Genetic Algorithm and Simulated Annealing for Optimal Robot Arm PID Control

D.P.Kwok Fang Sheng

Department of Electronic Engineering
Hong Kong Polytechnic
Hung Hum, Kowloon, Hong Kong

Tel: 852-7666208
Fax: 852-3628439
Email: enkwokdp@hkpcc.hkp.hk

Technical Session: The IEEE Conference on Evolutionary Computation
-- Applications of evolutionary computation
-- Comparisons between different variants of evolutionary algorithms

Presentation Preferred: Oral

Presenter: D. P. Kwok

0-7803-1899-4/94 $4.00 ©1994 IEEE

Genetic Algorithm and Simulated Annealing for Optimal Robot Arm PID Control

D.P.Kwok Fang Sheng

Department of Electronic Engineering
Hong Kong Polytechnic
Hung Hum, Kowloon, Hong Kong
Tel: 852-7666208
Fax: 852-3628439
Email: enkwokdp@hkpcc.hkp.hk

Abstract -- This paper describes the use of Genetic Algorithm(GA) and Simulated Annealing(SA) for optimizing the parameters of PID controllers for a 6-DOF robot arm. A GA and a SA are designed to optimal-tune the parameters of the PID controller of each joint for a single step response and for the tracking of other specified trajectories. The GA and the SA are required to optimize evaluation functions related to the combinations of different performance indices. Simulations are carried out on a PUMA 560 arm model being controlled by PID controllers with their parameters optimized using the proposed GA and SA. Based on the simulation results, the performances of genetic algorithm and simulated annealing are compared and discussed.

INTRODUCTION

During the past decades, robot arm control techniques have made great advances. Numerous control methods such as adaptive control, neural control, fuzzy control have been studied. However, controllers in industrial and commercial robot arms are still usually PID controllers. Unfortunately, it is quite difficult to optimize the parameter settings of PID controllers because robot arm systems have serious non-linearities and strong couplings. Conventional optimization methods are primarily derivative-driven and sometimes lock on local optima, and as such, they are hard to be applied to this problem, because the search space is non-differentiable, non-linear, time-variant and parameter dependent. Moreover, the PID values obtained, based on conventional optimization methods, frequently need engineer's experience and intuition. Random search method can be used to solve this problem, but, this is not effective because it searches for only a single point using a simple random way in the search space at every step. There is a need for a efficient and effective global optimal approach to optimize the parameter settings of robot PID controllers automatically.

Genetic algorithm(GA) and simulated annealing(SA) have received much interest in recent years. GA is a global search method that is inspired by the mechanics of natural evolution to guide their exploration in a search space. SA is also a global search method that is based on the analogy with the physical annealing process of solids. GA and SA both require little knowledge of the problem itself and need not require that the search space is differentiable or continuous. Therefore, they can solve nonlinear multi-objective optimization problems that difficult using other techniques. GA[1][2][3] and SA[4][5] have been successfully used in a wide variety of areas such as function optimization, design optimization, image processing, schedule optimization, traveling salesman problem, neural networks etc. In the automatic control area, a few applications with GA and SA have been accomplished. Hollstien[6] firstly applied GA to conduct function optimization of computer control systems. Michalewicz[7] utilized GA to discrete-time optimal control problems. Karr[8][9] proposed to use GA for the automatic design of fuzzy controllers. Kokate[10] employed SA to determine the compensator segmentation for rule-based control of slowly varying systems. Chiang[11] used SA for the optimal controller placements in large-scale linear systems.

In this paper, a GA-based approach and a SA-based approach are implemented to optimize the parameters of the robot arm PID controllers for various specified trajectories. Such effective and efficient approaches, with global optimal abilities and good robustness, are expected to overcome some shortcomings of conventional optimization approaches in this situation, and are more feasible for industrial applications.

