The Strategy of Improving Convergence of Genetic Algorithm

Jiang Jing*, Meng Lidong
School of Electrical and Electronic Engineering, Shandong University of Technology
Zibo city,Shandong Province,China
*corresponding author, e-mail: anyangjj@163.com

Abstract

Premature convergence is the main obstacle to the application of genetic algorithm. The study on convergence of GA is always one of the most important theoretical issues. Via analyzing the convergence rate of GA, the average computational complexity can be implied and the optimization efficiency of GA can be judged. This paper proposed an approach to calculating the first expected hitting time and analyzed the bounds of the first hitting time of concrete GA using the proposed approach. And this paper proposed a strategy which included transformation of fitness function, self-adaptive crossover and mutation probability and close relative breeding avoidance method in order to overcome premature convergence.

Keywords: *genetic algorithm; convergence rate; self-adaptive probability; fitness transformation*

1. Introduction

GA (genetic algorithm) was proposed by Holland in 1975 who was inspired by the biological evolution. Genetic algorithm is a naturally motivated general purpose optimization method which requires little knowledge of the problem to be optimized other than a fitness function. GA initially begins search with random points in the solution space then incrementally generate new points by applying copy, crossover and mutation operators. Which points are pursued is controlled by a probablistic decision procedure that guides the method into optimal regions of the solution space. GA is a highly parallel, random and self-adaptive algorithm which has many merits over traditional methods such as global optimization.But in practice, GA is often criticized for the lack of a solid theoretical foundation. Actually, such a theoretical foundation is very desired in order to gain deep understanding of the strength and weakness of GA and thus develop better algorithms. GA usually has the drawbacks such as premature convergence and slow convergent speed. Premature convergence means although GA has not reached a global or satisfactory optimum, it can't produce a better offspring individual which outperforms its parent individuals. When premature convergence occurs, it is difficult for GA to get rid of a local optimum and reach a global optimum[1,2]. Premature convergence is the main obstacle to a genetic algorithm's practical application. In order to overcome genetic algorithm's premature convergence, we should first have a good understanding of convergence and convergence rate of GA.

The convergence analysis theories only prove the possibility of finding a global optimal solution finally and can not be used to evaluate the practical algorithm performance.The computational time complexity of finding a global optimal solution is judged through analyzing a genetic algorithm's convergence rate. It is very important to study the convergence and convergence rate of GA. Firstly convergence of GA must be guaranteed and then convergence rate of GA should be analyzed to evaluate the computational complexity of finding a global optimal solution, so that the optimization efficiency of the algorithm can be judged and used to improve the algorithm correctly.Many scholars and experts at home and abroad have made a thorough and extensive study on convergence and convergence rate of GA. The concept of convergence rate of GA is proposed to study how rapidly GA converged to a global optimal solution, or it can be viewed as the estimation of the expected hitting time. The definition of EFHT(Expected First Hitting Time) was proposed by paper[3] and paper[3] made an explanation for GA hard and GA easy problems. Based on an absorbing Markov chain, paper[4] proposed analytical theories to estimate convergence rate of ant colony algorithm, and the

Received September 28, 2012; Revised November 4, 2012; Accepted November 12, 2012

author made concrete studies on concrete problems. Paper[5] proved convergence of GA in noisy enviroment and proposed an approach to estimating the first hitting time of GA in noisy enviroment. These scholars have made analysis on convergence and convergence rate from different viewpoints.

This paper analyzes the convergence rate of GA based on an absorbing Markov chain. Finally, this paper proposes a strategy which tries to overcome premature convergence from three aspects. The proposed strategy includes self-adaptive crossover and mutation probability, close relative breeding avoidance method based on Hamming distance and linear scale transformation of fitness function.

2. Calculating and Estimating the First Expected Hitting Time of GA

Since every genetic operation relies only on the present state and is not related with the previous one, a genetic algorithm can be described by a Markov chain model and its convergence can be studied through Markov chain theories[6,7].

