. A higher level of customer satisfaction will most likely result in a growing frequency of consumer transactions, either from the same consumer or new consumers as satisfied customers will spread the word. In our model this means that whether or not a customer is satisfied when requesting goods from the vendor influences the stochastic model that underlies the customer demand behavior. We can integrate this in the above model by changing the parameters that describe the distribution of the time between two subsequent customer visits, $\mu^{Df}(t)$ and $\sigma^{2,Df}(t)$. If a sale can be made (there is enough inventory, see Equation 5), the customer frequency increases and thus the time between two subsequent customer visits decreases. If the customer cannot be satisfied (no sale is made), the frequency decreases:

$$\mu^{Df}(t + 1) = \max \{1, C(t)\mu^{Df}(t)\}$$

$$\sigma^{2,Df}(t + 1) = \max \{1, C(t)\sigma^{2,Df}(t)\}$$

$$C(t) = \begin{cases} 
\frac{1}{2} & \text{if } D(t) \leq I(t) \text{ and } D(t) > 0 \\
2 & \text{if } D(t) > I(t) \text{ and } D(t) > 0 \\
1 & \text{otherwise } (D(t) = 0) 
\end{cases}$$

The influence of customer satisfaction now is one of true time–dependence, which can be seen as follows. A decision whether or not to order new goods from a supplier has a future effect on the size of the inventory. Depending on the level of the inventory, a sale can either be in the future made or not. Although this is already a form of time–dependence, customer satisfaction brings a secondary level of time–dependence. Whether or not a future sale can be made influences further future sales indirectly because of an altered customer frequency. This secondary effect is very important however, because it determines the rate of change in future profits. If a higher frequency of customer visits can be met as a result of proper inventory management, profits can be made faster. However, this type of time–dependent influence is often overlooked in literature [1]. Without foreseeing the future effects on the frequency of customer visits, the strategy optimizer will see no need to change the strategy beyond one that meets with the current expected rate of customer frequency. Although this doesn’t mean that profits become losses, the strategy can clearly not be optimal.

III. EA DESIGN

Many models exist for simple to hard problems in IM. For the simplest problems, exact optimal strategies are known. For practical problems however, there are typically multiple store–suppliers to choose from with different delivery times, order magnitudes and costs. Also demands and lead–times tend to be stochastic rather than deterministic. Although these aspects increase the benefit of a non–myopic view, they also make the problem harder to solve [15]. Consequently, no exact optimal strategies exist for the harder and more general cases. For specific cases, specific heuristics exist. There is no general flexible approach however that is applicable to a variety of IM problems. Here we follow a recently proposed approach that is a promising candidate.

A. Framework Issues

Recently, a framework for solving online dynamic optimization problems based on EAs was proposed [6]. This framework was designed to take into account time–dependence. The main motivation for using EAs is that the problem at hand without the dynamism is often already hard and we don’t have time to solve the problem to
optimality with traditional techniques. Also, there is typically not much time between two subsequent decision times. For these reasons, restarting optimization from scratch is often undesirable. The tracking of (near-) optima, once they have been found, is therefore desirable. To be able to do this, the optimization algorithm needs to have a proper degree of adaptivity. This again motivates the use of EAs since they are good candidates for dynamic optimization in general as a result of employing a set of solutions rather than a single solution. Adaptivity is then a virtue of issues such as maintaining diversity around (sub)optima and continuously searching for new regions of interest that may appear over time [7].

The framework itself much follows the definition of online dynamic optimization problems while taking into account time–dependence. Let \( \mathbf{x} \) denote the variables to be optimized. We will often refer to choosing a configuration for these variables as taking a decision in online dynamic optimization. Mathematically defined, dynamic optimization is to optimize a functional

\[
\max_{\mathbf{w}(t)} \left\{ \int_{t^\text{now}}^{t^\text{end}} \mathcal{S}_{\gamma(t)}(\mathbf{x}(t)) \, dt \right\}
\]

where \( \mathcal{S} \) is a function of \( \mathbf{x} \) and has dynamically changing parameters \( \gamma \). Note that in the discrete or discretized case, the integral in Equation 8 is replaced by a discrete sum, similar to our case of IM problems as introduced in the previous section. Function \( \mathcal{S} \) can be seen as the real world. Solving this problem online means that at any point in time \( t^\text{now} \), function \( \mathcal{S} \) cannot be evaluated for any \( t > t^\text{now} \).