The paper is organized as follows. Section 2 gives the basic concept and working principle of GA and

0-7803-1899-4/94 $4.00 ©1994 IEEE

SA. Section 3 describes the GA-based approach and SA-based approach for the optimal-tuning of robot arm PID controllers. Section 4 provides some results obtained from the simulation investigations, in which the GAs-based approach and SA-based approach are tested to optimize the robot arm PID controllers for various specified trajectories, and are compared with that obtained by traditional optimization methods. Section 5 summaries the conclusions from this application.

GENETIC ALGORITHM and SIMULATED ANNEALING

Generally, a GA uses three operators: reproduction, crossover and mutation. Given an optimization problem, a GA encodes the parameters concerned into finite bit strings, each of which presents a possible solution to the problem, and then works with a set of strings, called the population, using the three operators in a random way but based on the fitness function evolution iteratively. It performs the basic tasks of copying strings, exchanging portions of strings as well as changing some bits of strings, and finally find and decode the solutions to the problem from the last pool of mature strings. GA searches for a population of points, not a single point, so that they can arrive at the globally optimal point rapidly and meanwhile avoid locking at local optimum. It works with a coding of parameter sets, not the parameter themselves, so that it can get rid of the analytical limitation, such as discontinuities of search spaces,

The algorithm of simulated annealing mainly consists of the repeating of two steps: application of the generation mechanism and application of the acceptance criterion. Given an optimization problem, a SA starting off at a initial random state, a sequence of iterations is generated, each iteration applying a perturbation mechanism which transforms the current state into a next state selected from the neighbourhood of the current state. If this neighbouring state has a lower cost, the neighbouring state is accepted as the current state. If this neighbouring state has a higher cost, the neighbouring state is accepted with a certain probability determined by the acceptance criterion. The most important feature of the simulated annealing is that, besides accepting improvements in cost, it also, to a limited extent, accepts deteriorations in cost. This feature ensures the simulated annealing to be a global search algorithm while it still have the favourable features of local search algorithms, i.e. simplicity and general applicability.

PID OPTIMAL-TUNING

In this paper, GA and SA are applied to the optimization of the robot arm PID controllers.

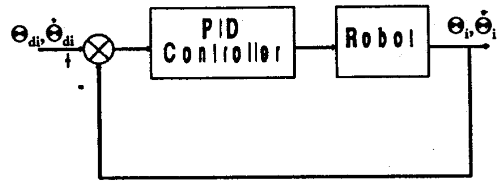


Fig. 1 Configuration of the robot arm PID control system.

Fig. 1 shows the configuration of the robot arm PID control system, where Θ_{id}, $\dot{\Theta}_{id}$ are the desired position, velocity of the joints respectively, and, Θ_i, $\dot{\Theta}_i$ are the actual position, velocity of the joints respectively. With reference to the desired trajectories, all joints of the entire system will coordinate to generate the actual trajectories. The role of the robot arm PID controllers is to drive the joints of the robot arm to move within the user's specifications. Obviously, the settings of the robot arm PID controllers should be fine-tuned so as to meet some strict requirements. Usually, the PID control of each joint of the robot arm is an independent joint PID control as follows:

$$G_c(s) = k_p + k_i \frac{1}{s} + k_d s \tag{1}$$

where K_p, K_i, and K_d are n*1 vectors respectively, if the DOF of robot arm is n, and are proportional gain, integral gain and derivative gain of joints respectively.

In this paper, the designed GA and SA have to automatically optimize the parameter settings of a 6-DOF robot arm modelled under the SUN SPARC Station computer environment using language C. The program consists of two parts: one to implement the GA and the SA, the other to simulate the 6-DOF robot arm. Performance evaluation is carried out on every generation produced under GA and SA.

The implementation of the GA-based optimal-tuning procedures is outlined as follows:

<1> Code the PID parameters K_{pj}, K_{ij} and K_{dj} (j=1,...,n) into binary strings. Each parameter is coded as a 16-bit string. They are translated linearly, according to their boundaries, into a 48n-bits binary string concatenated from 3n 16-bit binary strings.

<2> Produce the initial generation in a simple random way.

