Journal of Industrial Technology 9 (2), 2000, 59-73

OPTIMISATION OF MULTI-STAGE PRODUCTION-INSPECTION STATIONS USING GENETIC ALGORITHM

Azmi Hassan

Department of Mechanical Engineering
UKM, 43600, Bangi, Selangor Darul Ehsan
Malaysia

D.T. Pham

Intelligent Systems Laboratory
University of Wales Cardiff, CF2 3TE, United Kingdom

RINGKASAN: *Penentuan lokasi dan susunan stesen pemeriksaan yang optima akan dibentangkan dalam kertas ini. Pengertian optima dalam kes ini melibatkan faktor-faktor seperti kos pemeriksaan, kos membenarkan unit cacat menjadi keluaran, dan kos kegagalan dalaman. Tujuan utama dalam kerja ini adalah untuk membangunkan kaedah pengoptimuman yang effisien yang akan meminimakan fungsi kos kepada permasalahan yang dinyatakan. Kaedah pengoptimuman yang dicadangkan adalah Algorithma Genetik (GA). Pengenalan kepada konsep GA dan pengkodan penyelesaian akan dibincangkan. Pengesetan parameter kepada operator genetik dan bagaimana permulaan populasi penyelesaian juga dihuraikan. Akhir sekali, hasil yang diperolehi akan dibentangkan dan dibandingkan dengan kaedah lazim.*

ABSTRACT: The optimal allocation and sequencing of inspection stations is to be presented in this paper. The notion of optimality in this case encompasses factors such as the cost of inspection, the cost of allowing a defective unit to be output, and the cost of internal failure. The main goal of this work is to develop an efficient optimisation tool which will minimise the cost functions of the stated optimisation problems. The optimisation tool to be considered is Genetic Algorithm (GA). An introduction to the concepts of the GA and its encoding of the solutions to the problems in a genetic form and the evaluation of the resulting genetic code to give the fitness of those solutions will be discussed. The setting of the parameters for the genetic operators and the ways of initialising the solution population are also described. Finally, the results obtained are presented and compared to those produced by conventional methods.

KEYWORDS: Optimisation, Genetic Algorithm, Inspection Stations

INTRODUCTION

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 (involving human inspectors), automated and hybrid (combination of manual and automated systems). Recent technological developments in automated visual inspection, pattern recognition and image processing techniques have led to an increase in the implementation of automated systems. Errors and inconsistencies in manual inspection provide the motivation for this increased implementation. However, 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.

The Genetic Algorithm (GA) is a robust search technique that has proven to be a good solution to some difficult optimisation problems. It was first introduced by Holland (Holland, 1975). The name genetic algorithm derives from the analogy between the representation of a complex structure by means of a vector of components and the genetic structure of a chromosome or *called* string. GA are inherently parallel, general purpose optimisation procedures, which utilises a survival of the fittest approach. New strings of solutions are generated by applying genetic operators to strings in the current population. A standard GA employs reproduction, crossover, mutation and inversion operators. Detail descriptions of how the GA works is available in (Goldberg, 1989). Now GA are used to resolve complicated optimisation problems, example, timetabling, job-shop scheduling, numerical optimisations and games playing.

MULTI-STAGE

A multi-stage production-inspection system can exist in a serial or non-serial configuration. Figure 1 shows these two types of the systems. 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. Only a serial system is to be considered in this paper on inspection stations allocation with the following criteria (Raz and Kaspi, 1991):

i. There are a series of production stages.

ii. Discrete product units of a single type flow in a fixed linear sequence from one stage to the next.

iii. Products flow in batches or lots of size one.

iv. Each production stage consists of a single production operation followed by zero, one or more inspection operations in a fixed sequence.

v. Every production or inspection operation incurs a constant unit processing cost.

vi. In every inspection operation there are two kinds of inspection errors:
- classification of a conforming unit as non-conforming.
- classification of a non-conforming unit as conforming.

These errors will impact on the cost, effectiveness and credibility of quality assurance efforts.

vii. Product units classified as non-conforming are removed from the production line and are disposed of in two ways: scrap or rework.

viii. Delivering a non-conforming unit to the customer will cause a penalty cost.

