A Comparative study of simulation based optimization methods in a continuous review (*s, S*) inventory control system

Amr B. Eltawil

Production Engg. Dept., Faculty of Engg., Alexandria University, 21544 Alexandria, Egypt
amreltawil@alex.edu.eg

This paper presents a comparison between two simulation based optimization methods in the context of a simulation model of a continuous review (*s, S*) inventory control system. The model is used to find the optimum settings for the reorder level, and the order up to level quantities, that minimize the total inventory costs. The inventory costs considered are ordering, holding with full backordering or shortages and lost sales costs. Two simulation optimization based methods were used to find the optimum settings; response surface methodology and metaheuristics based search. The comparison was done under several settings of demand rate, demand size, lead time, and shortage policy. The response surface methodology was found to be superior under all system settings. Although this result cannot be generalized, it is an indicator for the superiority of the response surface method in optimization of such models and likewise settings. On the other hand, due to their ease of integration with simulation packages, metaheuristics are finding much more applications as compared to response surface method.

فىهذا البحث تم بناء نموذج لمنظومة تحكم مستمر فى المخزون بإستخدام أسلوب المحاكاة، وتم استخدام هذا النموذج فى عقد دراسة مقارنة بين الطريقتين المذكورتين للعثور على الحل الأمثل لهذه المنظومة، وذلك عن طريق تحديد كل من حد إعادة الطلب والحد المستهدف للمخزون اللذان يحققان أقل تكلفة إجمالية لمنظومة التحكم فى المخزون. تتكون التكلفة الإجمالية من العناصر التالية: تكلفة الطلب، تكلفة التخزين، وتكلفة العجز عن الوفاء بالطلب. الطريقتان التى تم استخدامهم هما طريقة سطح مستوى الأداء والطرق الإرشادية. تمت مقارنة الطريقتين تحت فرضيات مختلفة لمدخلات المنظومة وهى معدل الإستهلاك، حجم الطلبيات، زمن التوريد، وسياسة حساب العجز. هذا وقد تفوقت طريقة سطح مستوى الأداء فى جميع الأحوال التى تم دراستها. بالرغم من أن هذه النتائج لا يمكن تعميمها، فهى تعتبر مؤشر ذو دلالة على تفوق طريقة تحقيق الأمثلية باستخدام سطح مستوى الأداء فى نفس ظروف النظام الذى تم اختياره. على جانب آخر تتميز الطرق الارشادية بسهولة استخدامها وسهولة دمجها فى نظم المحاكاة لذا تجتذب هذه الطرق الباحثين والمستخدمين على حد سواء وتلقى رواجا أكبر عموماً.

Keywords: Response surface methodology (RSM), Inventory models, Continuous review model, Simulation optimization, Metaheuristics

1. Introduction

Extensive research is currently focusing on how to combine simulation and optimization in practice. Although simulation optimization has been an active area of research for a considerable time, optimization packages have only been recently incorporated into commercial simulation software. Examples of such optimization packages include ProModel's SimRunner and AutoMod's AutoStat that use evolutionary and genetic algorithms, SIMUL8's OPTIMIZ that uses neural networks, and OptQuest package, which works with Arena and Crystal Ball, and uses scatter search, tabu search, and neural networks.

The mentioned examples illustrate how commercial simulation optimization packages are currently dominated by metaheuristic approaches. Thus, in simulation optimization practice, such methods appear to take precedence over other methods that have received more attention by the academic research community and may have more appealing convergence properties. There are several reasons for this. One explanation is that convergence properties such as asymptotic convergence have limited relevance in practice, and the metaheuristics are generally fast, robust, and generate multiple alternative solutions while focused on finding the optimal solution as illustrated in Olafsson and Kim [1].

Response Surface Methodology (RSM) is based on approximation of the stochastic objective function by a low order polynomial on a small sub region of the solution domain. The coefficients of this model are obtained by ordinary least squares applied to a number of observations of the stochastic function. Based on the fitted model the local best point is derived which is used as a current estimator of the optimum and as a center point of a new region of interest Neddermeijer et al. [2]. Heuristic methods search the solution space by building and then evolving a population of solutions. This population is achieved by creating new trials solutions out of the combination of two or more solutions that are in the current population April et al. [3].

The problem of interest involves an (*s, S*) continuous review inventory control model. The objective is to find values of reordering level *s* and order up to level *S* that minimizes the total inventory costs. Firms are continuously revising their inventory management policies to meet the dynamic nature of the market in order to achieve better profitability. This is done by lowering holding costs through higher inventory rotation, but without triggering substantial stock-outs or backorders, caused by demand peaks and/or lead time delays. The current mathematical models for inventory control do not give enough coverage for such systems; also they do not consider many important aspects of the complex dynamic behavior of today's markets. Such aspects like the crossing of replenishment orders, the sporadic orders superimposed on regular demand patterns, the higher variability in replenishment lead times due to global sourcing and due to changes in replenishment order sizes.

The Simulation model was developed using Arena™ in a generic way, where the arrival pattern, demand size, lead time pattern, and, shortage policy (lost sales versus backorders) could be pre-specified for the studied case. Optimization models were built for response surface method using Design Expert™ and for heuristic methods using OptQuest™. Both treats the simulation model as a black box, that is, the optimization results does not affect the simulation model. Design Expert™ uses Response Surface Methodology with the observations obtained from the simulation model. OptQuest™ uses the responses generated by the simulation model to make decisions regarding the selection of the next trial solution.