<3> Decode the binary strings into the corresponding parameters K_{pj}, K_{ij} and K_{dj}(j=1,...,n).

<4> Simulate the robot arm PID control system using the decoded parameters.

<5> Calculate the fitness function based on different individual and the combinations of various

performance indices. In this paper, the performance indices are adopted as follows:

(a) Integral of sum of squared errors of joints (ISE)

$$\int\sum_{i=1}^{n}\tilde{\theta}_i^2 dt=\sum_{k}\sum_{i=1}^{n}\tilde{\theta}_i^2(kT)\,T \qquad (2)$$

where n is the degrees of freedom, T is the integral step and $\tilde{\Theta}_i$ is the error of joint i.

(b) Integral of sum of squared control torques of joints(IST)

$$\int\sum_{i=1}^{n}(\tau_i/\tau_{im})^2 dt=\sum_{k}\sum_{i=1}^{n}(\tau_i/\tau_{im})^2(KT)\,T \qquad (3)$$

where T_{im} is the maximum drive torque of joint i, and T_i is the drive torque of joint i.

<6> Reproduce a new generation of 48n-bit strings by the roulette wheel selection. Fitter strings have a higher probability of reproducing offspring in the next generation.

<7> Crossover pairs of members in the new generation according to the probability of crossover. Site of crossover is generated randomly.

<8> Mutate every member in the new generation according to the probability of mutation.

<9> Reserve the member of the largest fitness function in the old generation to the new generation.

<10> Repeat <3> to <9> iteratively until it reaches the ending condition which is adopted as the generation number in this program.

In the GA-based optimal-tuning procedure, there are several important genetic parameters that affect the convergence, the ultimate performance and the efficiency of the GA. They are basically the population size, crossover rate and mutation rate. In this paper, these genetic parameters are chosen based on Grefenstette's work [12].

The implementation of the SA-based optimal-tuning procedure is outlined as follows:

<1> Initialize the state *i* in a simple random way

<2> Initialize the control parameter change counter *k* and the control parameter T_k

<3> Calculate the length of Markov chains L_k

<4> Generate the state *j* which is selected from the neighbours of state *i*

<5> Calculate the cost functions F_i, F_j which are related to the different individual and the combinations of various performance indices. If the F_i-F_j is less or equal to 0 , then the state *j* is accepted as the current state; otherwise, the state *j* is accepted with a certain probability which is given by

$$\exp\left(\frac{F_i-F_j}{T_k}\right) \qquad (4)$$

<6> Repeat <4><5> until it reaches the length of Markov chains L_k

<7> Calculate the control parameter as follow:

$$T_{k+1}=\alpha\cdot T_k,\ k=1,2,\ldots \qquad (5)$$

where α is a decrement constant smaller than but close to 1.

<8> Increase k by 1 and repeat <3> to <7> iteratively until it reaches the ending condition which is adopted as the generation number in this program.

In the SA-based optimal-tuning procedure, the initial control parameter, decrement constant and the length of Markov chains have to be carefully chosen to ensure good final solution quality.

The simulation of the 6-DOF robot arm system [13] includes trajectory planning, kinematics and dynamics computation, control system simulation in consideration of practical factors such as the driving torque limitation of each joint, maximum speed of each joint, maximum and minimum angles of each joint etc. The trajectory planning is a process by which the desired positions and velocities of joints are interpolated from the path points in the joint space or in the Cartesian space in every planning time period. The computation of the kinematics is achieved by homogenous transformations of matrices and the calculation of the Jacobian matrix. The computation of the dynamics is implemented by Newton-Euler numerical method.

