Intern. J. Computer Math., Vol. 73, pp. 139–155
Reprints available directly from the publisher
Photocopying permitted by license only

FINE-GRAINED PARALLEL GENETIC ALGORITHM: A GLOBAL CONVERGENCE CRITERION

A. MUHAMMAD[a], A. BARGIELA[b,*] and G. KING[a]

[a] *Southampton Institute, Southampton, UK;*
[b] *Department of Computing, The Nottingham Trent University, Burton Street, Nottingham NG1 4BU, UK*

(Received 10 February 1999)

This paper presents a fine-grained parallel genetic algorithm with mutation rate as a control parameter. The function of the mutation rate is similar to the temperature parameter in the simulated annealing [3, 8, 10]. The motivation behind this research is to develop a global convergence theory for the fine-grained parallel genetic algorithms based on the simulated annealing model. There is a mathematical difficulty associated with the genetic algorithms as they do not strictly come under the definition of an algorithm. Algorithms normally have a starting point and a defined point of termination which genetic algorithms lack. The parallel genetic algorithm presented here is a stochastic process based on Markov chain [2] model. It has been proven that fine-grained parallel genetic algorithm is an ergodic Markov chain and that it converges to the stationary distribution. The theoretical result has been applied to in the context of optimisation of a deceptive function of 4-th order.

Keywords: Stochastic optimisation; parallel model; Markov chain; global convergence

C. R. Categories: I2.8, G1.6, G.3, I6.4, C1.2

1. INTRODUCTION

Parallel genetic algorithms are becoming more popular among researchers, due to the increased speed and efficiency [6]. There are two most popular parallel models of genetic algorithms. These are, distributed or coarse-grained model [9, 13], and massively parallel or fine-grained model [4, 7, 14]. The

*Corresponding author. e-mail: andre@doc.ntu.ac.uk

coarse-grained (distributed) parallel genetic algorithms assume the division of a large population into several subpopulations which are processed concurrently on different processing nodes. Each processor runs a simple genetic algorithm executing selection, crossover, and mutation on the corresponding subpopulation. There are two basic versions of the distributed parallel genetic algorithms, (a) mutually exclusive algorithms, which assume no communication between the processors, (b) non mutually exclusive, which permit exchange of the best individuals between processing nodes after few generations. There could be many versions of such algorithms depending on the subpopulations size, rate of the exchange of individuals and the number of individuals exchanged. However, the basic idea behind these algorithms is not different from that of the simple genetic algorithm, since each processor is running a simple genetic algorithm with only infrequent communications between processors. Consequently the convergence properties of the distributed and the simple genetic algorithms are comparable [1, 11, 12].

The fine-grained (massively parallel) genetic algorithms assume the placement of only one member of the population on each processing node (Fig. 1). Any individual can only mate with individuals located on the neighbouring processing nodes. The selection is a local phenomenon rather than a global one in contrast to simple and distributed genetic algorithms. Depending on the topology of the network of the processors there could be several versions of massively parallel genetic algorithms. In this study a simple toroidal mesh of processors has been used. Each node in this mesh is connected to four immediate neighbours. The advantage of the toroidal mesh topology is the reduction in the communication radius. Simple genetic algorithms normally suffer from the implementation problem as the

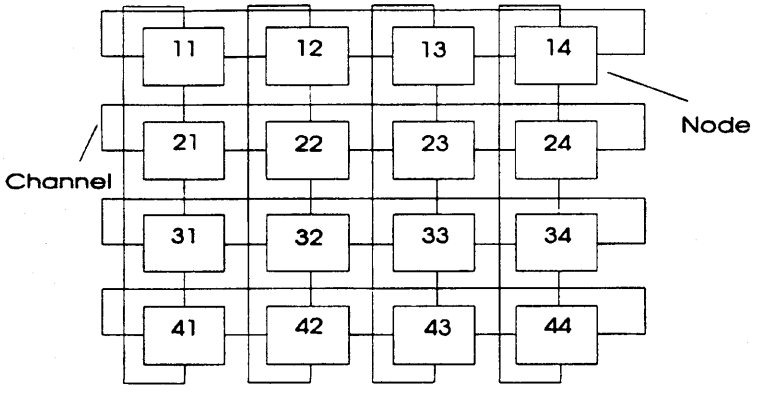

FIGURE 1 16-node toroidal mesh of processors.

architecture of the algorithm does not readily map onto the available hardware, whereas fine-grained parallel model can be implemented on any cellular arrays hardware.