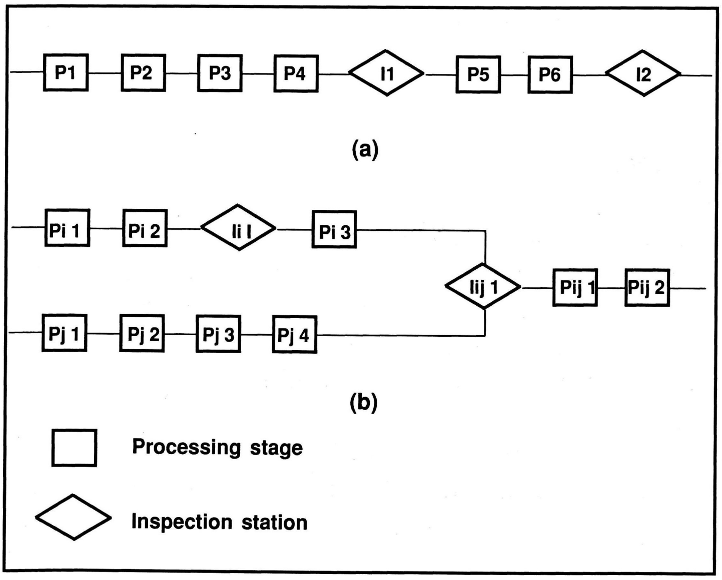


Figure 1(a). *Serial Production-inspection System (b) Non-serial Production-inspection System*

The optimal design of inspection stations has been studied since the 1960s. Raz (1986) 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 (1982 and 1985) developed a production-inspection model allowing for inspection errors in a serial production system. Chakravarty and Shtub (1987) suggested a shortest-path heuristic method to determine the strategic location of inspection activities and the production lot-sizes. Peters and Williams (1987) 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 (1991). 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 (1986). Kang *et al.* (1990) proposed a Rule-based technique to determine near-optimal inspection stations. The present work extends the work of Raz and Kaspi (1991) to global optimisation by the use of AI techniques.

PROBLEM STATEMENT

The inspection stations allocation model is formulated with the objective of minimising the total cost per product unit. The total cost includes the unit cost of inspection and cost of defective items. There are two kinds of cost of defective items. One is the rework and replacement cost before the defective items are released from the company. The other is the penalty cost for each non-conforming unit reaching the customer.

The problem is to decide which inspection operations will be performed immediately following each production stage. The constraints on the optimisation problem are based on a required outgoing fraction of non-conforming units and the number of inspection operations.

In this paper, the multi-stage production-inspection model developed by Raz and Kaspi (1991) is considered. The model is called the Transfer Functions Model (TFM). The TFM provides a unified framework for the analysis of multi-stage systems with different types of production and inspection operations. This model facilitates the implementation of the calculations required to find the optimal solution. The TFM at each stage, whether a production or inspection stage, is described by two transfer functions: the Cost Transfer Function and the Quality Transfer Function.

The Cost Transfer Function (CTF) relates the cumulative unit costs before and after completion of the production or inspection operation denoted by C_i and C_o respectively. The Quality Transfer Function (QTF) of an operation relates the probabilities q_i and q_o that a unit is non-conforming before and after the operation respectively. Figure 2 shows the TFM for a single processing and inspection stage. The description of the parameters could be found in (1991).

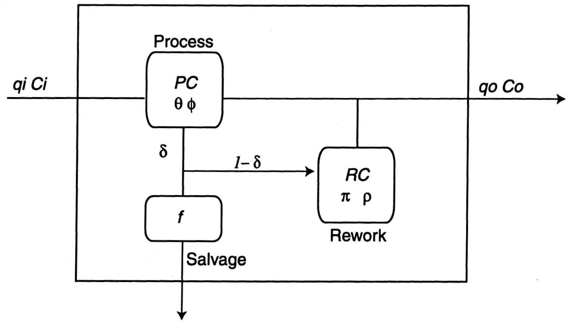


Figure 2. *Transfer Functions Model for a Single Processing and Inspection Stage*

The following multi-stage problem is to be addressed:

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

The total cost $C_{TS]}$ of a single stage is given by Raz and Kaspi (1991):

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

where

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

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

s - 0 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:

$$CT_{[s+1]} = C_{o[s+1]} + q_{o[s+1]}.PC_{[s+2]}$$

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 $C_{T[p]}$:

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

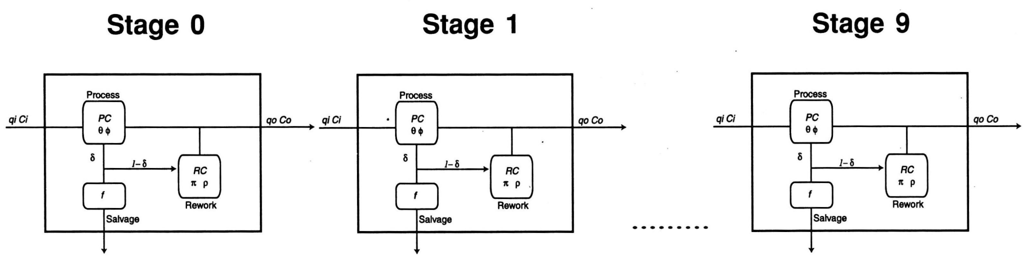


Figure 3. *TFM for the Whole Production Line*

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.

