

OPTIMISATION OF INSPECTION STATIONS BY USING SIMULATED ANNEALING

Azmi Hassan¹ and D.T.Pham²

1. Department of Mechanical Engineering, Faculty of Engineering, Universiti Kebangsaan Malaysia, Bangi, 43600, Selangor, MALAYSIA.
2. Division of Systems Engineering, School of Engineering, University of Wales Cardiff, P.O. Box 688, Cardiff, CF2 3TE, UNITED KINGDOM.

ABSTRACT

Simulated Annealing (SA) is a heuristic solution strategy to find an optimal solution in a problem with multiple local minima by dynamically changing the probability of accepting inferior solutions during problem solving. It is an intelligent approach designed to give a good though not necessarily optimal solution, within a reasonable computation time. SA has received much attention since its introduction and has been applied to a wide range of discrete and continuous optimisation problems. In this paper, the SA algorithm was employed to solve the problems of combinatorial optimisation of the design of inspection systems in multi-stage manufacturing systems. The problem could be addressed as follows: Consider a multi-stage manufacturing system, where there are up to three possible inspection stations to be located at any of ten processing stages, stage 0 to stage 9. The raw material is input to stage 0 and the finished product is output from stage 9. The optimal location will minimise the total cost. The objective function of SA will correspond to the total cost of the manufacturing problems. This paper will describe the representation of the solutions in a binary form, the re-configuration mechanism, and the evaluation of the objective functions based on empirically found cooling schedule. Finally the results of optimisation experiments will be presented and compared to those obtained by Branch and Bound technique found earlier in the literature.

Keyword: Simulated Annealing, Multi-stage, Optimisation

1. Overview

Inspection is concerned with separating product units that conform to specifications from those which do not

and preventing non-conforming (defective) product units from reaching the customer or the external user.

The inspection activity can be carried out in many ways: manual. Recent technological developments in automated visual inspection, pattern recognition and image processing techniques have led to an increase in the implementation of automated systems. Higher implementation costs and technical difficulties can be associated with automated systems. The selection and location of such inspection stations must be carefully considered since it will have a significant effect not only on the product quality but also on the total cost of manufacturing.

Simulated Annealing (SA) is a heuristic solution strategy to find an optimal solution in a problem with **multiple local minima** by **dynamically changing the probability of accepting inferior solutions** during problem solving. The SA technique was first introduced by Kirkpatrick [1]. This idea was based on the *Metropolis algorithm* [2]. SA simulates the cooling process of solid materials - known as **annealing**. However this analogy is limited to the physical movement of the molecules without involving complex thermodynamic systems.

SA has received much attention since its introduction and has been applied to a wide range of discrete and continuous optimisation problems [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18].

Research on improving this algorithm and widening its application areas still continues. In this paper, the SA algorithm was employed to solve the problems of combinatorial optimisation of the design of inspection systems.

2. Multi-Stage

A multi-stage manufacturing system can exist in a serial or non-serial configuration. In a serial system, each processing stage except the first in the series has a single immediate predecessor. Also, each stage except the last has a single immediate successor. Inspection stations may exist between the processing stages. In a non-serial system, at a certain stage, the product may be assembled or joined with products from other processing lines. Hence the optimisation of the allocation of inspection stations is more complex.

The optimal design of inspection stations has been studied since the 1960s. Raz [19] surveyed the elements of the inspection stations allocation problems and the models that had been proposed in the literature. His paper quoted 17 models developed from 1964 to 1984 for serial and non-serial production systems. Most of the models used dynamic programming techniques. Ballou and Pazer [20, 21] developed a production-inspection model allowing for inspection errors in a serial production system. Chakravarty and Shtub [22] suggested a shortest-path heuristic method to determine the strategic location of inspection activities and the production lot-sizes. Peters and Williams [23] used dynamic programming and direct search techniques to determine the location of quality monitoring stations. The problem with dynamic programming is that when the number of processing stages increases the complexity of computation increases dramatically. A non-optimum solution would be accepted as optimum as the problem becomes larger. Furthermore, the optimisation decisions are taken separately, *stage by stage*, rather than by performing *global optimisation* of the multi-stage system.