The pseudo code for the fine-grained parallel genetic algorithm is given in the following:

```
for each node do in parallel
        generate an individual randomly
end parallel do
while not stop_criterion_satisfied do
      for each node do in parallel
            evaluate the fitness of the individual
            get the fitness values of four neighbouring
            individuals
            find out the optimum fitness value
            get the neighbouring individual
            corresponding to optimum fitness
            uniform crossover with the local
            individual according to the crossover rate
            mutate the individual according to the
            mutation rate
      end parallel do
      test the stopping criteria
end while
```

The next section provides the foundations of finite discrete-time Markov chains, with particular reference to genetic algorithms. In the subsequent sections the stochastic equations for local selection, uniform crossover and mutation operator have been developed. These stochastic equations have been summed up in a finite discrete-time Markov chain based model. This model has been used to study the asymptotic convergence properties of the fine-grained genetic algorithm. Finally the results of the theoretical study have been verified using computer simulations.

2. FINITE DISCRETE-TIME MARKOV CHAINS

In the algorithm presented above a population at one generation depends only on the previous generation and not the generation before, therefore the fine-grained parallel genetic algorithm is a stochastic process like the sequential counterpart. Likewise, state space of the algorithm is dependent

on the representation of the search parameters. Let the population be represented as binary strings of length L then the set S of all possible individuals have 2^L members, and the individuals can be indexed with i such that $0 \leq i < 2^L$.

The distribution of the population at each generation of the algorithm could be defined as vector,

$$\bar{q}_k = \{q_k(0)q_k(1)\cdots q_k(2^L-1)\}^T \tag{1}$$

where k is the number of the generation and $q_k(i)$ is the multiplicity of the individual $i \in S$ at the generation k. The distribution vector defined in Eq. (1) does not take into account the spatial distribution of the population on the toroidal mesh of the processors. Such a distribution vector could be in fact representing several spatial population distributions. The set, S', contains all such distributions form the search space or the state space of the genetic algorithm. The size of the set S' is given as

$$N' = \frac{(2^L-1)!}{(2^L-1-N)!} \tag{2}$$

where N is the population size or the number of nodes in the mesh of processors.

During a genetic search the distribution of the population at generation $i+1$, $\bar{q}_{i+1}$, depends only on the distribution of the previous generation, $\bar{q}_i$. This implies that the fine-grained parallel genetic algorithm as a stochastic process is a Markov chain (App., Def. 1). There are two approaches to the analysis of the Markov chain process. In the first method a transition matrix is obtained. The size of this matrix corresponds to the size of the state vector. In case of a genetic algorithm of individuals of length 10, the size of the transition matrix will be 1024×1024 which implies a significant computational load. The second method, adopted here, is to derive a general formula for a transition probability and to use it in the convergence analysis. The transition rule of a genetic algorithm is composed of three basic transition rules, associated with the stochastic genetic operators: selection, crossover and mutation respectively. These transition rules are aggregated into a transition rule for the fine-grained parallel genetic algorithm. It has been observed, from empirical studies of genetic algorithms, that given an appropriate mutation process the genetic search converges to the best region of the search space regardless of the initial (random in general) population.

3. THE STOCHASTIC TRANSITION RULE DUE TO SELECTION

There are number of strategies available for the selection of individuals in sequential algorithms. However, they all depend only on the probability distribution. The selection mechanism in the massively parallel model is entirely different from the sequential one. In particular, the topology of the computing nodes influences the selection of individuals in the subsequent generations.

