



Journal of Applied Sciences

ISSN 1812-5654

science
alert

ANSI*net*
an open access publisher
<http://ansinet.com>

Design, Development and Performance Optimization of a New Artificial Intelligent Controlled Multiple-beam Optical Scanning Module

¹S.P. Koh, ²I.B. Aris, ¹V.K. Ramachandaramurthy,

²S.M. Bashi and ²M.H. Marhaban

¹College of Engineering, Universiti Tenaga Nasional (UNITEN), km 7,
Kajang-Puchong Road, 43009 Kajang, Selangor Darul Ehsan, Malaysia

²Department of EE Engineering, Faculty of Engineering, Universiti Putra Malaysia,
43400 Serdang, Selangor, Malaysia

Abstract: This research presents a new approach to optimise the performance of a multiple-beam optical scanning system in terms of its marking combinations and speed, using Genetic Algorithm (GA). The problem has been decomposed into two sub problems; task segregation, where the marking tasks need to be segregated and assigned for each scanner head and path planning where the best combinatorial paths for each scanner are determined in order to minimise the total motion of marking time. The knowledge acquired by the process is interpreted and mapped into vectors, which are kept in the database and used by the system to guide its reasoning process. The main motivation for this study is to introduce and evaluate an advance new customized GA. Comparison results of different combinatorial operators and tests with different probability factors are shown. Also, proposed are the new modifications to existing crossover operator called DPPC (Dynamic Pre-Populated Crossover) together with modification of a simple crossover selection method, called BCS (Bi-Cycle Selection Method). The performance of the new operator called GA_INSP (GA Inspection Module) for a better evolutionary approach to the time-based problem has been discussed in the study. The representation approach has been implemented via a computer program in order to achieve optimized marking performance. This algorithm has been tested and implemented successfully via a dual-beam optical scanning module.

Key words: Multiple-head optical scanner, genetic algorithm

INTRODUCTION

The optimization of marking problem is the task of finding the best marking sequences through a given set of objects. For a single-head marking system, the approach is similar to the Traveling Salesman Problem (TSP). The TSP is the task of finding the shortest possible tour through a given set of cities. Surveys of works on the TSP can be found by (Bentley, 1992), Fogel, 1993a,b, Yang (1997) and Mitchell *et al.* (2000). This method has been further enhanced in order to suit complicated applications such as the multiple-head marking system.

The study of algorithms to achieve this practical goal has been carried out by applying many different methods from many areas such as heuristic methods (Bentley, 1992), neural networks (Haykin, 1999) and genetic algorithms. Goldberg (1997) and Goldberg and Lingle (1985) explain why the Evolutionary Algorithm (EA) performs an effective search compared to other methods.

The EA presents a continual improvement using the pair selection and mutation, working as a local search where the mutation operator slightly modifies a solution. If this new solution is better than previous ones, it will be accepted with high probability by the selection mechanism. On the other hand, the pair selection and crossover avoids the process being trapped in a local minimum, executing an intelligent jump to another search space region.

The usual genetic operators of crossover, reproduction and mutation on binary strings are insufficient to solve the problem. Grefenstette (1987) pointed out that genetic algorithms are qualified to identify high performance regions of the search space and he recommends. It may be useful to invoke a local search routine to optimize the members of the final population. Syrakow and Szczerbicka (1994) have proposed a local search using a hill-climbing strategy. If the local search does not obtain an optimum value, a backtracking process will be applied to execute a new genetic simulation.

Ridao *et al.* (2001) have presented a method to minimize the total motion time of two manipulators along their paths, avoiding collision regardless of the accuracy of the dynamic model used. The evolutionary algorithm is used to obtain collision-free motion plans in a multi-robot environment. Global optimization algorithms imitating certain principles of nature such as simulated annealing and the field of evolutionary algorithms have proven to be useful tools for the optimization of high dimensional and highly nonlinear problems.

As Mitchell *et al.* (2000, 2003) point out some types of encoding require specially defined crossover and mutation operators for example the encoding used in the genetic programming, or encoding for problems like the Traveling Salesman Problem (TSP) in which the tasks to find a correct ordering for a collection of objects. From here, the possibilities of developing new alternative GA operators that work well with the encoding scheme are explored.