The rest of the paper is organized as follows: section two contains a review for the basic simulation optimization methods, section three presents the simulation model validation and the experimental study, then optimization results, the conclusions are illustrated in section four, finally, the future work is presented.

2. Simulation based optimization

A simulation experiment is normally used to study a system using a simulation model. A simulation experiment can be defined as a test or a series of tests in which meaningful changes are made to the input variables of a simulation model so that the reasons for changes in the output variable(s) can be observed and identified. When the number of input variables is large and the simulation model is complex, the simulation experiment may become computationally prohibitive. Besides the high computational cost, an even higher cost is incurred when sub-optimal input variable values are selected. The process of finding the best input variable values from among all possibilities without explicitly evaluating each possibility is called simulation optimization Carson and Maria [4].

One of the disadvantages of simulation is that it was not originally introduced as an optimization technique. An analyst would simulate a relatively small number of system configurations and choose the one that appeared to give the best performance. However, based on the availability of faster computers, most discrete-event simulation-software vendors have now integrated optimization packages into their simulation software. It could arguably be said that optimization is the most significant new simulation technology Fu et al. [5]. The goal of an "optimization" package is to orchestrate the simulation of a sequence of system configurations each configuration corresponds to particular settings of the decision variables (factors) so that a system configuration is eventually obtained that provides an optimal or near optimal solution. Furthermore, it is hoped that this optimal solution can be reached by simulating only a small percentage of the configurations that would be required by exhaustive enumeration.

During the simulation optimization process, the output of the simulation model is used by the optimization method to provide feedback on the progress of the search for the optimal solution. This in turn guides further input to the simulation model. There are five major categories of simulation optimization methods: Gradient based search methods, stochastic optimization, response surface methodology, heuristic methods, and statistical methods. Carson and Maria [4] reviewed and classified the different methods and associated simulation optimization techniques; based on their classification, an updated version has been prepared. The updated version is illustrated in fig. 1. The simulation optimization methods considered in this work will be further discussed.

2.1. Gradient based search methods

Methods in this category estimate the response function gradient to assess the shape of the objective function and employ deterministic mathematical programming techniques for optimization purposes. One frequently used gradient estimation method is Perturbation Analysis (PA). PA is a sample-path-based technique for estimating a gradient from a single simulation of a discrete-event system. Infinitesimal Perturbation Analysis (IPA) is the simplest and most efficient version of the technique, but its domain of applicability is very limited. In infinitesimal perturbation analysis (IPA) all partial gradients of an objective function are estimated. The idea is that in a system, if an input variable is perturbed by an infinitesimal amount, the sensitivity of the output variable to the parameter can be estimated by tracing its pattern of propagation. This will be a function of the fraction of propagations that die before having a significant effect on the response of interest. IPA assumes that an infinitesimal perturbation in an input variable does not affect the sequence of events but only makes their occurrence times slide smoothly, so it require continuity in performance measure. Smoothed Perturbation Analysis (SPA) is a very general technique, but the generality often comes at the expense of complexity. The general idea of smoothed perturbation analysis is to smooth discontinuities in the sample performance measure that infinitesimal perturbation analysis cannot handle. SPA can trace the effect of discontinuity of performance measure. An experimental study using this method was developed by Fu [6], and Fu and Healy [7], on an (s, S) inventory system. The results of Fu and Healy [7] are used to validate the simulation model used in this paper.

2.2. Heuristic methods

Heuristic methods represent the latest development in the field of direct search methods that are frequently used for simulation optimization. Many of these techniques balance exploration with exploitation thereby resulting in efficient global search strategies as discussed by Carson and Maria [4].

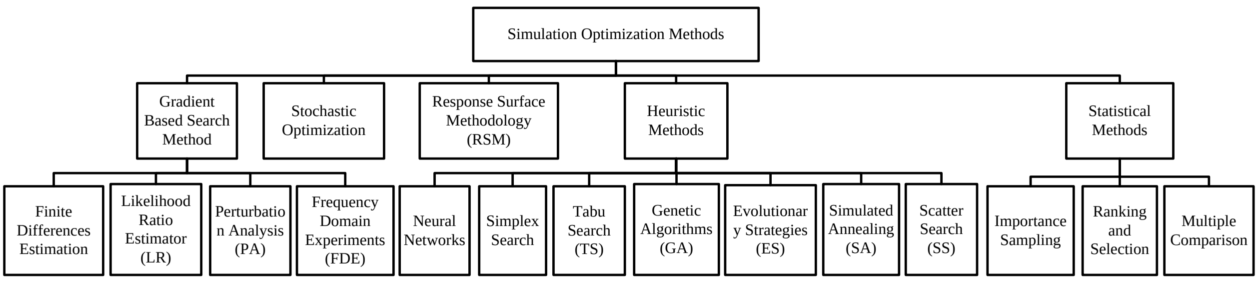


Fig. 1. Simulation optimization methods.

2.2.1. Tabu Search (TS)

Tabu search is a meta-heuristic that guides a local heuristic search procedure to explore the solution space beyond local optimality. One of the main components of tabu search is its use of adaptive memory, which creates more flexible search behavior. Memory-based strategies are therefore the hallmark of tabu search approaches, founded on a quest for “integrating principles,” by which alternative forms of memory are appropriately combined with effective strategies for exploiting them. Glover and Laguna [8] illustrated the principles and applications of tabu search in many areas.