More recent work has been carried out by Raz and Kaspi [24]. The Branch-and-Bound technique was suggested to allocate imperfect inspection operations in a fixed serial multi-stage production system. This methodology still involves *stage by stage optimisation*. The application of Artificial Intelligence (AI) was suggested by Raz [19]. Kang *et al.* [25] proposed a Rule-based technique to determine near-optimal inspection stations. The present work extends the work of Raz and Kaspi [24] to *global optimisation* by the use of AI techniques.

3. Transfer Functions Model

In this paper, the multi-stage production-inspection model developed by Raz and Kaspi [24] is considered. The model is called the **Transfer Functions Model** (TFM). This model facilitates the implementation of

the calculations required to find the optimal solution. The TFM comprises the following parameters (see Figure 1):

- δ fraction of non-conforming units that are scrapped. Therefore the fraction of the units reworked is $(1 - \delta)$
- θ probability that a non-conforming unit will be classified as conforming
- π probability that a conforming unit becomes non-conforming as a result of improper performance of a production or repair operation
- ρ probability that a non-conforming unit remains non-conforming at the completion of a repair operation
- ϕ probability that a conforming unit will be classified as non-conforming
- f fraction of the cumulative cost of a scrapped unit that is retained as a salvage value
- PC unit processing cost added to each unit
- RC unit rework cost
- C_i cumulative unit cost just prior to the operation
- C_o cumulative unit cost after the completion of the operation
- q_i probability that a unit nonconforming just prior to the operation
- q_o probability that a unit nonconforming after the completion to the operation

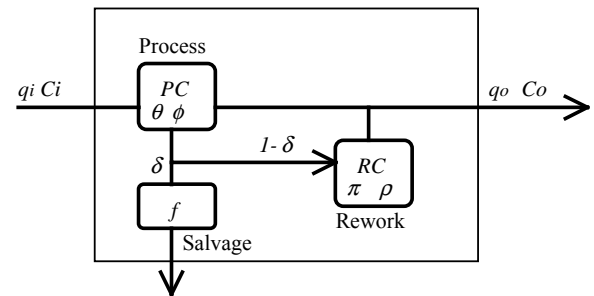


Figure 1: TFM for a Single Processing and Inspection Stage.

The final forms of both transfer functions after completion of the single operation are given as [24]:

$$C_o = \frac{(C_i - PC) - f(C_i + PC)\delta[q_i(1 + \theta) + (1 - q_i)\phi] + RC(1 - \delta)[q_i(1 - \theta) + (1 - q_i)\phi]}{[q_i(1 - \theta) + (1 - q_i)]}$$

$$q_o = \frac{q_i\theta + (1 - \delta)\{\rho[q_i(1 - \theta) + \pi(1 - q_i)\phi]\}}{[q_i(1 - \theta) + (1 - q_i)]}$$

These equations are applied to calculate the Cost Transfer Function (CTF) and Quality Transfer Function (QTF) for a **single production** or **inspection**

station. The following multi-stage problem is to be addressed:

Consider a multi-stage manufacturing system depicted in Figure 2, where there are up to 3 possible inspection stations to be located at any of 10 processing stages, stage 0 to stage 9. The raw material is input to stage 0 and the finished product is output from stage 9. The optimal location will minimise the total cost.

The total cost $C_{T[s]}$ of a single stage is given by [24]:

$$C_{T[s]} = C_{o[s]} + q_{o[s]} \cdot PC_{[s+1]}$$

where

$$C_{T[s]} \quad - \text{total cost at stage } s$$

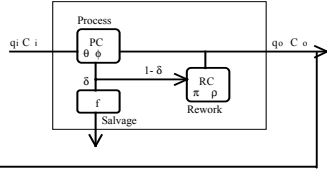
$$PC_{[s+1]} \quad - \text{production cost at the next stage } (s+1)$$

$$s \quad - 0 \text{ to } 9$$

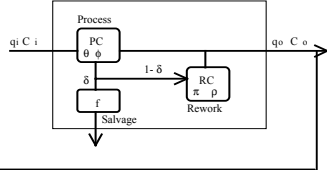
The value of $C_{T[s]}$ depends on the inspection configuration at each stage. It is given that the output of the CTF and QTF of one stage will be input to the next stage. Therefore the $C_{o[s]}$ and $q_{o[s]}$ are taken as $C_{i[s+1]}$ and $q_{i[s+1]}$. Equations 1 and 2 are applied again to find $C_{o[s+1]}$ and $q_{o[s+1]}$. The total cost for the next stage can then be described as:

$$C_{T[s+1]} = C_{o[s+1]} + q_{o[s+1]} \cdot PC_{[s+2]}$$

Stage 1



Stage 2



....Stage 9

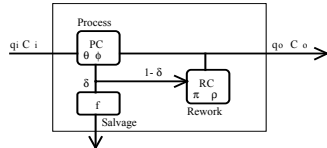


Figure 2: TFM for the Whole Production Line

When the product exits the whole system and reaches the customer or external user, a penalty cost is incurred for each non-conforming unit. The unit penalty cost is PEN . The CTF is denoted by $C_{T[p]}$:

$$C_{T[p]} = C_{o[9]} + q_{o[9]} \cdot PEN$$

The total cost per product unit, C_T , for the entire system can be described as:

$$C_T = C_{T[9]} + C_{T[p]}$$

C_T is the **objective function** of the inspection stations allocation and sequencing problem. The values of the production and inspection parameters used in this work are available from the author.

4. Principles of Simulated Annealing

SA is an intelligent approach designed to give a good though not necessarily optimal solution, within a reasonable computation time. The idea in SA, as in iterative improvement, is to make some **random perturbation**, such as moving a molecule to a new location, then **evaluate** the **resulting change in energy**, ΔE . If the energy is reduced, $\Delta E < 0$, the new configuration has less energy and is accepted as the starting point for the next move. However, if the energy is increased, $\Delta E > 0$, the move may still happen: the new, higher energy configuration may be acceptable with some probability. In physical systems, jumps to higher energies can happen, but they are moderated by the current temperature T . At higher temperatures, the probability of large *uphill moves* in energy is high. See Figure 3. At low temperatures the probability is small.

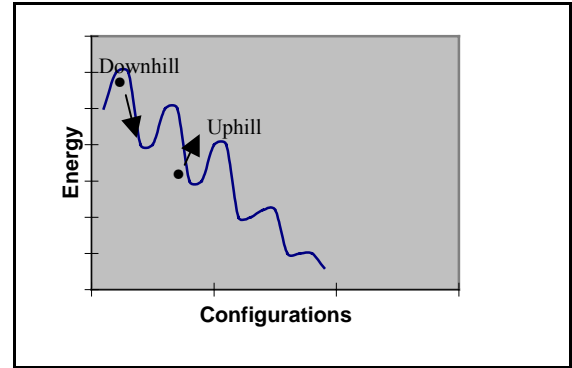


Figure 3: Downhill and Uphill Moves

By using the Boltzmann distribution, the *Metropolis algorithm* was developed [2] to accept *uphill moves* with probability:

$$Pr[accept] = \exp(-\Delta E/kT_j)$$

where $\Delta E = E_j - E_i$ (change in energy after move i to j),

$k = \text{Boltzmann constant}$,

$T_j = \text{current temperature}$.

In practice, this probabilistic acceptance is achieved by generating a uniformly random number R in $[0,1]$ and comparing it with $Pr[accept]$. If $R < Pr[accept]$, then the move is accepted, otherwise the move is rejected. By lowering the temperature and continuing to run this algorithm, the simulation of a solid material reaching equilibrium continues at each newly reduced

temperature. The analogy between physical annealing and simulated annealing can be summarised as follows:

- i. the *physical configurations* or *states of the molecules* correspond to *optimisation solutions*.
- ii. the *energy* of molecules corresponds to the *objective function* or *cost function*.
- iii. a *low energy state* corresponds to an *optimal solution*.
- iv. the *cooling rate* corresponds to the *control parameter* which will affect the acceptance probability.

A detailed description of SA is available in the literature [e.g. 26, 27]. SA is also known as a search algorithm. However SA has the advantage over local search of allowing *uphill moves*. Figure 4 shows a typical optimisation curve in SA.

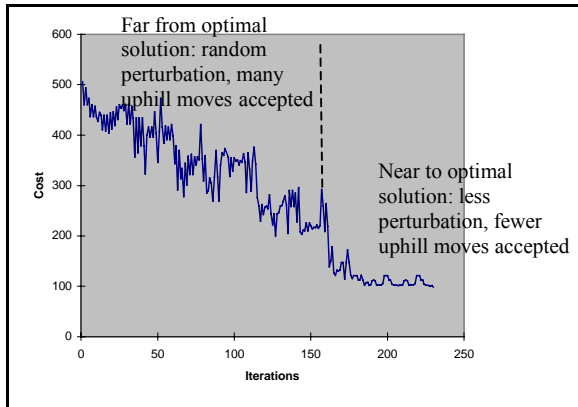


Figure 4: Example of Optimisation Curve

5. Implementation

Prior to presenting the results obtained, this section will discuss the implementation details. It is recalled that there are 10 processing stages with 3 types of inspection station, **I1**, **I2** and **I3**. The inspection stations allocation problem has a solution of 16^{10} possible solutions. This is because there are 10 processing stages and at each stage there are:

$$\sum_{i=0}^3 \frac{3!}{(3-i)!}$$

or 16 possible inspection arrangements. The processing stages are arranged serially. The inspection stations can be allocated to any stage of processing. Therefore the total number of configurations for all processing stages is 16^{10} (2^{40} possible solutions). The same 40-bit binary string representation scheme applied in the case of the Genetic Algorithm (GA) [28] was implemented for SA because of its flexibility and

ease of computation. There are many techniques that could be used to generate new solutions from the current configuration. *Mutation* and *inversion* will be applied in this implementation. The cost function for this problem is the **objective function**.

The annealing process started at a high temperature, $T = 600$ units. So most of the moves were accepted. The cooling schedule was represented by:

$$T_{i+1} = \lambda T_i$$

where λ is the cooling rate parameter, which was determined experimentally. 3 cooling rates were used ($\lambda = 0.9, 0.98, 0.99$). With $\lambda=0.9$, the temperature reduced rapidly. With λ set to 0.98, the cooling rate was medium. For $\lambda=0.99$, the rate of cooling was very slow.

6. Experimentation and Results

The algorithm applied to the problem is given in Table 1. The initial stopping criterion was set at a total unit cost of **265.38**. This value was the optimal solution found by the GA [28]. A lower value was subsequently implemented if this stopping criterion was activated. Another stopping criterion was that the maximum allowed computation time of 12 hours had elapsed.

Figure 5 (a-c) shows the optimisation curves for 3 experiments with different values of λ (0.9, 0.98, 0.99). The maximum number of iterations was set at 200. Experiment 1 showed much random perturbation while the cost quickly reduced. After 84 iterations it became fixed at a local optimum and the programme was stopped after 12 hours. The total unit cost reached was **275.40**. Experiment 2 again showed much random perturbation at the beginning of the iterations. This perturbation was necessary to avoid local optima. After approximately 70 iterations, the cost stabilised noticeably and then reduced gradually until it reached the stopping initial criterion, which is the **expected optimal solution** to this problem. The algorithm terminated at 136 iterations after 1 hour 38 minutes of computation time. Experiment 3 showed much random perturbation although the cost eventually began to stabilise. However before attaining the stopping cost value, the algorithm reached the maximum number of iterations (200) with a total unit cost of **274.06**. The maximum number of iterations could have been increased, but it is expected that the algorithm would have reached maximum computation time before the optimum solution could be found. This is noticeable from high perturbation at the maximum number of iterations.

Experiment 2 was conducted again with a lowered stopping criterion. However no improvement was found even after 12 hours computation time. Therefore the solution found by the GA was accepted as the optimal solution The final inspection configuration is shown in Table 2.

```

Program Simulated Annealing;
Initialise
Set Initial Temperature
Generate Random Configuration
FOR 1 to Iteration, N
  WHILE stopping criterion = FALSE
    IF N > 1 THEN
      generate (new configuration)
    END
    Calculate  $\Delta E$  (new configuration, configuration)
    IF  $\Delta E < 0$  THEN accept new configuration
    ELSE
      IF  $\exp(-\Delta E/T) > \text{Random}(0,1)$ 
      THEN accept new configuration
    END
  END
  Set  $T = \lambda T$ 
END
END.

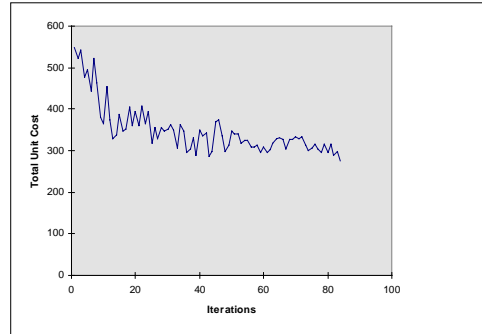
```