In this study, a method is presented to minimize the total motion time taken for a dual-head marking operation. The method uses a coordination diagram and trajectories planner that can easily be implemented in the industrial machines. For a multiple-head marking system, the objects from a group need to be sorted and divided in order to be marked by each of the scanner head. Figure 1 shows a randomization approach to find the solution of the dual-head marking optimality through a set of random combinations of both heads.

A part of this research involves developing a machine-learning system and program via genetic algorithm that is capable of performing the following analysis:

- Reliably and consistently learn and repeat a given tasks.

- Ability to look for the optimum multi-heads marking sequences via genetic algorithm.
- Incorporating new alternative operators for efficient evolutionary solutions.

The other part of this study consists of constructing a prototype, which consists of a multiple head optical scanning module and a system controller that is capable of performing the given tasks. This prototype is then interfaced with the marking program to form a complete system as shown in Fig. 2.

PROBLEM STATEMENTS

The problem can be stated as: Given two scanning heads H_1 and H_2 , a set of known fixed objects coordinates with the initial and final configurations of H_1 and H_2 , find a coordinated motion plan for the scanning head from their initial configuration to their final configuration, optimizing the overall time taken for the dual-beam optical scanning module as shown in Fig. 3.

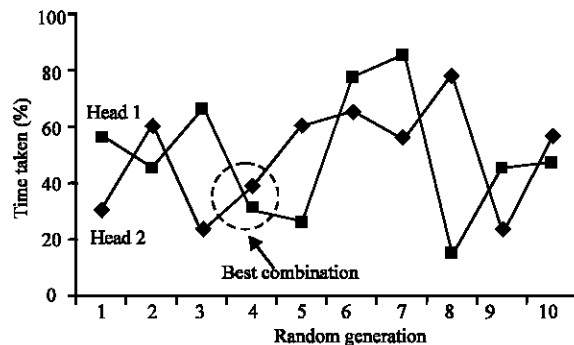


Fig. 1: Random marking combinations

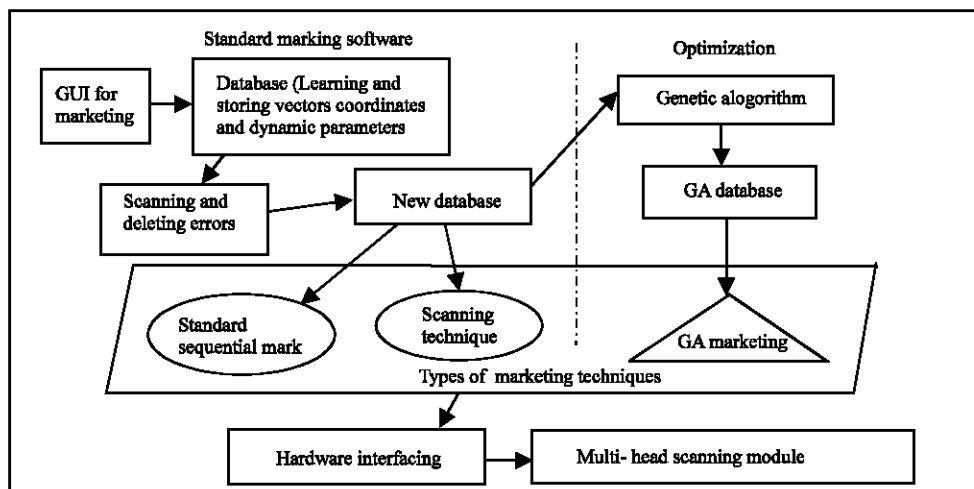


Fig. 2: Intelligent multi-head scanning system

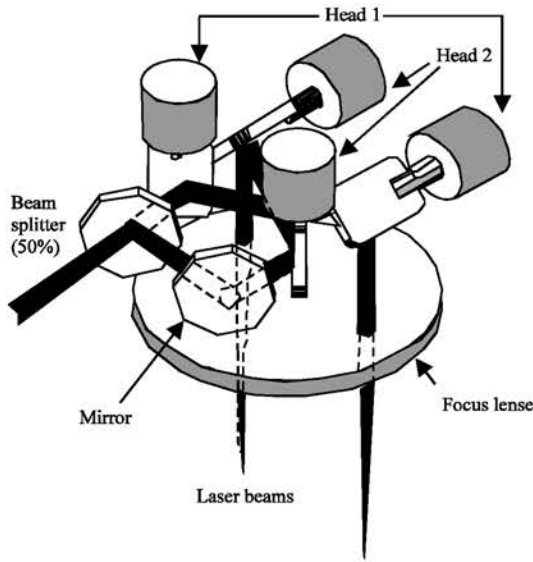


Fig. 3: Two-dimensional dual-beam laser scanner

To give an idea of the complexity of the problem, let's consider a number of n coordination points and two origin points for each scanning head fixed at positions (x_1, y_1) and (x_2, y_2) . The solution adopted here is to consider variable-length chromosomes. The length of the chromosomes, h_1 and h_2 define the number of synchronization points of the sequence for each scanning head where $h_1 + h_2 = n$. The number of potential solutions for this problem can be computed as in Eq. 1.

$$q = \sum_{m=0}^{m=n} (n-m)!m! \quad (1)$$