The method can be viewed as an iterative technique which explores a set of problem solutions by repeatedly making moves from one solution to another solution located in the neighborhood. These moves are performed with the aim of efficiently reaching a solution that qualifies as "good" (optimal or near-optimal) by the evaluation of some objective function to be minimized. The tabu search approach seeks to counter the danger of entrapment at a local sub-optimum, by incorporating a memory structure that forbids or penalizes certain moves that would return to a recently visited solution. The notion of using memory to forbid certain moves can be formalized in general by saying that the solution neighborhood depends on the time stream, hence on the iteration number. A comprehensive review of TS can be found in Glover et al. [9].

2.2.2. Scatter Search (SS)

Scatter search (from the standpoint of meta-heuristic classification) may be viewed as an evolutionary algorithm that constructs solutions by combining others. It derives its foundations from strategies originally proposed for combining decision rules and constraints. The goal of this methodology is to enable the implementation of solution procedures that can derive new solutions from combined elements. The way scatter search combines solutions and updates the set of reference solutions used for combination sets this methodology apart from other population based approaches.

As Fu et al. [5] illustrated, the combination strategy has been devised with the belief that this information could be exploited more effectively when integrated than when treated in isolation (i.e., when existing selection rules are selected one at a time). In general, the decision rules created from such combination strategies produced better empirical outcomes than standard applications of local decision rules. They also proved superior to a “probabilistic learning approach” that used stochastic selection of rules at different junctures, but without the integration effect provided by generating combined rules. In integer and nonlinear programming, associated procedures for combining constraints were developed, which likewise employed a mechanism for creating weighted combinations. In this case, nonnegative weights were introduced to create new constraint inequalities, called surrogate constraints. The main function of surrogate constraints was to provide ways to evaluate choices that could be used to create and modify trial solutions, an illustrative review of SS can be found in.

2.2.3. Neural networks

Neural Networks is an information processing paradigm that is inspired by the way biological nervous systems process information. It is composed of a large number of highly interconnected processing elements (neurons) working in harmony to solve specific problems. Neural Networks, like people, learn by example. Neural Networks is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. Metaheuristic optimizers typically use meta models as filters for screening out solutions that are predicted to be inferior compared to the current best known solution. Raymond-smith et al. [10] demonstrate the different types of heuristic based search methods an elaborate on their applications and the differences between them.

In this paper a comparison is held between a metaheuristics based simulation optimization method using OptQuest and a Response Surface based method using DesignExpert. As illustrated in April et al. [3], OptQuest uses neural networks to build a meta model and then applies predefined rules to filter out potentially bad solutions.

2.2.4. The OptQuest® optimization algorithm

The Solution of the optimization problem can be represented by a variable x_i, (for i = 1, ..., n) Laguna [11]. The objective function value $f(x)$ is obtained by running a simulation model that uses x as the input factors, a set of linear constraints (equality or inequality) are imposed on x. The algorithm starts by generating an initial population of reference points. The initial population may include points suggested by the user, and it always includes the following midpoint:

$$x_i = l_i + \frac{(u_i - l_i)}{2}.$$

Where u_i and l_i are the upper and lower bounds on x_i, respectively. Additional points are generated with the goal of creating a diverse population. A population is considered diverse if its elements are “significantly” different from one another. A distance measure to determine how “close” a potential new point is from the points already in the population is used, in order to decide whether the point is included or discarded. Every reference point x is subjected to a feasibility test before it is evaluated (i.e., before the simulation model is run to determine the value of $f(x)$). The feasibility test consists of checking (one by one) whether the linear constraints imposed by the user are satisfied. An infeasible point x is made feasible by formulating and solving a linear programming (LP) problem. The LP (or mixed-integer program, when x contains integer variables) has the goal of finding a feasible x^* that minimizes the absolute deviation between x and x^*.

The population size is automatically adjusted by the system considering the time that is required to complete one evaluation of $f(x)$ and the time limit the user has allowed the system to search. Once the population is generated, the procedure iterates in search of improved outcomes. At each iteration two reference points are selected to create four offspring. Let the parent-reference points be x_1 and x_2, then the offspring x_3 to x_6 are found as follows:

$$\begin{aligned} x_3 &= x_1 + d \\ x_4 &= x_1 - d \\ x_5 &= x_2 + d \\ x_6 &= x_2 - d \end{aligned}.$$

Where $d = (x_1 - x_2)/3$. The selection of x_1 and x_2 is biased by the values $f(x_1)$ and $f(x_2)$ as well as the tabu search memory functions. The iteration ends by replacing the worst parent with the best offspring, and giving the surviving parent a tabu-active status for given number of iterations. In subsequent iterations, the use of two parents is forbidden.

The restarting strategy: When searching for a global optimum, the population may contain many reference points with similar characteristics. That is, in the process of generating offspring from a mixture of high-quality reference points and ordinary reference points that are member of the current population, the diversity of the population may tend to decrease. A strategy that remedies this situation considers the creation of new population.