Time–dependence now means that \( \gamma(t^\text{now}) \) may depend on previous decisions \( \mathbf{x}(t), t < t^\text{now} \). If only the current situation is taken into account, the decision that immediately leads to the highest reward is optimal. To prevent bad decisions due to the use of a myopic approach, the decision for the current situation needs to be regarded simultaneously with future decisions in (near) future situations. Mathematically, this amounts to solving:

\[
\max_{\mathbf{w}(t)} \left\{ \int_{t^\text{now}}^{t^\text{end}} \mathcal{S}_{\gamma(t)}(\mathbf{x}(t)) \, dt \right\}
\]

Theoretical, this approach is clearly optimal because of the similarity with Equation 8. The problem is of course that this approach involves evaluating function \( \mathcal{S} \) for times beyond the current time, which is not possible in online optimization. The only way we can still take into account the future is to learn to predict it and use the predicted future instead. In dynamic optimization, this is called anticipation.

The nice thing about anticipation is that under the condition of perfect prediction and the possibility of finding the optimum with the optimization algorithm of choice, optimal decisions can be taken. Summarizing, the approach is to build and maintain (i.e. learn online) an approximation of \( \mathcal{S}_{\gamma(t)} \) and to optimize decisions for the present and for the approximated future simultaneously. In practice, it is typically not possible to optimize the future indefinitely, so only part of the approximated future will be optimized:

\[
\max_{\mathbf{w}(t)} \left\{ \min_{t^\text{now} < t < t^\text{end}} \left\{ \int_{t^\text{now}}^{t} \mathcal{S}_{\alpha}(t, \mathbf{x}(t)) \, dt \right\} \right\}
\]

where \( \alpha \) are the parameters that pertain to the function class which we use to learn approximation \( \mathcal{S} \). In logistics settings, typical examples of \( \alpha \) include the rate of customer demand or the parameter describing the distribution of customer demand. Function \( \mathcal{S} \) can be seen as a simulation of the real world. Note that an apparent drawback is that if the consequences of decisions extend beyond the prediction length \( t^\text{end} \), optimal decisions can no longer be guaranteed.

The framework describes how to solve Equation 10. A first version of the framework already described the explicit modeling of anticipation for online dynamic optimization, but only for non–stochastic problems [3]. Real–world online dynamic optimization problems are however often stochastic. In IM for instance, the customer visits to the vendor typically follow a certain probability distribution as do the number of goods they want to buy and the lead times for the suppliers. In the stochastic case, it is often better to optimize a strategy than to optimize the decisions to be made directly [6]. A strategy is basically a function that returns a decision for any given situation. The reason for this is that optimization is to be done over a future time interval (see Equation 10). Because the function to optimize over this future time interval is stochastic, multiple scenarios need to be considered which are all drawn from the probability distribution that causes the stochasticity. Because the optimal trajectory of decisions is likely to be quite different for each scenario, this amounts to performing optimization of a trajectory of decisions once for each scenario. Optimization is already generally considered to be hard for the type of problem that we are interested in. If the parameters of a strategy are optimized however, the strategy only needs to be evaluated for each scenario. The goodness of a strategy is the average profit obtained over the sampled scenarios.