There is a need for a functional representation of the topology of the population in order to explain the selection method. For a toroidal mesh topology we can write,

$$\{\alpha, \beta, \gamma, \delta\} = nbhd(i,j) \tag{3}$$

The function *nbhd* takes the location of the node as an argument and returns the individuals in the four immediate neighbouring nodes.

$$m = \text{optimum}\{\alpha, \beta, \gamma, \delta\} \tag{4}$$

The function optimum, as the name suggests, returns the fittest of the four arguments.

$$\begin{aligned} \sigma(i,(j,k)) = 1,\ f(i) &= \text{optimum}(nbhd(j,k)) \\ &= 0,\ \text{otherwise} \end{aligned} \tag{5}$$

The function σ takes two arguments i and (j,k), where i is the index of the individual and (j,k) is the location of the node, and it returns 1 if fitness value of individual i is the maximum in the neighbourhood of (j,k) or it returns 0 otherwise.

The probability of selecting an individual $i \in S$ from a population given the distribution of the current generation is

$$\frac{\sum_{m,n} \sigma(i,(m,n))}{N} \tag{6}$$

Where N is the population size, subscripts m and n represent the physical locations of the nodes and the summations goes through all nodes. It is obvious from the above expression that the probability of selection ranges from 0 to 1, inclusive. It is worth noting that selection rule is not given as an explicit function of the individual's fitness, this posses difficulty in simulating the model.

4. THE STOCHASTIC TRANSITION RULE FOR CROSSOVER

In the fine-grained parallel genetic algorithm the rate of crossover has been fixed to 1, so two selected individuals are always 'crossovered'. Uniform crossover is the generalisation of the n-point crossover. In order to cover a class of crossover methods a uniform crossover has been used in this model. The crossover of two individuals is controlled by a random binary string called mask. The mechanism the crossover could be explained as multiplexing two binary numbers using a control mask.

example:

$$s_1 = 01010101$$
$$s_2 = 11110000$$
$$m = 10101011$$
$$c = 01010001$$

A1 at position i of mask m selects the i-th bit from the parent s_i and 0 select the i-th bit from the parent s_2. Since, the length of the mask is the same as the length of the individuals in the population, L, the number of all possible masks is 2^L. Uniform crossover is ineffective on the similar bits of the mating individuals.

Let's define a Hamming distance function, $H(i, j)$. If i and j are two binary strings then the Hamming distance between these two strings is defined as

$$H(i,j) = |i \; XOR \; j|$$

Where *XOR* is the bitwise *exclusive-OR* function and the modulus | | is the number of 1's in the string. The probability of obtaining individual k as a result of uniform crossover between individual i and j could be defined as,

$$\frac{1}{2^{H(i,j)}}$$

The probability of obtaining an individual $i \in S$, as a result of the selection of two parents and then their uniform crossover, from the current generation given the distribution, is

$$\sum_{j,k} \left(\frac{\sum_{m,n} \sigma(j,(m,n))}{N} \right) \left(\frac{\sum_{m,n} \sigma(k,(m,n))}{N} \right) \frac{1}{2^{H(j,k)}} \qquad (7)$$

5. THE STOCHASTIC TRANSITION RULE FOR MUTATION

The mutation operator used in this method is the same as in any sequential algorithm for binary population. The mutation rate is defined as p_m and unlike the crossover rate it is variable. Search space of the binary strings could be visualised as a hypercube, which is a metric space. Again the Hamming distance, $H(i,j)$, is defined as a metric function of this search space.

The probability of obtaining individual i from individual j or *vice versa*, just by using mutation operator is $p_m^{H(i,j)}(1-p_m)^{L-H(i,j)}$, the first factor is the probability of differing bits to flip and the second factor is the probability of the survival from mutation of the same bits of the two individuals.

The probability of obtaining an individual $i \in S$, as a result of selection and then mutation from the current generation given the probability distribution is

$$\sum_{j \in S} \left(\frac{\sum_{m,n} \sigma(j,(m,n))}{N} \right) p_m^{H(i,j)} (1-p_m)^{L-H(i,j)} \tag{8}$$

The mutation rate could be fixed to a value or it could change dynamically with generations.