Adaptive memory and the age strategy: Some of the points in the initial population may have poor objective function values. Therefore, they may never be chosen to play the role of a parent and would remain in the population until restarting. To additionally diversify the search, the “attractiveness” of these unused points is increased over time. The idea is to use search history to make reference points not used as parents “attractive,” by modifying their objective function values according to their age.

The neural network accelerator: Neural Networks is an information processing paradigm that is inspired by the way biological nervous systems process information. It is composed of a large number of highly interconnected processing elements (neurons) working in harmony to solve specific problems. This strategy is designed to increase the power of the system's search engine. The concept is to "screen out" values x that are likely to result in a very poor value of $f(x)$.

2.3. Response Surface Methodology (RSM)

Response Surface Methodology (RSM) searches for the input combination that optimizes the simulation output (by evaluating it at several input variable values) and optimizing the resulting regression function. The process starts with a first order regression function and the steepest ascent/descent search method. After reaching the vicinity of the optimum, higher degree regression functions are employed Angun et al. [12]. The establishment of a clear and consistent RSM optimization algorithm is of significant importance for its use as a tool in scientific applications, e.g. for estimation of model parameters, where results should be reproducible and derived via a clear method. A complete and clear definition of all steps and choices in a RSM algorithm is also necessary for automated optimization where all choices concerning the algorithm have to be made at the outset of an application. Automated optimization is less time-consuming, since there is no need to interfere in this optimization process. This is an advantage in large-scale time-consuming applications. However, there is no consensus about such a standard RSM algorithm.

The RSM procedure comprises two phases. In the first phase, the objective function is locally approximated by first-order polynomials; in the second phase, the objective function is approximated by a second-order polynomial. In both phases a region of interest has to be defined, this region is a sub-region from the domain. When the first-order model is found to be adequate a steepest descent procedure is applied to find a new region of interest. Otherwise the RSM moves to the second phase. When a second-order model is approximated and found to be adequate a stationary point needs to be found and classified and an appropriate action should be taken.

In design for first order approximation there are many designs to choose from, like fractional or full factorial, and two-level or three-level designs. In non-automated optimization the user tries to fit a first-order approximation with different designs, apply coding of the parameters to find better regression estimates or recalculate the objective value at the design points. For an automated RSM procedure the objective function is evaluated once in the 2^k points of a two-level full factorial design, where k is the number of factors, and five times in the center point of the current region of interest. This design is orthogonal and does not require as many points as a three-level full factorial design. Two-level fractional factorial designs consist of too few points to approximate objective functions with two or three parameters well enough. Furthermore, full factorial designs can quite easily be augmented to derive a second-order design. A comprehensive presentation and explanation of RSM can be found in Myers and Montgomery [13].

In design for second order model in the region of interest, the regression coefficients of this model are again determined by regression analysis, applied to observations performed in an experimental design. A popular second-order design is the Central Composite Design (CCD). The CCD arises when the full (or fractional) factorial design is augmented by the first-order design with 2^k axial points. This design is made spherical by choosing the new points such that all points are equidistant to the center point of the current region of interest. This design is chosen for two reasons. First of all this design can almost be rotated and the loss in rotation is trivial. Furthermore, in a rotatable design, the distance of the new points to the center point would be large as compared to the distance of the existing points to the center point.

Neddermeijer et al. [2], presented a framework for an automated response surface algorithm, the framework is illustrated in fig. 2.

It is assumed that a screening phase, in which factors that are considered unimportant are eliminated from the optimization problem, as well as possible transformations of the factors and the response, have already taken place. At the start of the algorithm, an initial starting point and initial step sizes should be given. Choosing the initial step sizes at the start of the algorithm should be done. The framework includes the following steps:

1. Approximate the simulation response function in the current region of interest by a first-order model
2. Strategic moves: Test the first-order model for adequacy

Usually, a test for lack of fit and a test for significance of regression are performed. The test for lack of fit is a joint test for interaction between factors as well as for curvature. If the first order model is fitting, and some of the regression coefficients are not equal to zero, then a line search is performed. If one of the tests fails then this model is not adequate and a second order model needs to be fitted.

3. Perform a line search in the steepest descent/ascent direction
4. Solve the inadequacy of the first-order model

This can be done by reducing the size of the region of interest or increase the simulation size used in evaluating a design point.

5. Approximate the objective function in the current region of interest by a second-order model
6. Strategic moves: Test the second-order model for adequacy

This module checks if a second-order model describes the behavior of the objective function in the current region of interest. Similar to the first-order model a lack of fit test can be used. Now, the null hypothesis of this test is that the true regression model is quadratic.

7. Solve the inadequacy of the second-order model

This can be done in the same way mentioned before.

8. Perform canonical analysis

If the second-order model is found to be adequate, then canonical analysis is performed to determine the location and the nature of the stationary point of the second-order model.