Each solution for this problem can be obtained as two random successions of synchronization points in such a way that a synchronization point is the pair (x_i, y_i) coordinate and the probability for each solution is given in Eq. 2.

$$b = \sum_{m=0}^{m=n} {}^nC_m \left(\frac{1}{2}\right)^m \left(\frac{1}{2}\right)^{n-m} \quad (2)$$

For the case of h-scanning head with n number of coordination points, the number of possible solutions are given in Table 1.

Table 1: Number of possible solutions

No. of objects (n)	No. of scanning head (h)	
	1	2
5	120	312
10	3628800	8254080
50	3.04E+64	6.21E+64

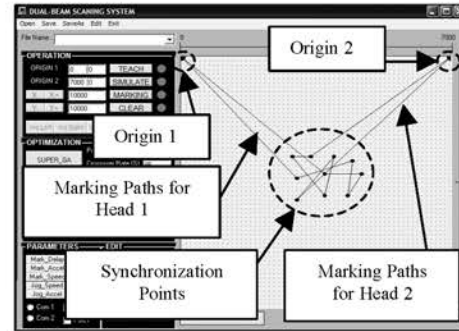


Fig. 4: Non-optimized dual-head marking performance

A special dedicated software called Dual Beam Scanning System was developed by the author to simulate and evaluate the possible solution for this problem. This software was written in VB/C++ and the databases used are Microsoft Access and Excel. A possible non-optimized solution for $n = 10$ is shown in Fig. 4.

GENETIC ALGORITHM OPTIMISATION METHOD

The basic genetic operators are reproduction, crossover and mutation. Our reproduction operator is a standard method called tournament selection. The crossover and mutation operators are based on domain-specific knowledge (Kennedy and Spears, 1998).

A normalized path representation is used to represent an object, which is represented by a point. Let n be the length of the chromosome. The initial and the last points of the sequence are the path origins of each scanning head, (x_1, y_1) and (x_2, y_2) . Between the initial and the last vertices, an individual will be a set of synchronization points $\{P_i\}$. The list of the synchronization points will be segregated to have a dual-head marking combination where points to be marked by the first head, $H_1 \in \{P_i \mid 1 < i < n, P_i > 0\}$ and the second head, $H_2 \in \{P_i \mid 1 < i < n, P_i < 0\}$. That is, a list $\{-3, 2, -1, -4, 5, 7, 6\}$ means the marking sequences for both scanning heads are given by $\{3, 1, 4\}$ and $\{2, 5, 7, 6\}$, respectively. For the chromosomal representation, unrepeated generation for the string of genes is required as each of the objects can only be marked once. A population consisting of a set of chromosomes is shown in Fig. 5. The binary numbers represent the sign of each number in the chromosome.

A new population must now be developed from this initial population by genetic processes: reproduction based on fitness, crossover and mutation.