Given a Markov chain {P(n),n≥0} and a optimum space Y*, if {P(n),n≥0} satisfies P(P(n+1)∩Y*=Φ|P(n)∩Y*≠Φ)=0, then {P(n),n≥0} is called an absorbing Markov chain. It can be easily drawn that the generation sequence {P(n),n≥0} of GA which adopts preserving optimum strategy is an absorbing Markov chain, the optimum is the absorbing state.

Given an absorbing Markov process $\{\xi(t)\}_{t=0}^{+\infty}(\forall\xi(t)\in Y)$ and an optimal state space $Y^*\subset Y$; if μ which is a random variable satisfies:when t= μ , $\xi(t)\in Y^*$ when 0≤t< μ $\xi(t)\notin Y^*$,then the expected value Eμ is called the first expected hitting time.

The expected convergence time Eμ describes GA reaches an optimum the first time with probability one[8]. The smaller is its value, the more rapidly is GA's convergence rate and the higher is its efficiency. So the first expected hitting time can be viewed as an index of GA's convergence rate. The first expected hitting time of GA can be obtained by calculating the value of Eμ. The following theorem gives a most direct approach to calculating the value of Eμ.

Theorem 1 Given an absorbing Markov process $\{\xi(t)\}_{t=0}^{+\infty}(\forall\xi(t)\in Y)$ of GA and its optimal state space $Y^*\subset Y$, $\lambda(t)=P\{\xi(t)\in Y^*\}$ and $\lim_{t\to+\infty}\lambda(t)=1$, the first expected hitting time is $E\mu=\sum_{i=0}^{+\infty}(1-\lambda(i))$

Proof.

Because $\xi(t)$ is a convergent Markov process, sup pose μ is the first hitting time,

For $\forall t=1,2,\cdots$ we get

$$\lambda(t)=P\{\xi(t)\in Y^*\}=P\{\mu\le t\}$$
$$\Rightarrow\lambda(t)-\lambda(t-1)=P\{\mu\le t\}-P\{\mu\le t-1\}$$
$$\Rightarrow P\{\mu=t\}=\lambda(t)-\lambda(t-1)$$
$$Then\quad E\mu=0\cdot P\{\mu=0\}+\sum_{t=1}^{+\infty}t\cdot P\{\mu=t\}$$
$$=\sum_{t=1}^{+\infty}t\cdot(\lambda(t)-\lambda(t-1))=\sum_{i=1}^{+\infty}\sum_{t=i}^{+\infty}(\lambda(t)-\lambda(t-1))$$
$$=\sum_{i=0}^{+\infty}\left[\lim_{t\to+\infty}\lambda(t)-\lambda(i)\right]=\sum_{i=0}^{+\infty}(1-\lambda(i))$$
$$E\mu=\sum_{i=0}^{+\infty}(1-\lambda(i))\, s\tan ds.$$

If for $\forall t=1,2,\cdots,$ *the value of* $\lambda(t)$ is known, then we can calculate exactly the expected convergence time of GA and thus estimate its convergence rate. But in practice, for

$\forall t = 1,2,\cdots,$ *the value of* $\lambda(\mathrm{t})$ is difficult to be known exactly, so it is common to estimate the bounds of the expected convergence time.

Theorem 2 Given two discrete random nonnegative integers u and v,let Du(·) and Dv(·) be distribution functions of u and v, when $D_u(t) \ge D_v(t)$ stands for $\forall \mathrm{t} = 0,1,2,\cdots$, the expected value of u and v satisfies Eu≤Ev.