9. Perform ridge analysis

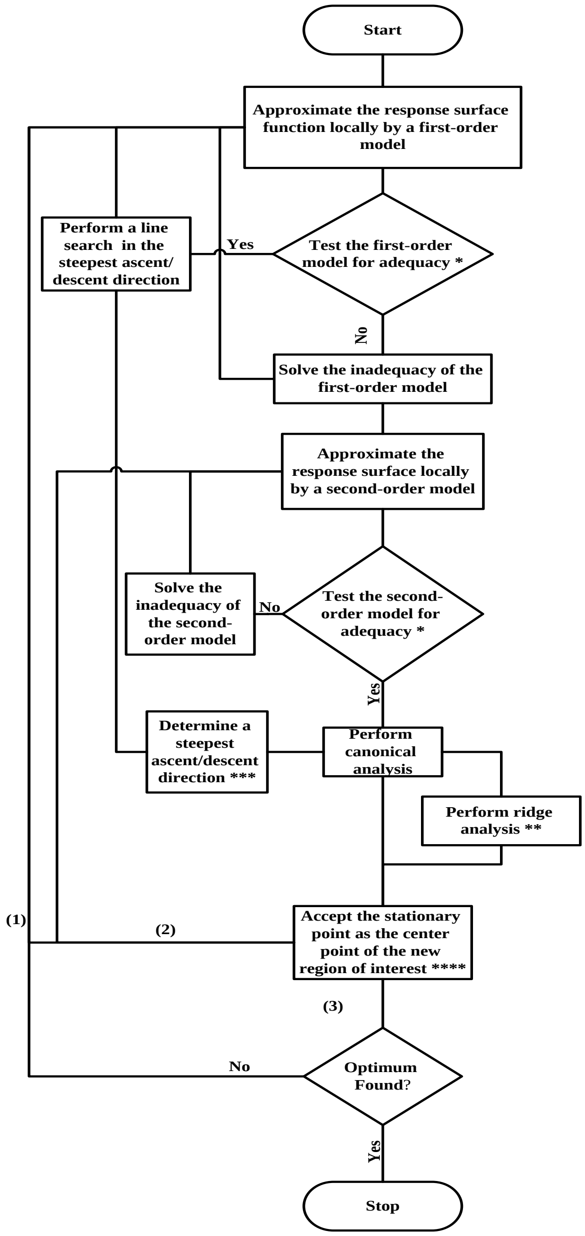
It is not advisable to extrapolate the second-order polynomial beyond the current region of interest. Therefore, if the stationary point is a minimum which lies outside the current region of interest, the stationary point is not accepted as the center of the next region of interest. If the stationary point is a maximum or a saddle point, then the stationary point is rejected as well. In these cases, ridge analysis is performed, which means a search for a new stationary point such that the second order model has a minimum at this stationary point.

10. Accept the stationary point
11. Determine a steepest descent direction from the second-order model

The gradient of the second-order model at the center point of the current region and the results of the canonical analysis can be used to determine a direction of steepest descent. Next, a line search can be performed using this direction, resulting in a new center of a region of interest. In this region the simulation response surface will be approximated by a first-order model.

In the RSM literature, it is often proposed to end the algorithm after fitting only one second order polynomial. This strategy assumes that a minimum inside the current region is found, and therefore excludes the cases in which either a minimum outside the current region is found or a maximum or a saddle point is found. It is recommended ending the optimization exercise if: (i) the estimated optimal simulation response value does not improve sufficiently anymore, (ii) if the region of interest becomes too small, or, in case there are budget constraints, (iii) if a fixed maximum number of evaluations have been performed. Next, a confidence interval about the response at the estimator for the optimum and the location of this estimator can be determined.

In this paper an attempt is done to compare the application of Metaheuristics and RSM methods of simulation based optimization to find the optimum values of the parameters s and S that will minimize the long-run average total inventory costs. These costs are associated with ordering, holding, and shortages.



* Stategic moves: a test for lack of fit and a test for significance of regression are performed

** Ridge analysis is performed if the stationary point lies outside the current region of interest

*** Line search is performed in order to find a new center of a region of interest

**** Depending on the results of the canonical analysis:

(1) first-order approximation is performed for the next region to find a direction of improvement when the optimum is located far away.

(2) second-order approximation is performed for the next region.

(3) Stopping Criteria

Fig. 2. Framework for RSM Algorithm.

3. Simulation optimization case study

3.1. The simulation model and model validation

A simulation model for a continuous review (*s, S*) inventory control system of a single item has been developed. An (*s, S*) ordering policy specifies that an order is placed when the level of inventory on hand plus that on order (the inventory position) falls below the reorder level (*s*), and that the amount of the order is the difference between the order up to level (*S*) and the inventory position, as illustrated in fig. 3.

The logic chart of the simulation model, is illustrated in fig. 4, the flowchart terminology is listed in table 1. At every demand transaction, the time counter i is incremented by the value of the time between arrivals distribution I, then the stock level is checked to calculate the shortage or to update the stock and add the holding cost. After this, the stock level is checked again but this time by the inventory position Y to know if there is need to order a new replenishment Q or not. If a new replenishment order is placed, the lead time LT is calculated and the time of receipt is recorded.