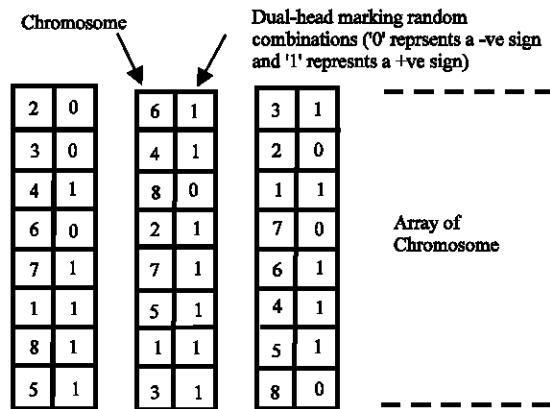


Fig. 5: Structure of an initial population

NEW ALTERNATIVE GA OPERATORS

A set of chromosomes is selected at the reproduction stage based on natural selection. Thus members of the population are chosen for reproduction on the basis of their fitness defined according to specified criteria. One solution has been the partially matched crossover system (Goldberg and Lingle, 1985). This technique reduces the effect of crossover by matching a section in each individual and then performing a limited crossover. This technique is restrictive as there might be repeated numbers in the list, causing redundant marking operations, as depicted in Fig. 6a.

A new form of operator includes a cleanup operation after crossover, which has out-performed of the standard evolutionary algorithm mechanism. This operator converges faster on all given comparisons, producing better results in every instance. The operator Cleanup has been specifically designed for use in real world evolutionary TSP systems (Yang, 1997).

DPPC (DYNAMIC PRE-POPULATED CROSSOVER)

In this study, a new alternative crossover operator named the Dynamic Pre-Populated Crossover (DPPC) has been introduced and its performance evaluated for a variety of input sizes. The results indicate that the use of the proposed operator has a marked influence on the time necessary to converge on the best combination of multiple marking heads.

In the Cleanup template, the replication of numbers within the string is tested for by a referenced template, which is constructed at the very initialization of the program. The new offspring and the genetically repaired offspring are depicted in Fig. 6b. By using the

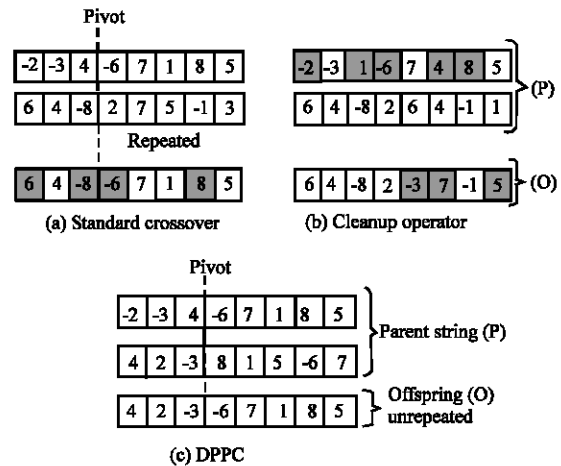


Fig. 6: Dynamic pre-populated crossover operation (DPPC)

newly proposed DPPC operator, an earlier checking has been executed at the beginning of each population before crossover. Initially, the pivot point for crossover, which is dynamically located, will be determined. Then, the components at each side of the pivot point will be randomly shuffled only in their own group, as depicted in Fig. 6c. Similar process applies to the other chromosome before the crossover takes place.

Therefore, the cleanup and reintroduction of the numbers excluded in the string, in order to genetically repair the offsprings are unnecessary in DPPC. With this new proposed operator, the population is generated in a systematic way instead of a purely random selection. Obviously, this will save most of the processing time, neglecting the time lost on repairing the genes.

BI-CYCLE FITNESS SELECTIONS

In most implementations, crossover is performed by choosing and removing a random subset of genes in a chromosome and then exchanging it with a corresponding set removed from a different chromosome. There will be a higher probability of more of the fittest chromosomes in the mating pool. This process of selection is complex and is based on a process, which simulates the use of a roulette wheel. The percentage of the roulette wheel that is allocated to a particular string is directly proportional to the fitness of the chromosome. By applying this procedure to initial population, a new generation is produced. Each chromosome with the highest fitness value will occupy the largest area, while the chromosome with a lower value takes the slot of smaller sector. In most of the cases, the best chromosome will be chosen repeatedly as if it covers a very large sector.