Proof.

$$\textit{Because } \mathrm{D_u(t) = P\{u \le t\}} \textit{ and } \mathrm{D_v(t) = P\{v \le t\}(\forall t = 0,1,2,\cdots)},$$

$$\textit{so we get } \mathrm{Eu} = 0 \cdot \mathrm{D_u}(0) + \sum_{\mathrm{t}=1}^{+\infty} \mathrm{t}[\mathrm{D_u(t) - D_u(t-1)}] = \sum_{i=0}^{+\infty}[1 - D_u(i)]$$

$$Eu - Ev = \sum_{i=0}^{+\infty}[1 - D_u(i)] - \sum_{i=0}^{+\infty}[1 - D_v(i)] = \sum_{i=0}^{+\infty}[D_v(i) - D_u(i)] \le 0 \Rightarrow Eu \le Ev$$

Theorem 3 Given an absorbing Markov process $\{\xi(t)\}_{t=0}^{+\infty}(\forall \xi(t) = (X(t), T(t)) \in Y)$ of GA and its optimal state space $Y^* \subset Y$,if $\lambda(t) = P\{\xi(t) \in Y^*\}$satisfies $0 \leqslant \mathrm{D_l(t)} \le \lambda(\mathrm{t}) \le \mathrm{D_h(t)} \le 1$ $(\forall t = 0,1,2,\cdots)$ and $\lim\limits_{t \to +\infty} \lambda(t) = 1$,then $\sum\limits_{\mathrm{t}=0}^{+\infty}(1 - \mathrm{D_h(t)}) \le \mathrm{E\mu} \le \sum\limits_{\mathrm{t}=0}^{+\infty}(1 - \mathrm{D_l(t)})$

Proof.

Construct two discrete random nonnegative integer variables h and l,whose distribution functions are $\mathrm{D_h(t)}$ and $\mathrm{D_l(t)}$ respectively. Obviously, μ is also a discrete random integer variable, whose distribution function is $\lambda(t) = P\{\xi(t) \in Y^*\} = P\{\mu \le t\}$.

$$\textit{Because } 0 \le \mathrm{D}_l(t) \le \lambda(t) \le D_h(t) \le 1, \textit{ from the above analysis, we get:}$$

$$\mathrm{E}h \le E\mu \le El \Leftrightarrow \sum_{t=0}^{+\infty}(1 - D_h(t)) \le E\mu \le \sum_{t=0}^{+\infty}(1 - D_l(t))$$

The above formula tells that if the upper and lower bounds of λ(t) can be known, then the upper and lower bounds of the expected convergence time can be determined. The following contents give a practical approach to estimating the expected convergence time Eμ.

Theorom 4

Given an absorbing Markov process $\{\xi(t)\}_{t=0}^{+\infty}(\forall \xi(t) = (X(t), T(t)) \in Y)$ of GA and a target space Y^*, if $\lambda(t) = P\{\xi(t) \in Y^*\}$ satisfies $a(t) \le P\{\xi(t) \in Y^* \mid \xi(t-1) \notin Y^*\} \le b(t)$, then $\sum\limits_{t=0}^{+\infty}\left[(1 - \lambda(0))\prod\limits_{i=1}^{t}(1 - b(i))\right] \le E\mu \le \sum\limits_{t=0}^{+\infty}\left[(1 - \lambda(0))\prod\limits_{i=1}^{t}(1 - a(i))\right]$.

Proof.

$$\textit{Because } \lambda(\mathrm{t}) = [1 - \lambda(\mathrm{t-1})]\mathrm{P}\{\xi(\mathrm{t}) \in \mathrm{Y}^* \mid \xi(\mathrm{t-1}) \notin \mathrm{Y}^*\} + \lambda(\mathrm{t-1})\mathrm{P}\{\xi(\mathrm{t}) \in \mathrm{Y}^* \mid \xi(\mathrm{t-1}) \in \mathrm{Y}^*\}$$
$$(\forall t = 0,1,2,\cdots)$$

$$\textit{and } \mathrm{P}\{\xi(\mathrm{t}) \in \mathrm{Y}^* \mid \xi(\mathrm{t-1}) \in \mathrm{Y}^*\} = 1$$

$$\textit{thus we get } 1 - \lambda(\mathrm{t}) \le [1 - a(t)][1 - \lambda(\mathrm{t-1})] = [1 - \lambda(0)]\prod_{i=1}^{t}[1 - a(i)]$$

$$\mathrm{E}\mu = \sum_{i=0}^{+\infty}(1 - \lambda(i)) \le \sum_{t=0}^{+\infty}\left[(1 - \lambda(0))\prod_{i=1}^{t}(1 - a(i))\right]$$