IMPLEMENTATION

This section will describe how to represent solutions in a binary form and the evaluation of the fitness function. Finally the results of optimisation experiments will be presented and compared to those obtained by Raz and Kaspi (1991) with the Branch-and-Bound technique.

Representation

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. In order to represent all 16 possible configurations in each stage, 4 bits are used. Table 1 lists all possible 4-bit strings and the corresponding configurations. There are 10 processing stages so 40 bits are required to represent all possible configurations. An example of 40-bit string representing a particular solution to the problem is shown in Figure 4.

Table 1. Binary Representation of Inspection Stations in a Single Stage

Binary Value (4 bits at each Stage)	Decimal Value	Order of Inspection Stations Stations
0000	0	No Inspection
0001	1	I1
0010	2	I1 - I2
0011	3	I1 - I2 - I3
0100	4	I1 - I3
0101	5	I1 - I3 - I2
0110	6	I2
0111	7	I2 - I1
1000	8	I2 - I1 - I3
1001	9	I2 - I3
1010	10	I2 - I3 - I1
1011	11	13
1100	12	I3 - I1
1101	13	I3 - I1 - I2
1110	14	I3 - I2
1111	15	I3 - I2 - I1

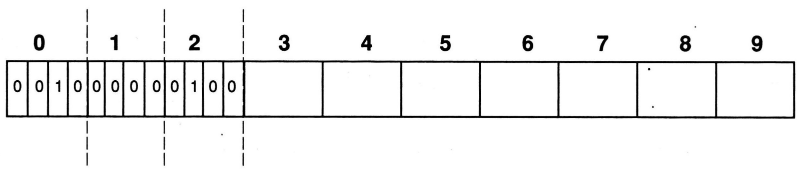


Figure 4. *Binary Representation for 10 Processing Stages*

Note that in this representation no invalid solution would occur and all strings correspond to possible inspection station configurations.

Fitness Evaluation

The inverse of the objective function of CT is the fitness function for the GA. The fitness value will correspond to the inverse of the total unit cost of manufacturing.

Table 2 shows an example of fitness evaluation for a given string. The fitness evaluation routine is coded in Turbo Pascal and linked to the main GA programme, also implemented in Turbo Pascal.

Table 2. Fitness Evaluation

Stage	Inspection Order	$C_{o[s]}$	$q_{o[s]}$	$C_{T[s]}$
0	No Inspection	10.00	9.000×10^{-2}	11.80
1	No Inspection	30.00	1.537×10^{-1}	32.31
2	No Inspection	45.00	2.553×10^{-1}	57.76
3	I1	104.12	2.450×10^{-2}	104.85
4	No Inspection	134.12	8.303×10^{-2}	136.61
5	No Inspection	164.12	1.380×10^{-1}	165.50
6	No Inspection	174.12	1.811×10^{-1}	181.36
7	No Inspection	214.12	2.221×10^{-1}	218.56
8	I2	254.23	1.859×10^{-2}	254.56
9	No Inspection	274.23	6.766×10^{-2}	274.23
	PEN	274.23	6.766×10^{-2}	308.07
FINAL				**308.07**

Fitness = 1/308.7

EXPERIMENTATION AND RESULTS

The algorithm generates the initial population randomly and performs the reproduction, crossover and mutation operations according to the flow diagram of Figure 5. Ten experiments were carried out with different settings of *GA* parameters. The maximum number of generations was 500. Table 3 shows the empirically chosen *GA* parameter settings and the final results of the experimentation. Figures 6(a-j) show how the total unit cost C_T reduced as the optimisation progressed. Experiment 10 gave the best result with C_T equal to 265.38. Figure 7 shows the inspection stations allocation produced from Experiment 10.

Further experiments were carried out by changing the *GA* parameters and increasing the number of generations. However, no improvement was obtained and thus the result from Experiment 10 can be regarded as the optimum solution found by the *GA* to the given problem.