In order to validate the developed simulation model, two approaches were used, first, a hand simulation was performed using deterministic values, then, the performance of the developed model was compared with the results obtained from an experimental study for a periodic review (*s, S*) inventory control system developed by Fu and Healy [7], where perturbation analysis (PA) was used for optimization. As Fu model is a periodic review model, the review period in the developed simulation model was set as one transaction, i.e. the inventory status is reviewed at each time period. The experiment was performed with a run length of 50,000 periods, 20 replications and 95% confidence levels Fu and Healy [7] assumed the expected demand quantity to be exponentially distributed with an average of 200 units per period and, the optimum values obtained by the gradient based algorithm for the reordering point and ordering quantity were 341 and 200 units respectively. The recorded average inventory cost was \$740.9 per period, and the obtained

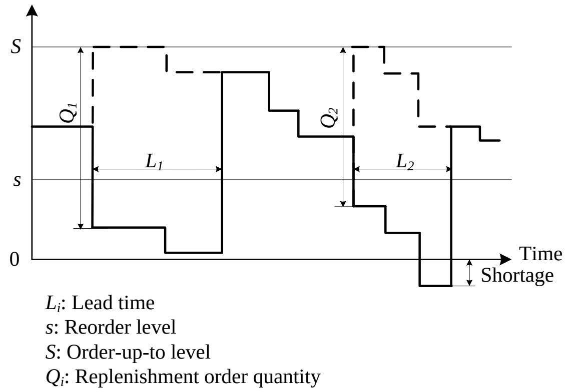


Fig. 3. The continuous review (s, S) inventory control system.

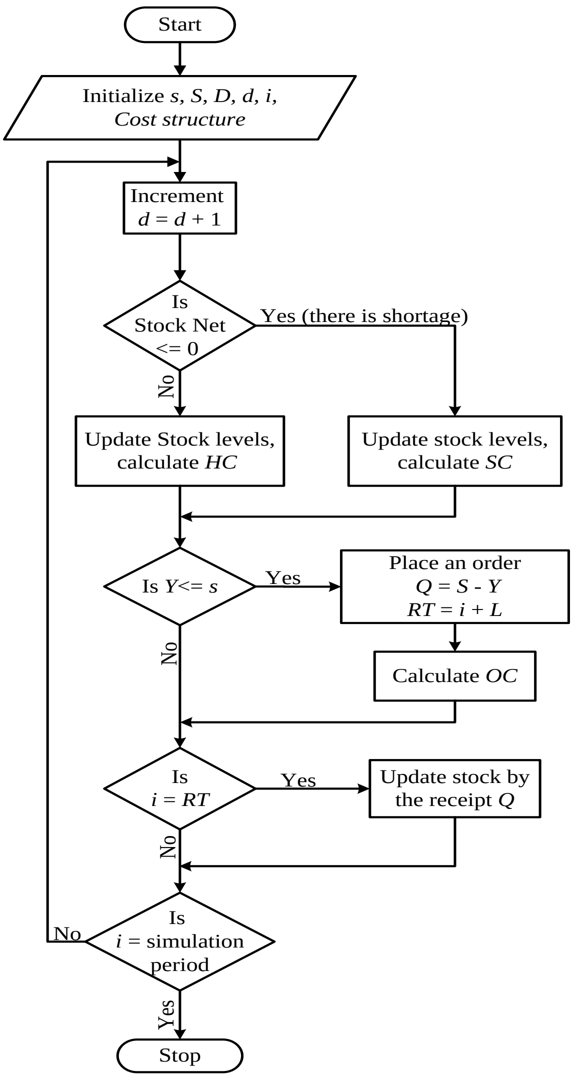


Fig. 4. The simulation model logic chart.

Table 1
Flowchart symboles

i	Simulation time counter	Q	Replenishment order quantity
s	Reorder level	RT	Receipt time
S	Order up to level	I	Cumulative area under stock curve
D	Demand size distribution	OC	Ordering cost
L	Lead time	HC	Holding cost
d	Demand instances	SC	Shortage cost
Y	Inventory position		

average inventory cost per period is $740.97. As there is no significant difference between the two means the simulation model was considered valid.

3.2. The Experimental study

Different cases were studied and the optimum solution for each case was derived using the methods under consideration. The total inventory cost was calculated for different values of demand rate, demand size, lead time, and shortage policy. The distributions of the demand size, time between demand transactions, lead time, and the reorder and order up to level values were chosen assuming an average consumer product like household appliances for example. The cost structure is as follows: the inventory carrying charge is $0.0005 per dollar per day, the cost of placing an order from the supplier is $1 per order, a lost sale results in a loss of goodwill estimated to cost $100, finally, the cost for maintaining a backorder is $50. The results and optimum solutions of the different cases are summarized in table 2. The response surface plot and contour plot for two of the examined cases are shown in fig. 5. The response surface plot is a 3D graphical representation of the fitted model. The contour plot represents lines of equal response values. Case no. 3 and case no. 8 have the same input parameters values. The shortage policy for both of them is different. In case no. 3, approximately 50% decrease in cost is obtained, In case no. 8, the cost has decreased by 15%.

It can be observed that the RSM based approach generally provides a better optimum solution than that found using the heuristics based approach. The Heuristic based approach have the advantage of ease of computations. This is because the optimization model directly interacts with the simulation model. On the other hand RSM is based on design of experiments, so the system performance can be assessed statistically. Several observations can be obtained from the response plot, regarding the effect of changing decision parameters on the response. Also, the developed model can be used for point prediction, i.e. the value of the response can be predicted at points which were not examined by the simulation model. For example, in the contour plots of case no. 8 illustrated in fig. 5-b, if the decision maker is examining the impact of changing the value of the order up to level (S) for a specific reorder level (s). If the value of s=8 units increasing the value of S from 27 to 30, 33, 37, or 42, is expected to result in a total cost of 19848, 16250, 13011, 9773, and 7365 respectively.