Table 1: SA Algorithm for Inspection

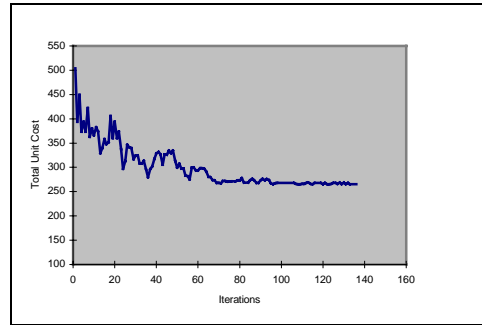
Stage	Binary No.			
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	1	1	0
8	0	0	0	0
9	0	0	1	0

Order of Inspection Stations:
 I2 after Stage 7; I1 followed by I2 after Stage 9

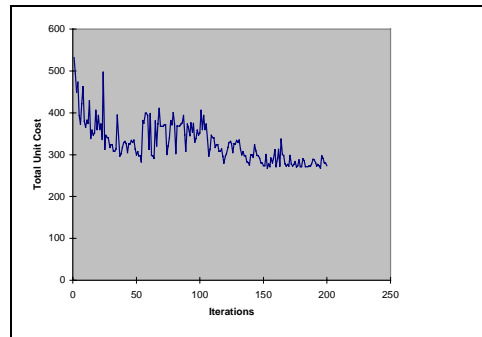
Table 2: Final Result for Inspection Stations Allocation by SA



(a): Experiment 1



(b): Experiment 2



(c): Experiment 3

Figure 5: Optimisation Curve for Inspection Stations Allocation

7. Conclusions

Table 3 shows the best results found by the SA procedure and the Branch-and-Bound technique [24].

The results of the experiment have confirmed that the cooling rate determines the quality of the solutions. If the cooling rate is too low, the configuration can not achieve the optimal solution before it reaches the maximum number of iterations. If the cooling rate is too high, the process could become stuck at a local optimum. Overall SA needed longer computation times compared to the GA [28].

Like a GA, the annealing algorithm is not deterministic. It will produce different answers each time it is run, even with the same problem and set of parameters. This is due to the probabilistic nature of choosing the new configurations and accepting *uphill moves*. In particular, there is no guarantee of obtaining precisely the optimum answer in any annealing operation or the same answer on multiple runs.

What the annealing offers is the possibility of escaping from local optima which is not the same as guaranteeing to find the optimum solution.

Allocation & Sequencing	Total Unit Cost		Percentage Improvement
	Simulated Annealing	Branch-and-Bound	
Experiment 2	265.38	267.52	0.81

Table 3: Comparison between SA and the Branch-and-Bound Technique for Inspection Stations Allocation and Sequencing Problem

REFERENCES

- Kirkpatrick A.S., Gelatt C.D. and Vecchi M.P. (1983) "Optimization by Simulated Annealing", *Science*, Vol. 220, No. 4598, pp. 671-680.
- Metropolis N., Rosenbluth A.W., Rosenbluth M.N., Teller A.H. and Teller E. (1953) "Equation of state calculation by fast computing machines", *Journal Chem. Phys.* Vol. 21, pp. 1087.
- Shahookar K. and Mazumder P. (1991) "VLSI Cell Placement Techniques", *ACM Computing Surveys*, Vol. 23, No. 2, pp. 143-220.
- Murray M., Burr J.B., Stork D.S., Lueng M.T., Boonyanit K., Wolf G.J. and Peterson A.M. (1992) "Deterministic Boltzman Machine VLSI can be Scales Using Multi-Chip Modules". In Fortes J., Lee E. and Meng T. eds., *Applications Specific Array Processors*, pp. 206-217. IEEE Computer Soc. Press, Los Alamitos, CA.
- Bonomi E. and Lutton J.L. (1984) "The N-City Travelling Salesman Problem: Statistical Mechanics and the Metropolis Algorithm", *SIAM Rev.*, Vol. 26, No. 4, pp. 551-568.
- Press W.H., Flannery B.P., Teukolsky S.A. and Vetterling W.T. (1990) "Numerical Recipes in Pascal; The Arts of Scientific Computing", Cambridge Univ. Press, Cambridge, MA.
- Lin C. and Hsueh H.C. (1994) "A New Methodology of Simulated Annealing for