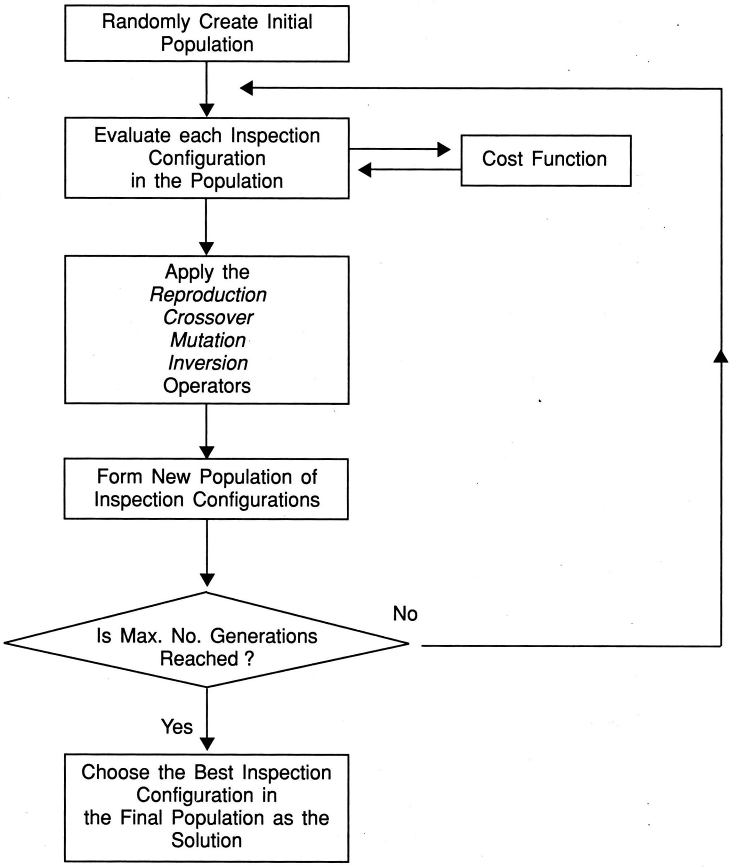
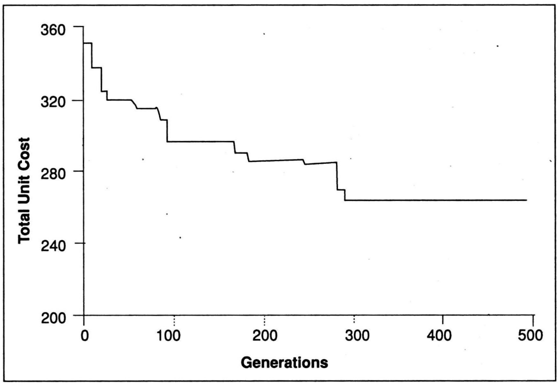


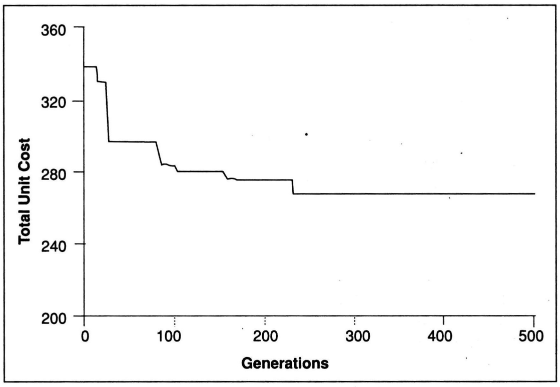
Figure 5. *Structure of GA Implemented for Inspection Stations Allocation*

Table 3. Final Results for Inspection Stations Allocation by GA

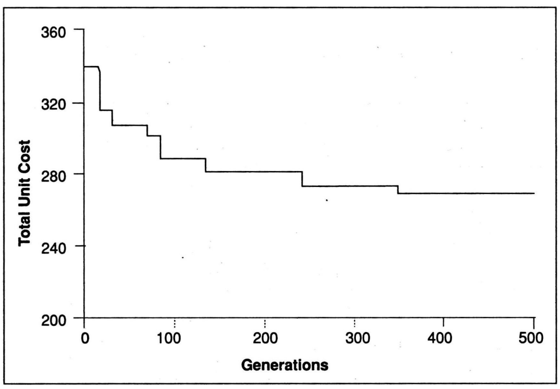
No. of Experiment	Population Size	Crossover Rate	Inversion Rate	Mutation Rate	**Total Unit Cost**
1	50	0.9	0.2	0.04	267.36
2	100	0.9	0.2	0.04	266.30
3	150	0.9	0.2	0.04	265.41
4	200	0.9	0.2	0.04	270.73
5	400	0.9	0.2	0.04	265.41
6	50	0.9	0.3	0.04	271.72
7	50	0.9	0.2	0.05	265.41
8	50	0.8	0.2	0.02	268.08
9	50	0.8	0.2	0.03	275.49
10	100	0.8	0.3	0.03	**265.38**