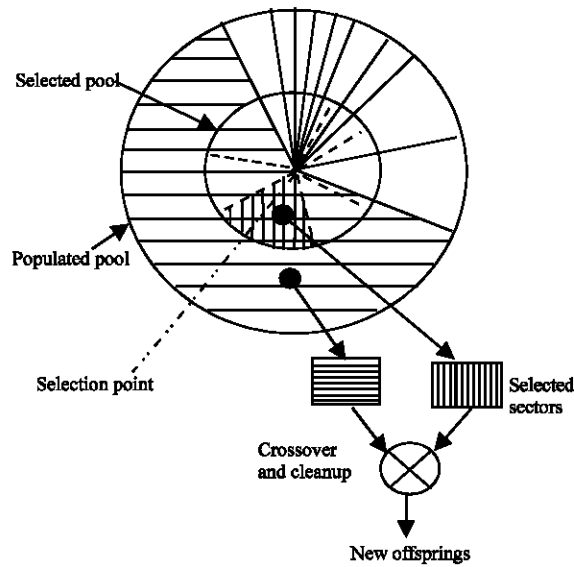


Fig. 7: Bi-cycle fitness selections

In this study, Bi-Cycle Crossover technique has been introduced and applied to increase the efficiency, as it helps to reduce the number of replications during the crossover. The best chromosomes from the Populated Pool will be transferred to the Selected Pool Cycle as depicted in Fig. 7. This will allow a chance for the other chromosomes to expand into larger sectors to be selected. A selection point will select the pair of chromosomes from both cycles for crossover. The use of the new selection method reduces the number of redundant offsprings, reducing the required number of generation evaluations for convergence.

GA INSPECTION MODULE

The main objective of the implementation is to develop a scalable and fault-tolerant algorithm. Generally, the main GA program will evaluate and look for the final optimal solution. During the GA process, the local optimal solutions will be tested and checked via the GA Inspection loops. The process will be managed and controlled by a timer in order to operate either in serial or parallel via a processing system. In the dual-head scanning operation, the solutions for each head will be extracted and transferred into two separate sub-databases as depicted in Fig. 8. The data will be reevaluated by the inspection loops and a better solution, consisting of better genes will be shared to the main GA, reducing the required searching space. This method is relatively easy to implement and a significant speedup can be expected if the communications cost does not dominate the computation cost. More convincingly, the shortest tour

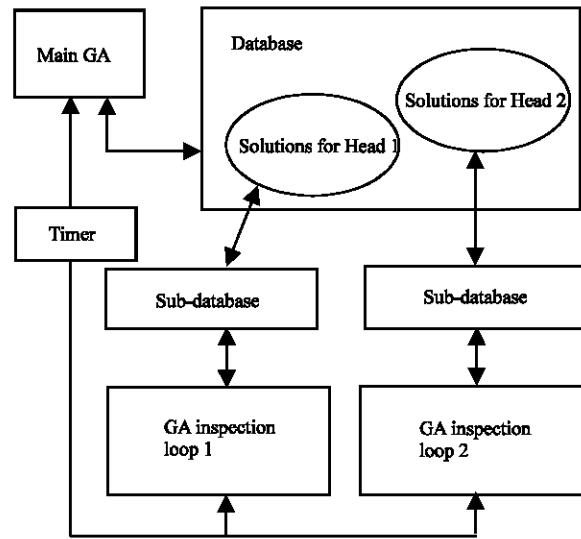


Fig. 8: GA inspection module

produced in this approach in every instance, outperformed that of the other standard evolutionary algorithm mechanism.

PROBLEM FORMULATIONS

This section links the marking durations of both marking heads into merit function, used to design the marking sequence via GA optimisation. Developing a fast, accurate means of computing the time profile of the marking operation is important in GA to design the marking sequences. If there is no unnecessary delay by the system, then the total time taken for the multiple marking heads is given by the summation of traveling time (t_r), stopping time delay (t_d) and marking time (t_m) for the n number of objects and also need to be considered the starting time (t_s) and the ending time (t_e) for the system to return back to the origin as defined in Eq. 3.