Also, from the point of view of total cost, if the decision maker selects S=37, it makes no difference if he uses s = 3 or 8. Of course, in such a case other intangible factors can lead to a suitable choice. The same goes for the cases of S=30 and S=27. The RSM plots represent a sub-region of domain where the optimum found was better than the solution found using heuristic methods. A decision should be made to move from the current sub-region to another where better improvement can be made. The RSM is based on approximations of the objective function by a low order polynomial on a small sub-region of domain. The strategic moves between sub-regions require time and effort from the side of the user. On the other hand, metaheuristics can work automatically on the whole domain, that's why most of the commercial simulation software packages use heuristic methods as its optimization engine. The metaheuristic approach to simulation optimization is based on viewing the simulation model as a black box function evaluator April et. al. [16]. Although, on a sub-region search, better solutions can be obtained by using RSM than the solutions obtained by using heuristic methods.

Table 2
Simulation optimization test cases

Case no.	Time between demand transactions	Transaction size	Lead time	Shortage policy	Heuristic methods			RSM		
					Total cost/year	s	S	Total cost/year	s	S
1	EXPO(0.5)	POIS(1)	LOGN(2, 2)	Backordering	4,602.74	4	45	3,885.28	2	45
2	EXPO(0.5)	POIS(2)	LOGN(7, 2.1)	Backordering	69,068.60	9	45	63,344.60	2	45
3	EXPO(1)	POIS(2)	LOGN(4, 2.8)	Backordering	8,032.06	9	45	3,854.91	6	43
4	EXPO(2)	POIS(1)	LOGN(2, 2)	Backordering	395.33	3	45	342.49	4	45
5	EXPO(2)	POIS(4)	LOGN(2, 2)	Backordering	6,482.47	2	45	5,436.96	4	43
6	EXPO(0.5)	POIS(1)	LOGN(2, 2)	Lost sales	4,501.37	3	45	4,264.92	6	45
7	EXPO(0.5)	POIS(2)	LOGN(7, 2.1)	Lost sales	105,051.00	2	44	103,801.00	3	45
8	EXPO(1)	POIS(2)	LOGN(4, 2.8)	Lost sales	7,764.97	7	44	6,560.72	6	44
9	EXPO(2)	POIS(1)	LOGN(2, 2)	Lost sales	348.35	3	45	380.57	4	45
10	EXPO(2)	POIS(4)	LOGN(2, 2)	Lost sales	9,560.65	4	45	6,108.12	8	45