- Optimisation Problems", *Physica A*, No. 205, pp. 367-374.
- Tian P. and Yang Z. (1993) "An Improved Simulated Annealing Algorithm with Genetic Characteristics and the Travelling Salesman Problem", *Journal of Information & Optimisation Sciences*, Vol. 14, No. 3, pp. 241-255.
- Sharpe R. and Marksjo B.S. (1985) "Facility Layout Optimization Using Metropolis Algorithm", *Environment and Planning B*, Vol. 12, pp. 443-453.
- Osman I.H. and Potts C.N. (1989) "Simulated Annealing for Permutation Flow Shop Scheduling", *Management Science*, Vol. 17, pp. 551-557.
- Meller R.D. and Bozer Y.A. (1991) "Solving the Facility Layout Problem with Simulated Annealing", *Technical Report 91-20*, Department of Industrial and Operations Engineering, The University of Michigan.
- Kouvelis P. and Chiang W. (1991) "A Simulated Annealing Procedure for Single Row Layout Problems in Flexible Manufacturing Systems", *International Journal Production Research*, Vol. 30, pp. 717-732.
- Suresh G. and Sahu S. (1993) "Multi Objective Facility Layout Using Simulated Annealing", *International Journal of Production Economics*, Vol. 32, pp. 239-254.
- Chen C.L., Cotruvo N.A. and Baek W. (1995) "A Simulated Annealing Solution to the Cell Formation Problem", *International Journal of Production Research*, Vol. 33, No. 9, pp. 2601-2614.
- Raghu T.S. and Rajendran C. (1995) "Due-Date Setting Methodologies Based on Simulated Annealing - An Experimental Study in Real-Life Job Scheduling", *International Journal of Production Research*, Vol. 33, No. 9, pp. 2533-2554.
- Zegordi S.H., Itoh K. and Enkawa T. (1995) "A Knowledgeable Simulated Annealing Scheme for the Early/Tardy Flow Shop Scheduling Problem", *International Journal of Production Research*, Vol. 33, No. 5, pp. 1449-1466.
- Boctor F.F. (1996) "Resource-constrained Project Scheduling by Simulated Annealing", *International Journal of Production Research*, Vol. 34, No. 8, pp. 2335-2351.
- Meller R.D. and Bozer Y.A. (1996) "A New Simulated Annealing Algorithm for the Facility Layout Problem", *International Journal of Production Research*, Vol. 34, No. 6, pp. 1675-1692.
- Raz T. (1986) "A Survey of Models for Allocating Inspection Effort in Multistage

- Production Systems”, *Journal of Quality Technology*, Vol. 18, No. 4, pp. 239-247.
20. Ballou P.B. and Pazer L.H. (1982) “The Impact of Inspection Fallibility on the Inspection Policy in Serial Production Systems”, *Management Science*, Vol. 28, No. 4, pp. 387-399.
 21. Ballou P.B. and Pazer L.H. (1985) “Process Improvement versus Enhanced Inspection in Optimised Systems”, *Int. Journal of Production Research*, Vol. 23, No. 6, pp. 1233-1245.
 22. Chakravarty A.K. and Shtub A. (1987) “Strategic Allocation of Inspection Effort in a Serial, Multi-Product Production System”, *IIE Transactions*, Vol. 3, pp. 13-22.
 23. Peters M.H. and Williams W.W (1987) “Economic Design of Quality Monitoring Efforts for Multi-Stage Production Systems”, *IIE Transactions*, Vol. 3, pp. 85-87.
 24. Raz T. and Kaspi M. (1991) “Location and Sequencing of Imperfect Inspections in Serial Multistage Production Systems”, *Int. Journal of Production Research*, Vol. 29, No. 8, pp. 1645-1659.
 25. Kang K.S., Ebeling K.A. and La S.H. (1990) “The Optimal Location of Inspection Stations Using a Rule-Based Methodology”, *Computers Ind. Engineering*, Vol. 19, No. 1-4, pp. 272-275.
 26. Aarts E.H.L. and van Laarhoven P.J.M. (1985) “A New Polynomial Time Cooling Schedule”, *Proc. IEEE Int. Conf. on Computer-Aided Design*, Santa Clara, CA, pp. 206-208.
 27. van Laarhoven P.J.M. and Aarts E.H.L. (1987) “Simulated Annealing: Theory and Applications”, Reidel, Dordrecht, Holland.
 28. Hassan A.B. (1997) “*Intelligent Optimisation Techniques for Multi-Stage Manufacturing System*”, PhD. Thesis, University of Wales Cardiff, UK.