$$T = t_s + t_e + \sum_{i=0}^{i=n} (t_r + t_d + t_m) \quad (3)$$

The degree of fitness of a solution is done by defining a proper fitness function and assigning a value to it. In this study, marking durations of each head and the time gap difference between the marking heads are to be minimised. The overall fitness function for these applications contains terms that quantify the fitness of the total distance or time taken (M_m) and the time equality (M_{eq}) for the m number of marking heads. The general fitness function is defined by Eq. 4.

$$M = \prod_{i=1}^{i=m} M_{hi} M_{eqik} \quad (4)$$

where, $j, k \in \{1, \dots, m \mid j \neq k, j < k\}$. Thus, for a dual-head marking system, the fitness function is defined by Eq. 5, as following:

$$M = M_{h1} M_{h2} M_{eq12} \quad (5)$$

where:

$$M_{h1} = \exp[-S(T_{mark1} - T_{min})^q] \quad (6)$$

$$M_{h2} = \exp[-S(T_{mark2} - T_{min})^q] \quad (7)$$

$$M_{eq12} = \exp[-S|T_{mark1} - T_{mark2}|^q] \quad (8)$$

In Eq. 6-8, the exponential function is chosen since the fitness function needs to peak sharply at $T_{mark} = T_{min}$ which indicates a good system during the GA optimisation process. The exponential accomplishes this nicely, but functions other than the exponential may have been chosen. The coefficient S determines the sensitivity of the fitness function to the time constraint. That is, the smaller the value of S , the broader the exponent function becomes. In this example, S was set to 0.01. The coefficient q is set to 0.5. This setting is to fine-tune the curve of the exponential fitness graph. In Equation 8, the difference of time taken between the two marking heads is given as $T_{mark1} - T_{mark2}$. As the time gap of both marking heads becomes narrower, approaching zero, M_{eq} increases substantially. The fitness function rewards those results, which tend to increase the value of M and penalise the unfit results as the GA optimisation searches throughout both the discrete and continuous parameter space.

EXPERIMENTAL RESULTS

Experiments were carried out using three models; one with DPPC and the standard roulette-wheel selection (RWS), DPPC and the Bi-cycle selection (BCS) and one with the GA inspection module (GA_INSP). The following is a summary of the new algorithm:

Step 1: Initializing the population by using the random populated with the pre-determined pivot point. A binary string determines the combinations of sequences for each of the dual-head.

Step 2: Applying the new developed fitness function and implementing the below methods of selection;

- Standard roulette wheel selection
- Bi-Cycle Selection method

Step 3: Reproducing the new generation by using tournament selection. An elitist selection rule is also applied where the previous generation will be preserved if the new one is worse. GA_INSP will be applied to check the results.

Step 4: The best fitness is determined if the current fitness has not been improved for more than a preset number of generations. Else, step 1 will resume with the current best marking combination.

Initially the algorithm was tested with a crossover probability of 80% and a mutation probability of 5% percent. The standard population size was set at 100 individuals and 10 objects to be marked. The following simulations (Fig. 9 to 12) show the optimized performance for different types of objects coordination in order to test the reliability and functionality of the developed algorithm.

As can be observed in Fig. 13, shorter time is required for DPPC over the others to achieve convergence. At some earlier points, better individuals were eventually born and the algorithm was resuming the improvement process. Figure 14 shows the percentage of contribution from the GA operators for evolutions. Notice that, the crossover operator has played a major role for evolutions to a better offspring, achieving contribution up to almost 90%.

Experiments were further carried out with more data to test the number of generations required to achieve optimality. The data used was a set of marking problems, principally 30, 50 and 70 objects to be marked. Throughout these experiments the mutation rate was fixed at 10% with the initial population of 100. From Fig. 15, it is clear how the operation of the system varies depending on the inclusion of the DPPC, RWS, BCS and GA_INSP. This chart compares the number of generations required to perform the best marking combinations of both marking heads. Overall, DPPC+BCS+GA_INSP out-performs the others by up to at least 10%.

The result of Bi-Cycle Selection Method with the standard Roulette Wheel Selection Method is now compared. From Fig. 16, the use of the Bi-Cycle selection method reduces redundant combinations of chromosomes with higher percentage of fitness, minimising the number of offsprings required for convergence.