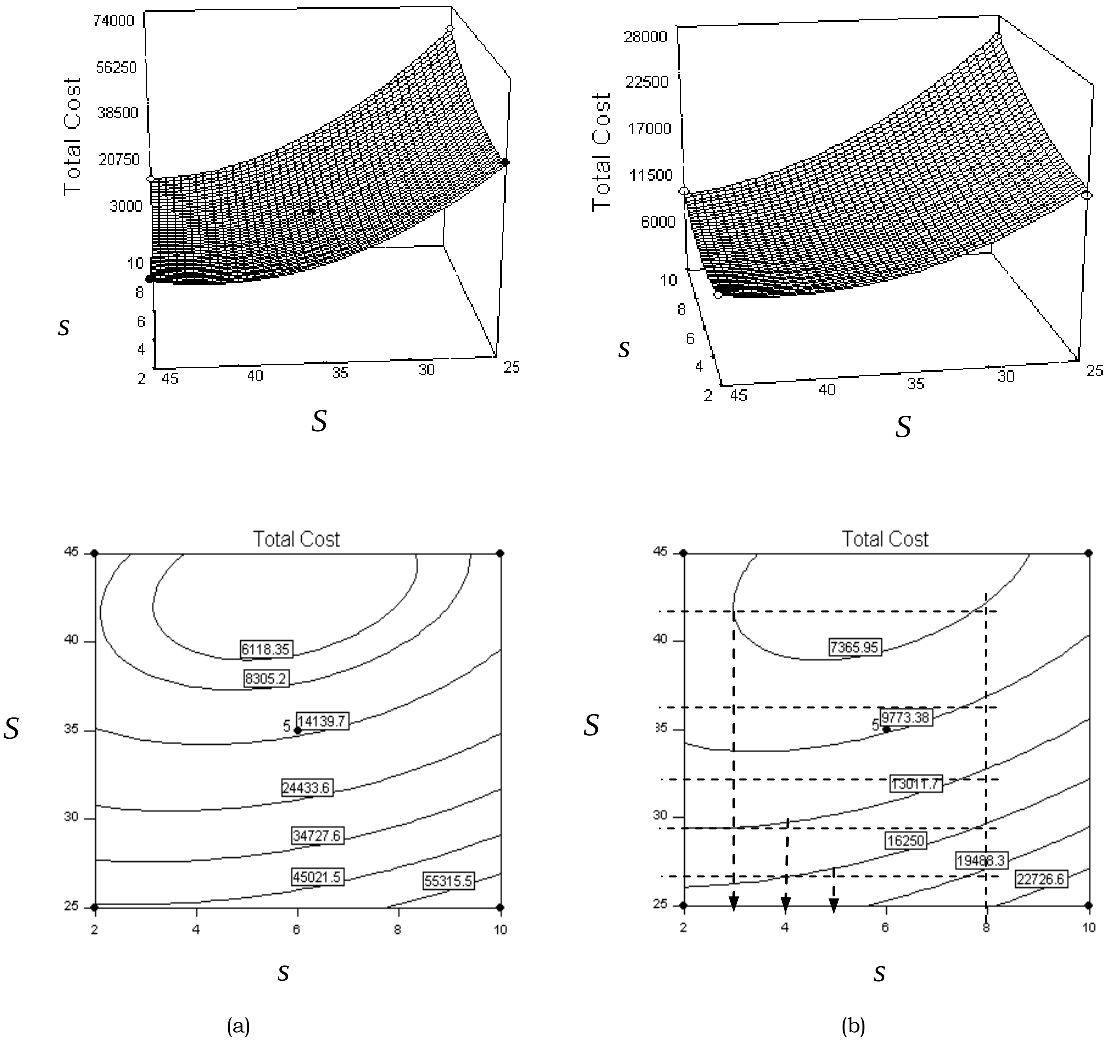


Fig. 5. Response surface and contour plots for (a) case 3 and (b) case 8.

4. Conclusion and future work

In this paper the major categories of simulation based optimization were reviewed. A comparison between heuristic based methods and response surface methodology was illustrated. The comparison was done on a simulation based continuous review (*s, S*) inventory control model and the objective was to minimize the total inventory relevant cost.

The results showed that the response surface method was superior in the tested cases. Although this is cannot be generalized, it can be argued that RSM can generally perform better for the same nature of problems and in the same settings. Another benefit of RSM is that it provides response prediction facility. On the other hand, heuristic based methods have the advantage of ease and speed of computations.

Analysis of the results illustrates how the total inventory costs can drastically change due to changes in the transaction size and delivery lead time. RSM can be further used to investigate the effect of each factor and the interaction effect on total inventory costs. This can be used to set some recommendations to the decision maker on appropriate settings for the different system parameters in such cases.

References

[1] S. Olafsson and J. Kim "Simulatiom Optimization", In: E. Yücesan, C.-H. Chen, J.L. Snowdon, and J.M. Charnes, Proceedings of the 2002 Winter Simulation Conference. Piscataway, New Jersey: IEEE Press (2002).

[2] H. Neddermeijer, G. Oortmarssen, N. Piersma and R. Dekker, "A Framework for Response Surface Methodology for Simulation Optimization", In: J.A. Joines, R.R. Barton, K. Kang, and P.A. Fishwick, Proceedings of the 2000 Winter Simulation Conference. Piscataway, New Jersey: IEEE Press, Vol. 1, pp. 129-136 (2000).

[3] J. April, F. Glover, J.P. Kelly and M. Laguna, Practical Introduction to Simulation Optimization, Proceedings of the 2003 Winter Simulation Conference. Piscataway, New Jersey, IEEE Press, Vol. 1, pp. 71-78 (2003).

[4] Y. Carson and A. Maria. Simulation Optimization: Methods and Applications. Proceedings of the 1997 Winter Simulation Conference Piscataway, New Jersey, IEEE Press, pp. 118–126. (1997).

[5] M. Fu, F. Glover and J. April, "Simulation Optimization: A Review", New Developments, and Application. Proceeding of the Winter Simulation Conference. Piscataway, New Jersey: IEEE Press, p.13 (2005).

[6] M. Fu, Sample Path Derivatives for (s, S) Inventory Systems. Operations Research, Vol. 42 (2), pp. 351-364 (1994).

[7] M. Fu and K. Healy, "Techniques for Optimization via Simulation: an Experimental Study on an (s, S) Inventory System", IIE Transactions, Vol. 29, pp. 191-199 (1997).

[8] F. Glover, M. Laguna, Tabu Search. Kluwer Academic Publishers, Norwell, MA, USA (1997).

[9] F. Glover, E. Taillard and D.A. Werra, "User's Guide to Tabu Search", Annals of Operations Research, Vol. 41, pp. 3-28.

[10] V. Raymond-smith, I.H. Osman, C.R. Reeves and G.D. Smith, Modern Heuristic and Search Methods. John Wiley and Sons (1996).

[11] M. Laguna, "Optimization of Complex Systems with OptQuest", OptTek Systems and Decisioneering Company,http://www.decisioneering.co m.optquest/complexsystems.html, (15/1/2008) (1997).

[12] E. Angün, J. Kleijnen, D. Hertog and G. Gürkan, "Response Surface Methodology Revisited", Proceedings of the 2002 Winter Simulation Conference. Piscataway, New Jersey: IEEE Press, pp. 373-383 (2002).

[13] R. Myers and D. Montgomery, Response surface Methodology, Process and Product Optimization Using Designed Experiments. John Wiley and Sons, New York, USA (1995).

[14] R. Nicolai, R. Dekker, N. Piersma and G. Oortmarssen, "Automated Response Surface Methodology for Stochastic Optimization Models with Unknown Variance", Proceedings of the 2004 Winter Simulation Conference. Piscataway, New Jersey: IEEE Press., Vol. 1, p. 499 (2004).

[15] E. Silver, D. Pyke, R. Peterson, Inventory Management and Production Planning and Scheduling, 3rd Edition, John Willy and Sons New York, USA (1998).

[16] J. April, M. Better, F. Glover, J. Kelly and M. Laguna, "Enhancing Business Process Management with Simulation Optimization", In the Proceedings of the Winter Simulation Conference, pp. 642-649 (2006).

Received January 15, 2008
Accepted October 27, 2008