In Figure 2 pseudo–code is given that summarizes the approach proposed above. The general idea is that in addition to a population, a current best strategy is maintained. This allows the EA to be run continuously. Whenever a decision needs to be made, the current best strategy can be applied. Evaluation of a strategy is done by running that strategy in the simulation multiple times using different random seeds. The quality of a strategy is then measured by its average evaluation value. The variance is also stored. The variance is required to compare the best strategy in the population with the current best strategy. Because multiple random seeds are used, corresponding to multiple drawings from the probability distribution that underlies the problem, statistical hypothesis tests are required to be certain that an improvement has been obtained. The statistical hypothesis
test that we used in our experiments is the Aspin–Welch–Satterthwaite (AWS) $T$–test at a significance level of $\alpha = 0.05$. The AWS $T$–test is a statistical hypothesis test for the equality of means in which the equality of variances is not assumed [14].

$$\begin{align*}
R & \leftarrow \text{INITIALIZEREALWORLD}() \\
S & \leftarrow \text{INITIALIZESIMULATION}() \\
\mathcal{P} & \leftarrow \text{INITIALIZEPOPULATION}() \\
s & \leftarrow \text{RANDOMSCENARIOSEEDS}() \\
\text{for } i \leftarrow 0 \text{ to } |\mathcal{P}| - 1 \text{ do} \\
\quad & \text{EVALUATE}(\mathcal{P}_i) \\
\quad & s_{\text{best}} \leftarrow \text{SELECTBESTSTRATEGY}(\mathcal{P}) \\
\text{while } t^\text{sim} < t^\text{end} \text{ do} \\
\quad & \text{EVOLVEPOPULATIONONEGENERATION}() \\
\quad & s \leftarrow \text{RANDOMSCENARIOSEEDS}() \\
\quad & \text{for } i \leftarrow 0 \text{ to } |\mathcal{P}| - 1 \text{ do} \\
\quad & \quad \text{EVALUATE}(s_{\text{best}}) \\
\quad & \quad \text{if } \text{SIGNIFICANTLYBETTER}(s_{\text{candidate}}, s_{\text{best}}) \text{ then} \\
\quad & \quad \quad s_{\text{best}} \leftarrow s_{\text{candidate}} \\
\text{end} \\
\text{end}
\end{align*}$$

Fig. 2. Outline of the algorithmic framework: the EA (top) and evaluating a strategy (bottom).

### B. Inventory–Management Issues

The EA follows the framework from Section III-A. It is common practice in IM to find a strategy instead of individual decisions. This fits perfectly into the framework. The strategy that we employ is a common one in IM. The strategy is a so–called $(s, Q)$ strategy [16]; $s$ is called the re–order point and $Q$ the order–up–to size. One such strategy is used for each supplier. Hence, the genotype contains $2n_s$ real values to be optimized, where $n_s$ is the number of suppliers. If the stock drops below the re–order point $s_i$ of supplier $i$, and no order is currently outstanding for supplier $i$, a new order is placed at supplier $i$ of size $Q – \text{stocklevel}$. Thus, in the case of two suppliers, if an order from the cheaper supplier is running late, the stock level will drop further and the rule for the more expensive, emergency supplier becomes active. It is not known whether this strategy can be optimal for problem II, but it is an often used, sensible choice.

### C. EA Issues

The parameter to be optimized by the EA are the parameters of the $(s, Q)$ strategies. Since these are real–values, we use a real–valued EA. Specifically, we use a recent, state–of–the–art real–valued EDA (estimation–of–distribution algorithm) called SDR–AVS–IDEA [4]. EDAs estimate the distribution of the selected solutions and samples new solutions from this estimated distribution in an attempt to perform variation more effectively. In SDR–AVS–IDEA normal distributions are estimated. SDR–AVS–IDEA additionally adaptively scales the estimated variances depending on whether improvements are found close to where the search is currently focusing. For this problem we did not learn any covariances in the normal distribution that is estimated for the various parameters to be optimized. This means that SDR–AVS–IDEA computes for each parameter the mean and variance in the selected set of solutions and resamples new solutions with these parameters using a separate one–dimensional normal distribution for each parameter to be optimized.