All these experiments indicate that the operators; DPPC, GA_INSP and BCS, having a very beneficial influence upon the operation of evolutionary algorithms for the multiple-head marking problems. It is observed that the genetic algorithm works efficiently with the new GA operators, achieving solution better than the standard

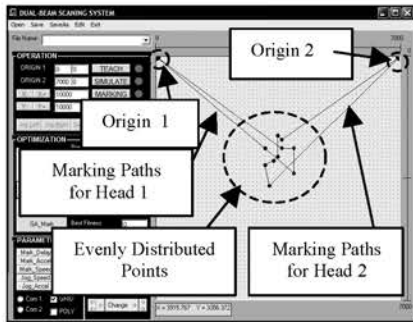


Fig. 9: Evenly distributed objects coordination

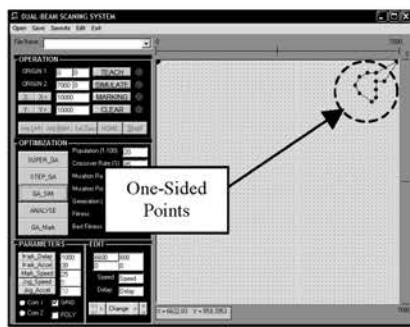


Fig. 10: One-sided load objects coordination

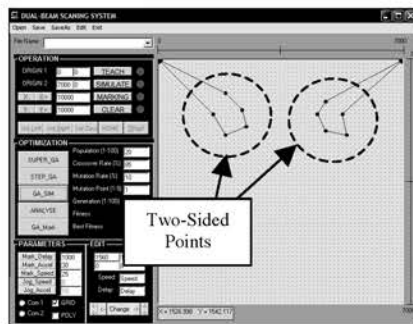


Fig. 11: Two-sided load objects coordination

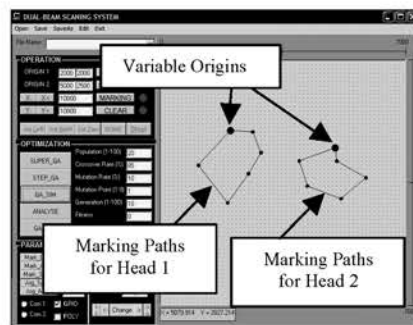


Fig. 12: Two-sided objects coordination with variable pre-determined origins

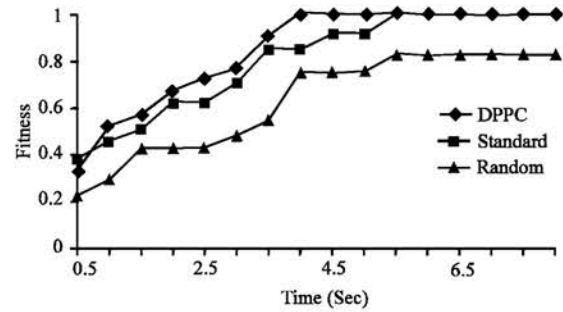


Fig. 13: Evolution of the algorithm with DPPC and standard operators

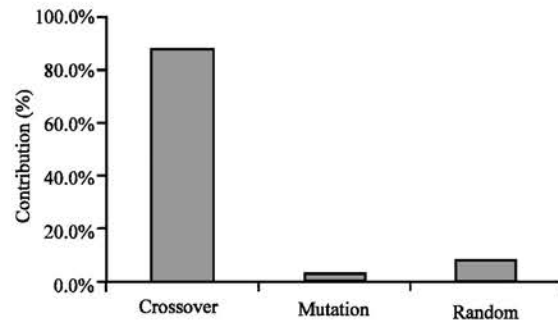


Fig. 14: Percentage of contribution

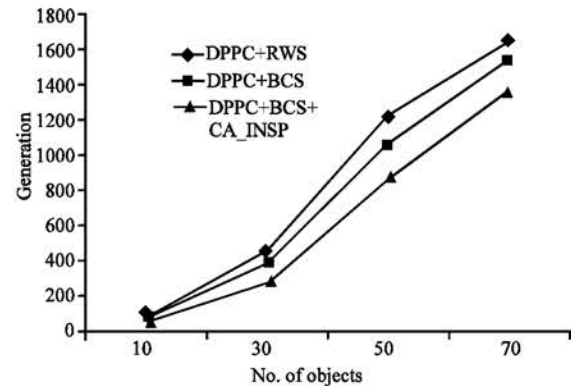


Fig. 15: Effect of DPPC, BCS, GA_INSP operator

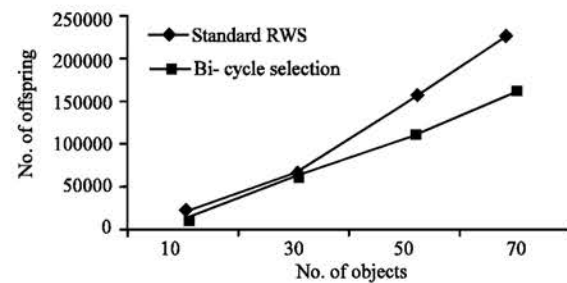


Fig. 16: Effect of the bi-cycle selection method

operation. It improves the efficiency of the system by reducing unwanted time delay due to complicated marking sequences arrangement.

CONCLUSIONS

The performance optimization of the single-head manipulation machines is based on the TSP algorithm. In this research, the extension of this algorithm was done and results indicate a promising solution for the dual head marking problems. The dual-head marking operation proposed performs better, as the time taken to mark can be less than half of the time required for a single head marking operation. This paper models a new GA fitness function for a multiple-head marking problem. The simulation results indicate that the algorithm is able to segregate and assign the tasks for each marking head and also able to find the shortest marking path for different types of objects coordination. Besides that, the implementation of the new genetic operators helps to converge faster, producing better results at every instance.

REFERENCES

- Bentley, J.L., 1992. Fast algorithms for geometric traveling salesman problems. *ORSA J. Computing*, 4: 387-411.