6. THE TRANSITION RULE FOR THE FINE-GRAINED PARALLEL GENETIC ALGORITHM

In this section the combined effect of the three genetic operators selection, crossover, and mutation are added into one Markov transition rule to give a complete model. The probability of obtaining an individual $i \in S$, as a result of fine-grained parallel genetic algorithm based search from the current generation given the distribution, is

$$\frac{1}{N^2} \sum_{j \in S} p_m^{H(i,j)} (1-p_m)^{L-H(i,j)} \sum_{k,l \in S} \left(\sum_{m,n} \sigma(k,(m,n)) \right) \left(\sum_{m,n} \sigma(l,(m,n)) \right) \frac{1}{2^{H(k,l)}} \tag{9}$$

The above probability could be easily gathered in a transition matrix $P = (p_{ij})$ to obtain the distribution of next generation using the multinomial probability distribution of next generation using the multinomial probability distribution. The resultant Markov transition matrix for fine-grained

parallel genetic algorithm is given as follows,

$$\bar{P} = \frac{N!}{\prod_{i \in S} q_k(i)!} \prod_{i \in S} \left\{ \frac{1}{N^2} \sum_{j \in S} p_m^{H(i,j)} (1 - p_m)^{L - H(i,j)} \right.$$
$$\left. \sum_{k,l \in S} \left[\sum_{m,n} \sigma(k, (m,n)) \right] \left[\sum_{m,n} \sigma(l, (m,n)) \right] \frac{1}{2^{H(k,l)}} \right\}^{q_k(i)} \quad (10)$$

Once the transition matrix for the Markov chain representing the genetic algorithm has been obtained, the convergence properties of this stochastic optimisation method could be studied.

7. CONVERGENCE

It is evident from the probability transition, given by Eq. (10), that the mutation rate p_m is the control parameter for the corresponding Markov chain. If the mutation rate p_m is fixed for all generations then the algorithm and the corresponding Markov chain is called time-homogeneous (App., Def. 3). The homogeneous algorithm could be defined as a sequence of homogeneous Markov chains, such that each Markov chain is generated at a fixed value of p_m and it is decreased between two subsequent Markov chains. The non-homogeneous algorithm could be defined as a single non-homogeneous Markov chain in which the value of p_m is changed at each step of transition. The value of p_m (the analogue of temperature in a simulated annealing) could be used to drive the genetic algorithm towards the global optimal solution. In this study the convergence of the homogeneous algorithm will be presented and as such it is essential to prove to existence of stationary distribution (under certain conditions). The Markov chain is required to be irreducible (App., Def. 5) and aperiodic (App., Def. 6) in order to be convergent to a unique stationary distribution. The first condition requires that it should be able to move from one state to another in finite time and the second condition requires that it should always be able to move from one state to another. A Markov chain is irreducible if it is not absorbing, *i.e.*, the elements of the transition matrix are less than 1. In order to be able to move from one state to another the transition probability should always be greater than zero; this implies aperiodicity.

Let

$$\lambda = \frac{p_m}{1 - p_m}$$

then

$$p_m = \frac{\lambda}{1+\lambda}$$

The transition rule for mutation, Eq. (8), can be rewritten as

$$p_3 = \frac{1}{(1+\lambda)^L} \sum_{j \in S} \left(\frac{\sum_{m,n} \sigma(j,(m,n))}{N} \right) \lambda^{H(i,j)}$$

and similarly, Eq. (9) can be rewritten as

$$p = \frac{1}{(1+\lambda)^L} \sum_{j \in S} \lambda^{H(i,j)} p_2$$

since $0 < p_m < 1$ and $0 \leq H(i,j) \leq L$ therefore $0 < \lambda < \infty$, and

$$\lambda^L \sum p_2 \leq \sum \lambda^{H(i,j)} p_2 \leq \sum p_2$$

Therefore,

$$\left(\frac{\lambda}{1+\lambda} \right)^L \leq p \leq \left(\frac{1}{1+\lambda} \right)^L$$

and consequently,

$$\frac{N!}{\prod_{i \in S} q_k(i)!} \left(\frac{\lambda}{1+\lambda} \right)^{NL} \leq \bar{P} \leq \frac{N!}{\prod_{i \in S} q_k(i)!} \left(\frac{1}{1+\lambda} \right)^{LN}$$