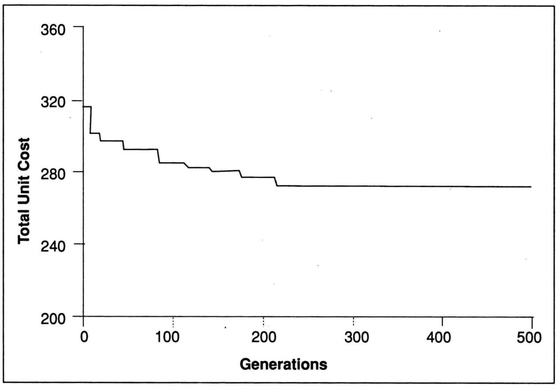
(a): Experiment 1



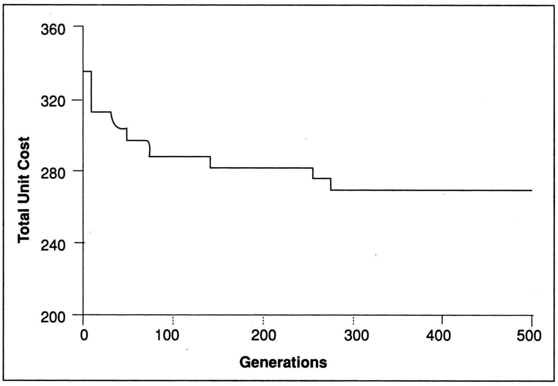
(b): Experiment 2



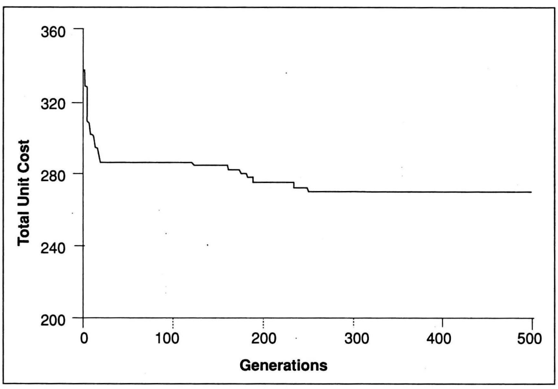
(c): Experiment 3



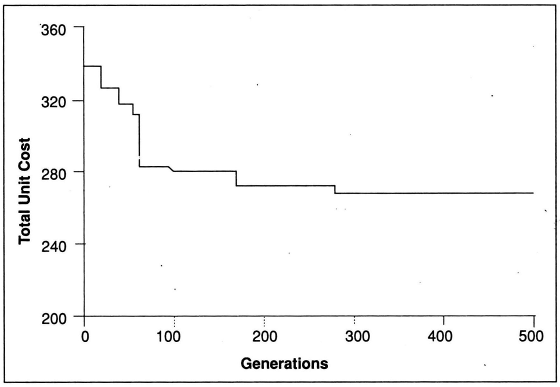
(d): Experiment 4



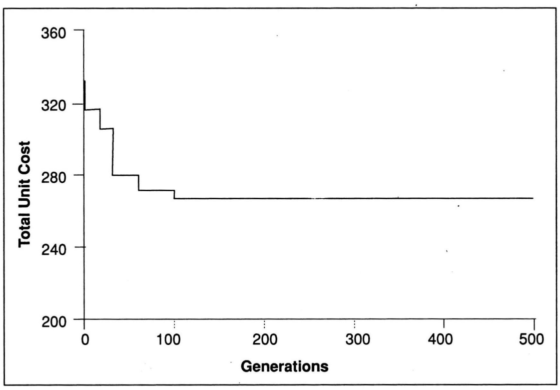
(e): Experiment 5



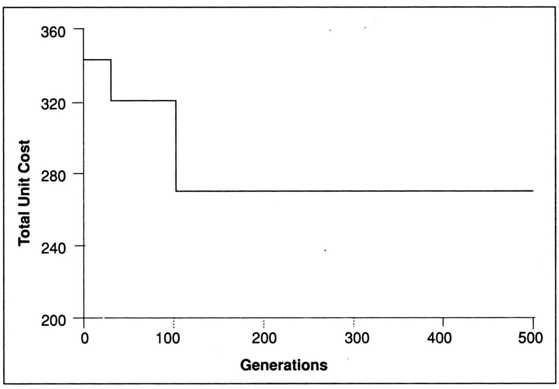
(f): Experiment 6



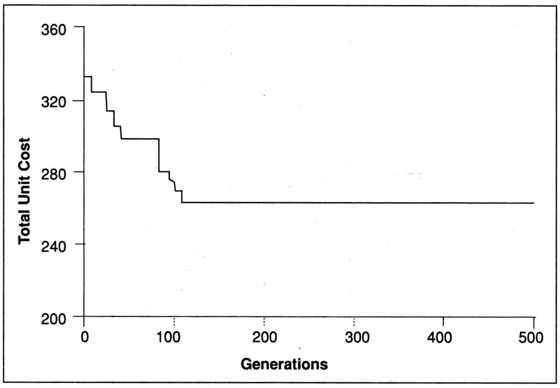
(g): Experiment 7



(h): Experiment 8



(i): Experiment 9



(j): Experiment 10

Figure 6. *Optimisation Curves for Inspection Stations Allocation*

Order of inspection stations:

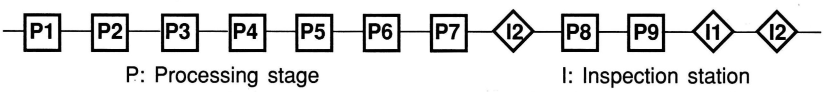


Figure 7. *Inspection Stations Allocation*

DISCUSSION

It can be seen that the *GA* is a powerful optimisation technique provided the appropriate set-up of the genetic operators was applied. The *GA* managed to produce the optimum solution even when it had no initial knowledge about inspection sequencing and location. The total unit costs were high at the initial generations, then reduced quickly to near the optimum level after relatively few generations. The final results obtained by the *GA* procedure is 265.38 while that of the Branch-and-Bound technique was 267.52 an improvement of 0.8%.

The *GA* used a binary string to represent the configuration of the inspection stations within all the processing stages. The evaluation of each string was based on the total unit cost (see steps 1 - 3). This is the only knowledge required by the *GA* unlike the Branch-and-Bound technique and other heuristic methods where initial knowledge of the configuration is required and the new configuration is decided from one processing stage to another. This would allow the possibility of discarding the optimum solution at the initial stage of the computation.