- Fogel, D.B., 1993a. Applying evolutionary programming to selected traveling salesman problems. *Intl. J. Cybernetics and Syst.*, 24: 27-36.
- Fogel, D.B., 1993b. Empirical Estimation of the Computation Required to Discover Approximate Solutions to the Traveling Salesman Problem Using Evolutionary Programming. *Proceedings of 2nd Annual Conference on Evolutionary Programming*, pp: 56-61.
- Goldberg, D. and R. Lingle, 1985. Alleles, loci and the tsp. In *proceedings of an international conference on genetic algorithms and their applications*, pp: 154-159.
- Goldberg, D.E., 1997. *The Design of Innovation: Lessons From Genetic Algorithms, Lessons for the Real World*. Internal Report 98004, Illinois Genetic Algorithms Laboratory, Department of General Engineering, University of Illinois at Urbana-Champaign, Illinois.
- Grefenstette, J.J., 1987. Incorporating Problem Specific Knowledge into Genetic Algorithms: Genetic algorithms and simulated annealing. Davis, L. (Eds.), Morgan Kauffmann Publishers, San Francisco, pp: 42-46.
- Haykin, S., 1999. *Neural Network: A Comprehensive Foundation*. 2nd Edn. New Jersey: Prentice Hall.
- Kennedy, J. and W.M. Spears, 1998. Matching algorithms to problems: An experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator. In *IEEE International Conference on Evolutionary Computation*, pp: 78-83.
- Mitchell, G.G., D. O'Donoghue and A. Trenaman, 2000. A new operator for efficient evolutionary solutions to the traveling salesman problem. *Applied Informatics 2000*, Innsbruck, Austria, pp: 771-774.
- Mitchell, G.G., D. O'Donoghue, D. Barnes and M. McCarville, 2003. GeneRepair-A repair operator for genetic algorithms. *The Genetic and Evolutionary Computation Conference (GECCO)*, Chicago IL, pp: 45-49.
- Ridao, M.A., J. Riquelme, E.F. Camacho and M. Toro, 2001. An evolutionary and local search algorithm for motion planning of two manipulators. *J. Robotic Syst.* 18: 463-476.
- Syrjakow, M. and H. Szczerbicka, 1994. Optimization of simulation models with REMO. In *Proceedings of Conference on Modeling Simulation*, pp: 274-281.
- Yang, R., 1997. Solving Large TSP with Small Populations. In *Proceedings of GALESLA97, Genetic Algorithms in Engineering System: Innovations and Applications*, pp: 247-251.