The irreducibility and aperiodicity follows as a result of the above inequality. From the theorem (App., Theorem) it could be concluded that since the Markov chain representing parallel genetic algorithm is time-homogeneous and aperiodic, a unique stationary distribution exists. The finegrained parallel genetic algorithm is an ergodic Markov chain. That is, there exists a unique limit distribution for the states of the chain which has a nonzero probability of being in any state at any time regardless of the initial distribution. The stationary distribution,

$$\bar{q}_\infty = \lim_{k \to \infty} \bar{P}^k \bar{q}$$

and

$$\bar{p}\bar{q}_\infty = \lim_{k\to\infty} \bar{P}^{k+1}\bar{q} = \bar{q}_\infty$$

It proves that the stationary distribution is an eigenvector of transition matrix P with eigenvalue $\lambda = 1$.

The convergence to a stationary distribution is in fact a negation of the convergence to the global optimal solution because even at limiting conditions $k \to \infty$ $p_{ij} > 0$, the algorithm has the positive probability to move away from the optimal solution. The reason behind this is the positive value of mutation rate p_m which implies that, even having reached the global optimal state, the Markov chain will move to some other state. In order to avoid this situation an elitist strategy is normally employed, in which the best solution from each generation is preserved. But in the context of a massively parallel model presented in this paper it is not possible to use elitist method so, instead, a greedy approach is proposed in which individuals are only replaced when they are better than their parents.

The transition probability p_{ij} of obtaining an individual $i \in S$, from individual $j \in S$ as a result of executing the fine-grained parallel genetic algorithm with greedy genetic search is

$$p_{ij} = \begin{cases} p & \text{if } f(x_i) \geq f(x_j) \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

p is the transition probability as given in Eq. (9)

This modified approach ensures that the transition matrix P (Eq. (10)) is in a reducible form. The transition matrix now represents an absorbing Markov chain. Now the algorithm is not only convergent to a stationary distribution but also to a global optimal solution.

8. FURTHER EVIDENCE

The theoretical result that the fine-grained parallel genetic algorithm with greedy search is a convergent Markov chain and that a unique solution exists, has been verified through computer simulations. It has also been shown that it is not possible to get the stationary distribution just be repeatedly multiplying the initial distribution with the transition matrix under the limiting condition because, the selection of individuals is not only based on the fitness value but also on the physical distribution of the

individuals on the network of processors. The algorithm proposed here is implemented on a toroidal mesh of 16 processors connected as shown in the Figure 1, and the results obtained at each generation were used to get the population distribution. The parallel system is simulated on a transputer board using parallel C which provides an ideal platform for a Communicating Sequential Processes (CSP) model of computation.

Example The function chosen for this particular example is the deceptive function of order 4 (Fig. 2). Deceptive functions [13] are basically walsh polynomials and they tend to mislead the genetic search. The length of binary strings L is 5, population size 16 and mutation rate is fixed at 0.0125. The mutation rate corresponds to one mutation per generation. The initial distribution of the population is give in Figure 3. The distribution after 19 and 100 generations are presented in Figure 4, and it can be seen that the algorithm has reached a steady state as there is no change in population distribution after a threshold point.

The above is the result of just one homogeneous Markov chain. It could be observed from the results that at the beginning of the simulations there is no obvious pattern of behaviour to the distribution, but in subsequent

FIGURE 2 Deceptive function of order 4.