Similarly, we get

$$\mathrm{E}\mu = \sum_{i=0}^{+\infty}(1 - \lambda(i)) \ge \sum_{t=0}^{+\infty}\left[(1 - \lambda(0))\prod_{i=1}^{t}(1 - \mathrm{b}(i))\right]$$

The above approach can be applied to practical analysis more easily, because $P\{\xi(t) \in Y^* \mid \xi(t-1) \notin Y^*\}$ describes the probability with which a nonoptimal state is transformed into an optimal one. The bounds of the probability can be estimated relatively easily. The estimated precision of the probability bounds has a direct influence on the estimated precision of Eμ.

Inference

Given an absorbing Markov process $\{\xi(t)\}_{t=0}^{+\infty}(\forall\xi(t)=(X(t),T(t))\in Y)$ and its optimal state space $Y^{*}\subset Y$ and $\lambda(t)=P\{\xi(t)\in Y^{*}\}$, if $a\le P\{\xi(t)\in Y^{*}\mid\xi(t-1)\notin Y^{*}\}\le b$, $(a,b>0)\ and\ \lim_{t\to+\infty}\lambda(t)=1$,then the expected convergence time $E\mu$ of GA satisfies: $b^{-1}[1-\lambda(0)]\le E\mu\le a^{-1}[1-\lambda(0)]$.

Given any two individuals x,y, supposing H(x,y) is Hamming distance between them, if every bit of x is flipped with probability P_m(P_m<0.5), then the probability with which x is transformed into y is:p(x→y)=$p_m^{H(x,y)}$×(1-p_m)$^{L-H(x,y)}$

Because p_m<1-p_m, so we get p(x→y)≥$p_m^{H(x,y)}$×$p_m^{L-H(x,y)}$= p_m^{L} ,Similarly : p(x→y)≤(1-p_m)^{L}

Thus we get : $\mathrm{p_m}^{\mathrm{L}}\le\mathrm{P}\{\xi(t)\in Y^{*}\mid\xi(t-1)\notin Y^{*}\}\le(1-\mathrm{p_m})^{\mathrm{L}}$

From the above analysis we get : $\frac{1}{(1-p_m)^{L}}[1-\lambda(0)]\le E\mu\le\frac{1}{p_m{}^{L}}[1-\lambda(0)]$

When the probability of mutation is close to 0.5, it can be easily drawn that the distance between the lower and upper bounds of the expected convergence time is very small.

3. A Method to Overcome Premature Convergence

3.1. Linear Scale Transformation of a Fitness Function

As a fitness function is the only criterion to judge performance of evolutionary individuals, it is important to select a correct fitness function for it affects a genetic algorithm's convergence rate and the ability to find a global optimum directly.

In the early stage of GA, a few individuals which have a big fitness value may occur in an evolutionary group. It is possible that these individuals may produce offspings in a big amount in a short time, so the diversity of the group decreases and GA finds a local optimum prematurely. So in the early stage, a good fitness function should lower the difference between strong and weak individuals in order to limit the amount of strong individuals and maintain diversity of the group. In the late stage of GA, difference between individuals is decreasing and it is possible that the evolutionary process lose competition and degrade to a random selection process. So a good fitness function should enlarge the difference between strong individuals and weak individuals in the late stage in order to increase competition between individuals.