Step 1: *Obtain a string*

Stage	Sub-strings			
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	1
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	1	1	0
9	0	0	0	0

Step 2: *Read the inspection parameters.*

-Read the value of inspection parameters (PC, RC, θ, π, etc.)
-Set *PEN* = 500
-Initialise $C_{i[0]} = 0$
-Initialise $q_{o[0]} = 0$

Step 3: *Calculate the Total Unit Cost (Inverse Fitness Value)*

RERERENCES

Ballou P.B. and Pazer L.H. (1982). The Impact of Inspection Fallibility on the Inspection Policy in Serial Production Systems, *Management Science,* **28** (**4**): pp 387-399.

Ballou P.B. and Pazer L.H. (1985). Process Improvement versus Enhanced Inspection in Optimised Systems, *Int. Journal of Production Research,* **23** (**6**): pp 1233-1245.

Chakravarty A.K. and Shtub A. (1987). Strategic Allocation of Inspection Effort in a Serial, Multi-Product Production System, *IIE Transactions,* **3**: pp 13-22.

Goldberg D.E. (1989) Genetic Algorithms in Search, Optimisation and Machine Learning, Addison-Wesley, Reading, MA.

Holland J.H. (1975) Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, Michigan, USA, First Edition (Second edition, 1992).

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,* **19** (**1-4**): pp 272-275.

Peters M.H. and Williams W.W (1987). Economic Design of Quality Monitoring Efforts for Multi-Stage Production Systems, *IIE Transactions,* **3**: pp 85-87.

Raz T. (1986). A Survey of Models for Allocating Inspection Effort in Multistage Production Systems, *Journal of Quality Technology,* **18** (**4**): pp 239-247.

Raz T. and Kaspi M. (1991) Location and Sequencing of Imperfect Inspections in Serial Multistage Production Systems, *Int. Journal of Production Research,* **29** (**8**): pp 1645-1659.