### IV. Experiments

#### A. Problems

We have designed four IM experiments. For all experiments, inventory is to be managed for 129600 minutes, i.e. 90 days. Orders can be placed any minute of the day.

1) **Problem I:** represents problems of the type for which an optimal strategy can be computed beforehand. There is one supplier and one product. Product quantities are integer. The product is sold to the buyers at a price of 50 and bought from the supplier at a price of 20. A fixed setup cost for each order placed at a supplier is charged at a price of 50. Inventory holding costs are 1 per day per unit. The lead time of the supplier is fixed to 3 days. The demand is fixed to an order of 1 item every hour.

2) **Problem II:** represents problems for which there is not a known optimal strategy. There are two suppliers. One supplier is cheaper than the other. The more expensive supplier can supply immediately, but costs twice as much. This type of setting is popular in IM research. It is typically known as IM with emergency replenishments and is known to be a hard problem [15]. The second supplier is used only if the stock has become really low and stock outs are imminent. To add to the difficulty of the problem, we have made the lead–time of the cheapest supplier both stochastic and periodically changing. The lead time of the slower supplier is normally distributed with mean (in minutes) of $4320(\cos(2\pi t)/43200) + 1)/2$, i.e. it varies between 0 and 3 days and the period–length of the cosine is 30 days. The variance is $14402(\cos(2\pi t)/43200) + 1)/2$, i.e. it varies between 0 and 1440 days, corresponding to a maximum standard deviation of 38 days with the same period–length as the mean. The periodically changing lead time causes the optimal strategy to change with time as well. In addition, the demand is now also stochastic. The time between two subsequent orders is normally distributed with a mean of one hour and a variance of 60 hours. The amount ordered is also normally distributed, with a mean of 3 products and a variance of 9 products. For this setting, there aren’t any known heuristics.

3) **Problem III:** is the same as Problem I but uses consumer satisfaction as defined in Equation 6.
4) Problem IV: is the same as Problem II but uses consumer satisfaction as defined in Equation 6.

B. Algorithmic setup

We used two different EA settings, a “small” setting and a “big” setting. The small setting corresponds to a situation in which there is only very little time to do optimization and thus the EA resources are small. The big setting corresponds to a situation in which there is more time and thus the EA resources are larger. In the small settings, the population size is 50, scenario–evaluation simulates 10 days into the future, 5 generations of the EA can be done per day, and 10 scenarios are used. In the big settings, all settings are three times bigger. The population size is 150, scenario–evaluation simulates 30 days into the future, 5 generations of the EA can be done every eight hours and 30 scenarios are used. To facilitate the simulation of the future, the EA learns the distribution behind the stochasticity of the buyer using maximum likelihood estimates. The stochasticity of the supplier is assumed to be known. All results were averaged over 100 independent runs.

C. Results

1) Problem I: In Figure 3 the average profit obtained is shown for both EA settings. The approach can be seen to be a scalable technique (also on Problem II) in the sense that allowing more resources results in better solutions. Investing in computing power thus results in a better policy for a vendor. Moreover, even the small settings for the EA lead to very good profits. The maximum profit that can be obtained on problem I is 58888. This profit corresponds to a setting of the strategy \((s, Q) = (143, 143)\) that is far outside the range in which we initialized the EA \((s, Q) \in [0, 25] \times [0, 50]\). Out of all settings in the initialization range, the maximum profit is only 25705. The EA is thus also capable of finding much better solutions when initialization is suboptimal. The big EA settings lead to near–optimal results.

![Fig. 3. Results on inventory–management problem I.](image1)

Figure 4 shows the strategies obtained with the big EA settings for both problems in a typical run of the EA. The lack of stochasticity in problem I translates into finding a stable strategy by the EA very quickly and maintaining that strategy throughout the run.