Application results show that a fitness function affects the performance of GA greatly. A fitness function decides the evolutionary direction of a group. It needs to properly enlarge or decrease individuals' fitness values in different stages[9,10]. This fitness value enlargement or decreasement is called fitness scale transformation. The equation of fitness value linear scale transformation is shown as the following:

$$F^{'}=a\cdot F+b \tag{1}$$

In (1), F is an original fitness value, F' is a new fitness value after the scale transformation. Coefficients a and b affect the scale transformation directly, which need to satisfy the following requirements:

$$a=\frac{(C-1)\cdot F_{avg}}{F_{\max}-F_{avg}} \tag{2}$$

$$b=\frac{(F_{\max}-C\cdot F_{avg})\cdot F_{avg}}{F_{\max}-F_{avg}} \tag{3}$$

In (2) and (3), F_{avg} is the average fitness value of the evolutionary group, F_{max} is the biggest fitness value of the group. Eq.(2) and (3) show that coefficient C decides coefficients a and b value. In (2) and (3), coefficients a and b value should not be fixed, or else they can't play the role of decreasing fitness difference in the early stage and enlarging fitness difference in the late stage. It can be said that in the different stages of the algorithm, coefficient C value should be adjusted accordingly, which needs to satisfy the following requirements:

In the early stage of GA:

$$1 < C < \frac{F_{\max}}{F_{avg}} \tag{4}$$

In the late stage of GA:

$$\frac{F_{\max}}{F_{avg}} < C < \frac{F_{\max} - F_{\min}}{F_{avg} - F_{\min}} \tag{5}$$

3.2. Self-adaptive Crossover and Mutation Probability

Rapid convergence and a global optimum are a couple of contradictory requirements. In order to ensure rapid convergence, GA needs an evolutionary group to produce the best individual rapidly, thus inevitably decreasing diversity of the group and easily making premature convergence occur. In order to ensure a global optimum and get rid of a local optimum, the algorithm needs to maintain diversity of the group at the cost of convergent speed.

Self-adaptive crossover and mutation probability synthesize the two requirements of speed and a global optimum. When the fitness values of individuals are near to each other, it needs to increase crossover probability P_c and mutation probability P_m in order to increase diversity of the group and get rid of a local optimum. When the fitness values of individuals are scattered, decreasing P_c and P_m can avoid random roaming and increase convergent speed.

Supposing F_{max} is the biggest fitness value of an evolutionary group, F_{avg} is the average fitness value of the evolutionary group, f' is the bigger fitness value of two paired individuals between which crossover operation is to be done, f is the fitness value of an individual on which mutation operation is to be done. The difference between the biggest fitness value F_{max} and the average fitness value F_{avg} expresses the stability of the evolutionary group to some extent. The smaller is the difference between F_{max} and $F_{avg,}$ the smaller is the difference between fitness values of the individuals, and the bigger is the probability that the group reach premature convergence[11,12]. The bigger is the difference between F_{max} and F_{avg}, the bigger is the difference between the fitness values of individuals, and the more scattered are the individuals of the group distributed.

In order to overcome premature convergence, when the difference between F_{max} and F_{avg} is small, the algorithm will increase P_c and P_m; otherwise when the difference between F_{max} and F_{avg} is big, the algorithm will decrease P_c and P_m, that is P_c and P_m are inversely proportional to the difference between F_{max} and F_{avg}. And different individuals should be given different P_c and P_m, an individual which has a good fitness value should be protected, its P_c and P_m should be decreased. Otherwise if an individual has a lower fitness value , its P_c and P_m should be increased. And considering that the best individual in early stage is not certain to be a global optimum, whose crossover and mutation probability should not be zero in the early stage.

In view of the above considerations, P_c and P_m are calculated as the following equation:

$$P_c = \begin{cases} P_{c1} - \dfrac{(P_{c1} - P_{c2})(f' - F_{avg})}{F_{\max} - F_{avg}} & f' \geq F_{avg} \\ P_{c1} & f' < F_{avg} \end{cases} \tag{6}$$

$$P_m = \begin{cases} P_{m1} - \dfrac{(P_{m1} - P_{m2})(F_{\max} - f)}{F_{\max} - F_{avg}} & f \geq F_{avg} \\ P_{m1} & f < F_{avg} \end{cases} \tag{7}$$

In (6) and (7), P_{c1} is 0.9, P_{c2} is 0.6, P_{m1} is 0.1, P_{m2} is 0.001

3.3. Close Relative Breeding Avoidance Method

In practice, crossover probability P_c is commonly set between 0.4~0.95, and mutation probability P_m is often set between 0.001~0.01. Crossover operator plays a key role in the evolutionary process. It is an effective way to overcome premature convergence by using crossover operator to produce new individuals in order to increase the diversity of the group. If the group can't produce new individuals by crossover operator, then it can be said that the group loses its ability to become competitive.