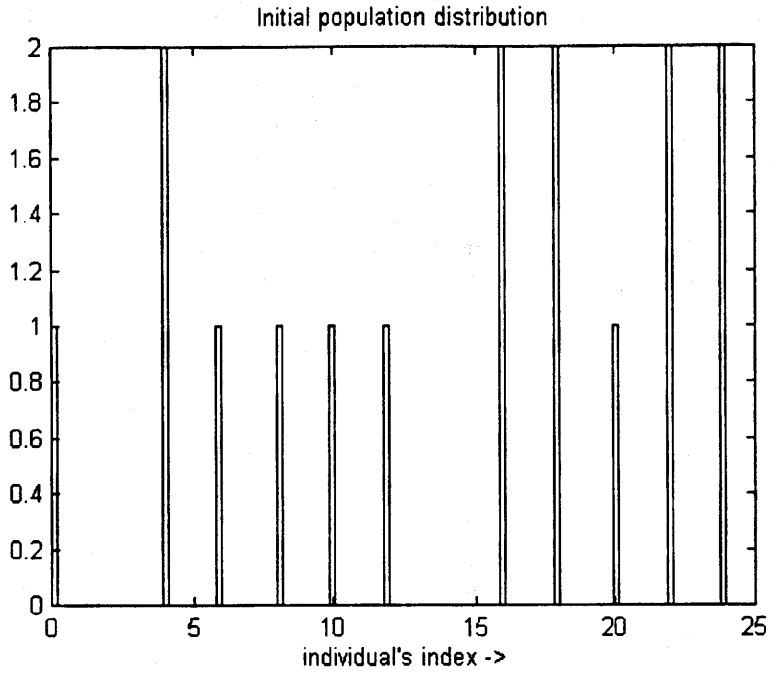

FIGURE 3 Initial distribution of the population.

generations individuals with higher fitness get more allocation in the next generation and after 19 generations all of the individuals display highest fitness value. It is evident that the greedy approach provides a directional force in the genetic search.

The convergent behaviour of a stochastic processes could also be studied using time series methods. The best solutions produced at each generation of the fine-grained parallel genetic algorithms could be considered elements in a time series.

DEFINITION A time series is an indexed collection $\{x(t), t \in T\}$ of random variables.

where

't', in the context of genetic algorithms, represents a generation index.

However it needs to be borne in mind that time series analysis applies to random processes which are defined in the form of closed mathematical or statistical equations. The difficulty with the random processes involved in genetic algorithms is that it is not described by the closed form equations

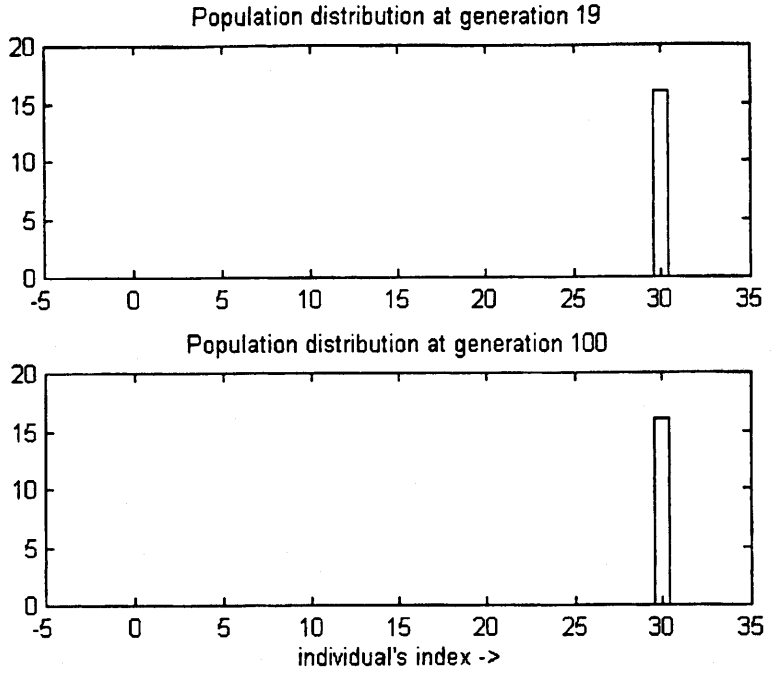

FIGURE 4 Population distribution after 19 and 100 generations.

which can be usable for the purpose of classical time series analysis. Consequently, the convergence of a time series is defined here as follows:

DEFINITION Let X_n be a sequence of random variables. Then,

(a) X_n converges to the constant l in probability if $\forall \varepsilon > 0$,

$$\lim_{n\to\infty} P(|X_n - l| > \varepsilon) = 0.$$

(b) X_n converges to X in mean square if

$$\lim_{n\to\infty} E(|X_n - X|^2) = 0, \quad \text{where } X \text{ is another random variable.}$$

A time series is also said to be weakly convergent in the mean if the average of a single realisation approaches the average at a single time t of the ensemble of all possible realisations. It can be seen from the simulation results, illustrated in Figure 5, that the best and average solutions all converge to a single value after certain number of iterations which supports our

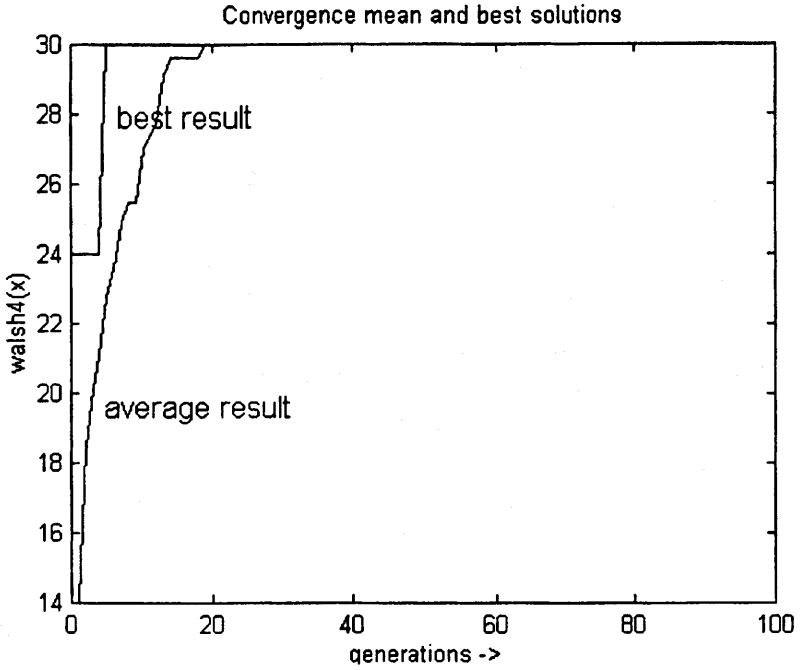

FIGURE 5 Convergence of the best and average genetic solutions.

conjucture that the time series generated by genetic algorithms are convergent in mean and consequently convergent to a unique solution.

9. CONCLUSION

The transition matrix for the Markov chain suggests that the chain is irreducible and aperiodic. These two conditions establish the fact that this chain is ergodic and a unique stationary distribution of the population exists. These properties provide enough information about the convergence of the algorithm, although they do not guarantee the convergence to the global optimal solution. However, the algorithm provides a basis for the development of variants which will have global convergence. The greedy approach in parallel model is similar to the elitist approach in sequential model. A greedy method of genetic search is convergent to global optimal solution. The speed of convergence of genetic algorithms makes them promising vehicles for the development of efficient optimisation methods. This will be reported in a separate publication.

Acknowledgements

The preliminary version of this paper has been presented at the 1st World Congress on Computer Simulation [5]. The authors would like to acknowledge the helpful comments of the anonymous referees.