![Fig. 4. Strategies evolved in a typical run of the EA on problem I.](image2)

2) Problem II: In Figure 5 the average profit obtained is shown for both EA settings. The profits on problem II are higher than on problem I. The average demand in problem II is three times higher than in problem I. Indeed, the EA is able to obtain a profit of about 3 times higher than on problem I even though problem II is far more difficult.

![Fig. 5. Results on inventory–management problem II.](image3)

Figure 6 shows that the adaptive capacity of the EA allows the algorithm to continuously adapt the strategies and find better solutions for the situation at hand as the lead time of the cheapest supplier changes with time. The periodic change of the lead time of the cheapest supplier \((S_0)\) is clearly translated into a periodic change in strategy. When the average and variance of the lead time of the cheapest supplier are small, less products need to be ordered and the threshold can be lower. The threshold for emergency replenishments can even become 0. When the lead time is the largest, emergency replenishments may become necessary and concordantly, the EA proposes a strategy in which emergency replenishments are made before the stock runs out completely. Also, in this case the re–order point for the cheapest supplier is much higher as is the number of products ordered from that supplier. It can be seen in Figure 7 that emergency replenishments are indeed made during the periods when the cheapest supplier is less reliable. Furthermore, note that the strategies are not exactly the same in each period. Note that while the EA is optimizing strategies, it is also still learning distributions. Learning converges to the true distribution over time. Finally, the periodic change in the lead time of the cheapest supplier can also be seen back in the obtained profits in Figure 5. When the lead time of the cheapest supplier is the smallest, the EA succeeds in finding a strategy that uses this supplier more and therefore obtains more profit, resulting...
in a steeper slope of the profit–versus–time graph at these moments.

3) Problem III: From the results on problems I and II it is now clear that giving more resources to the EA improves the results. For this reason we now only continue our experiments with the small EA settings. Figure 8 shows the average profit that was obtained if the EA is run with anticipation and without anticipation. The difference between the results is very large. Without anticipation, the EA is still able to obtain a profit. With anticipation however, the EA picks up on the fact that having a larger inventory eventually leads to a higher frequency of sales and the result is a much faster growing profit. The growth is actually exponential until the customer frequency (one customer per minute) can no longer increase.

Inventory growth as a result of ordering more supplies (i.e. a change in strategy) can clearly be seen in Figure 9. The strategy moves far away from its initialization range and continues to raise the number of goods to order as time goes by until customer satisfaction can no longer be increased (which happens approximately at the end of the run).

4) Problem IV: A difference that is similar in magnitude can be seen in Figure 10 between the EA that uses anticipation and the EA that doesn’t use anticipation on problem IV. The main difference with the results on problem III (besides the effect due to the the periodic change in the supply time of the main supplier) is that the increase in the rate of profit is almost immediate. On problem III it takes some time before the strategy matches the maximum customer satisfaction level. The reason for this difference is the emergency supplier. The emergency supplier can supply goods immediately. Ordering more from the emergency supplier therefore has a much faster effect on the customer satisfaction level. In Figure 11 it can indeed be seen that the strategy for ordering from the emergency supplier is changed very quickly: an increase in the number of goods to be ordered can be observed. This increase allows to immediately meet with the quickly growing demand of the customers. Meanwhile, the strategy for ordering from the main supplier also increases the number of goods to order, making more profit in the end by meeting the maximum level of customer satisfaction mostly through the supplies bought from the main (and cheaper) supplier rather than the emergency supplier.

V. CONCLUSIONS AND OUTLOOK

In this paper we have focused on inventory management (IM) problems. This important real–world type of optimization problem is dynamic and in practice must be solved online, i.e. as time actually goes by. We have argued that in many online dynamic optimization problems time–
One of the main avenues of future research that we shall explore is the design of EAs for other specific online dynamic optimization problems. Also there are many important questions still to be answered about time–dependence in general such as whether the length of the time–interval that is required to use future predictions over can also be detected online. Concluding, online dynamic optimization and the use of anticipation represents an important avenue of research to which this paper makes an important contribution.

REFERENCES