Close relative crossover avoidance method based on Hamming distance can avoid crossover operation between similar individuals to some extent. Hamming distance can reflect the difference extent between two binary coded individuals clearly. The equation to calculate Hamming distance is:

$$H(X_i(t), X_j(t)) = \sum_{k=1}^{L} \left| x_{ik}(t) - x_{jk}(t) \right| \tag{8}$$

In (8), $X_i(t)$ and $X_j(t)$ are two individuals of the t generation group, L is the length of the binary coded individual's genes. If the two binary coded individuals are the same, then Hamming distance between them is 0. The biggest Hamming distance is between two completely different individuals, and the biggest Hamming distance equals the length of the binary coded individual's genes. Hamming distance reflects the difference extent between the two binary coded individuals. The individuals are randomly paired, but crossover operation is not permitted to be done until Hamming distance between the two matched individuals surpasses a threshold value so as to maintain diversity of the group.

4. Conclusion

The smaller is the convergence time, the faster is convergence rate of GA and the higher is its efficiency. The first expected hitting time can be used as an index of convergence rate of GA. The development of research promotes scholars making more studies on convergence and convergence rate of GA. This paper makes studies on calculation of the first expected hitting time based on an absorbing Markov chain and analyzes convergence rate of the concrete GA. The proposed strategy tries to avoid premature convergence by adopting linear scale transformation of fitness function, adaptive crossover and mutation probability and close relative breeding avoidance method.

References

[1] Colak O, Kurbanoglu C. Milling surface roughness in abrasive waterjet machining process using artificial neural networks & regression analysis method. *Mater Process Technol.* 2007; 28: 657-666.

[2] Zain AM, Haron H, Sharif S. Application of GA to optimize cutting conditions for minimizing surface roughness in end milling machining process. Expert Syst Appl. 2010; 37: 4650-4659.

[3] Yu Y, Zhou Z H. *A new approach to estimating the expected first hitting time of evolutionary algorithms*. 21st National Conference on Artificial Intelligence. Boston, MA. 2006: 555-572.

[4] Huang Han, Hao Zhifeng. Convergence rate of ant colony algorithm*. Chinese Journal of Computers*. 2007; 30(8): 1344-1353.

[5] Li Junhua, Li Ming. An analysis on convergence and convergence rate estimate of GA in noisy environments. *Chinese Journal of Electronics*. 2011; 39(8): 1898-1902.

[6] Lu wei, Huang Tianmin. An intuitive explanation of convergence of GA. *Journal of Mianyang Teachers College*. 2002; 21(2): 22-23.

[7] T Nakama. *Theoretical analysis of genetic algorithms in noisy environments based on a Markov model*. Genetic and Evolutionary Computation Conference. Atlanta, Georgia. 2008:1001-1008.

[8] Lee Dukman, Lee Yongsug. Application of neural-network for improving accuracy of roll-force model in hot-rolling mill. *Control Engineering Practice*. 2002;10:473-478.

[9] S.F.Hwang, R.S.He. A hybrid real-parameter genetic algorithm for function optimization. *Advanced Engineering Informatics*. 2006, 20:8-9.

[10] S.Baskar, P.Subbaraj, M.V.C. Rao. Hybrid real coded genetic algorithm solution to economic dispatch problem. *Computers and Electrical Engineering*. 2006, 29:410-412.

[11] H.Lan, X.Wang, L.Wang. Improved genetic annealing algorithm for global optimization of complex function. *Journal of Tsinghua University*. 2002,42:1237-1239.

[12] L.Wang, TZ Shen. An improved adaptive genetic algorithm. *Systems Engineering and Electronics*. 2002; 24:76-78.