References

[1] Davis, T. E. (1991). A simulated annealing like convergence theory for the simple genetic algorithm, *International Conference on Genetic Algorithms and Applications.*
[2] Kemeny, J. G. (1960). Finite Markov Chains, Springer-Verlag.
[3] Lundy, M. (1986). Convergence of an annealing algorithm, Mathematical Programming, **34**, 111–124.
[4] Manderick, B. (1989). Fine-grained parallel genetic algorithms, *International Conference on Genetic Algorithms and Applications.*
[5] Muhammad, A., Bargiela, A. and King, G. (1997). Fine-grained genetic algorithm: A stochastic optimisation method, *1st World Congress on Computer Simulation, Singapore.*
[6] Muhlenbein, H. (1989). Parallel genetic algorithms, population genetics and combinatorial optimisation, *International Conference on Genetic Algorithms and Applications.*
[7] Muhlenbein, H. (1991). The parallel genetic algorithm as function optimiser, *International Conference on Genetic Algorithms and Applications.*
[8] Otten, R. H. J. M. (1989). The annealing algorithm, Kluwer Academic Publisher.
[9] Pettey, C. C. (1989). A theoretical investigation of a parallel genetic algorithm, *International Conference on Genetic Algorithms and Applications.*
[10] Romeo, F. (1985). Probabilistic hill climbing algorithms: properties and applications, *Chapel Hill conference on VLSI.*
[11] Rudolph, G. (1994). Convergence analysis of canonical genetic algorithms, *IEEE Transactions on Neural Networks*, **5**(1), 78–93.
[12] Suzuki, J. (1993). A Markov chain analysis on a genetic algorithm, *International Conference on Genetic Algorithms and Applications.*
[13] Tanese, R. (1989). Distributed genetic algorithms, *International Conference on Genetic Algorithms and Applications.*
[14] Tomassini, M. (1993). The parallel genetic cellular automata: Application to global optimization, *International Conference on Genetic Algorithms and Applications.*
[15] Linkens, D. and Nyongesa, H. (1995). Genetic algorithms for fuzzy control (Parts 1 and 2), *IEE Proc., Control Theory Applic.*, **142**(3), 161–185.
[16] Pham, D. and Karaboga, D. (1997). Genetic algorithms with variable mutation rates: application to fuzzy logic controller design, *Systems and Control Engineering Journal (Proc. Inst. Mech. Eng.)*, **211**(1), 157–167.

APPENDIX

DEFINITION 1 (Markov chain) Let $X = \{X_i : i \in S\}$ be a discrete time stochastic process with finite state space S. If

$$\Pr\{X_{k+1} = j | X_o = i_0 \ldots X_k = i\} = \Pr\{X_{k+1} = j | X_k = i\}$$

then X is a Markov chain.

DEFINITION 2 (Stochastic Matrix) A square matrix is called stochastic matrix if all rows are composed of probability vector or probability distribution. The following stochastic matrix

$$\bar{P}_k = [\Pr\{X_{k+1} = j | X_k = i\}]$$

is the single step transition matrix of the Markov chain X at step k. If the probability distribution at step k is given as $\bar{q}_k$ the probability distribution at the next step can be determined as follows

$$\bar{q}_{k+1} = \bar{P}_k \bar{q}_k$$

alternatively if the initial distribution is $\bar{q}_o$ then distribution at any step could be determined as follows

$$\bar{q}_k = \left(\prod_{j=0}^{k-1} \bar{P}_j \right) \bar{q}_o$$

DEFINITION 3 (Homogeneous Markov chains)Let $\bar{P}_k$ be the state transition matrix for Markov chain X at step k. Then, X is called time-homogeneous if $\bar{P}_k (= \bar{P})$ is independent of time step k. As a result of this if the initial probability distribution $\bar{q}_o$ is known then probability distribution of the Markov chain X at any step k could be determined as follows

$$\bar{q}_k = \bar{P}^k \bar{q}_o$$

DEFINITION 4 (Stationary distribution) It is defined as the vector $\bar{q} = \{\bar{q}_i\}$ such that

$$q_i = \lim_{k \to \infty} \Pr\{X(k) = i | X(0) = j\}$$

for any arbitrary j. It means if the stationary distribution exists then it is independent of the starting point.

DEFINITION 5 (Irreducibility) A Markov chain is irreducible if and only if for any two states i and j there is a positive probability of reaching j from i in a finite number of transitions, *i.e.*,

$$\forall\, i, j \quad \exists n : 1 \leq n \leq \infty \quad \text{and} \quad (P^n)_{ij} \geq 0$$

DEFINITION 6 (Aperiodicity) A Markov chain is called aperiodic if and only for all $i \in S$, the greatest common divisor of all integers $n \geq 1$, such that

$$(P^n)_{ii} > 0$$

is equal to 1.

THEOREM *The unique stationary distribution of a finite homogeneous Markov chain exists if the Markov chain is irreducible and aperiodic.*