EXPERIMENTAL INVESTIGATIONS

The GA and SA designed are being investigated through two simulation experiments in which a PUMA 560 is controlled for tracking some specified trajectories using PID controllers, with their parameters optimized according to various performance indices related fitness functions. The dynamic model of PUMA 560 is obtained from T. J. Tarn[14]. In Experiment 1, the GA and SA are tested for optimizing the controller for a step motion tracking. In Experiment 2, the GA and SA are examined to optimize the controller for a circle tracking. These results are compared with that of the random search method and the empirical method. For the two experiments, the sampling time period of the PID controller is set at 10ms, and the boundaries of the PID parameters are chosen as K_{pmj}(j=1,...,6)=1000.0, K_{imj}(j=1,...,6)=1000.0, and K_{dmj}(j=1,...,6)=100.0.

Experiment 1

In the step motion tracking, the step amplitudes of the six joints are 0.5, ISE and IST are selected as the performance indices.

For the GA, the parameters chosen are as follows: the population size is 100, the crossover rate is 0.9, the mutation rate is 0.01 and the generation number is 40. The fitness function is expressed as bellow:

$$\frac{1}{k_1 * ISE + k_2 * IST} \qquad (6)$$

For the SA, the parameters are selected as follows: the initial control parameter is 0.8, the length of Markov chains is 100, the decrement constant α is 0.85 and the generation number is 40. The cost function is $K_1*ISE+K_2*IST$. The values of K_1, K_2 are chosen in the same manner as the selections in the GA approach.

Fig.2(a) and (b) show the step responses of the six joints of PUMA 560 based on the GA and the SA. Fig.3(a) and (b) show the convergent tendency of the $K_1*ISE+K_2*IST$ function obtained using the GA and the SA in simulation of the step responses of the six joints of the PUMA 560. The final value of $K_1*ISE+K_2*IST$ function obtained using the GA and the SA are 0.153931 and 0.173394 respectively.

The results obtained from this experiment show that the GA-based and SA-based optimal-tuning approach can be used to obtain the optimal settings of PID parameters of the robot arm under performance indices related evaluation functions, and the GA-based optimal-tuning converged faster than SA-based optimal-tuning.

Experiment 2

In the circle tracking, three space points are defined to determine a circle in the working space of PUMA 560. ISE is selected as the performance index. The parameters for the GA and the SA are chosen in the same manner as the selections in Experiment 1. The fitness function of the GA is 1/*ISE*. The cost function of the SA is *ISE*. In this experiment, the random approach and the empirical approach are compared with the GA and SA designed for the same circle tracking. Fig.4(a)-(d) show the trajectories for tracking the same circle using the four optimal settings of the PID parameters obtained through the GA, SA, random and empirical approaches. The total tracking time is 5.3s.

The ISE for tracking the circle using the GA, SA, random and empirical approach are listed in Table 1.

Circle	ISE
GA	1.11567
SA	1.52551
Random	2.55083
Empirical	24.1659

Table 1. The ISE for tracking a circle using the GA, SA, random and empirical.

From this experiment, it can be recognized that the GA and the SA could produce the smaller performance indices, compared with the random and empirical approaches during the common observation period, and the GA produced the smallest performance index.

CONCLUSIONS

In this paper, the GA-based and SA-based optimal-tuning techniques are used to optimize the parameter settings of the robot arm PID controllers. The design, implementation and investigation of the proposed GA-based approach and SA-based approach have been described and compared with those based on traditional optimization methods. The results obtained from the two investigative experiments on simulation demonstrate that the GA-based and SA-based optimal-tuning techniques can work effectively and efficiently and have great potential to become common optimal-tuning approaches for the robot arm controllers. The performance of GA-based approach appeared to be better than SA-based approch. The features and the advantages of GA and SA such as problem-independent, global optimization, good robustness, simple mechanics and multi-objective driven will make the GA-based and SA-based optimal-tuning techniques effective and efficient optimization techniques in the control field.

REFERENCES

[1] J. H. Holland, Adaptation in Natural and Artificial Systems, Ann Arbor, MI: Univ. Mich. Press, 1975.

[2] D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Inc., 1989,pp.89-145.

[3] D. Lawrence(ed.), Handbook of Genetic Algorithms, New York: Van Nostrand Reinhold, 1991, ch2, pp103-372.

[4] P. J. M. van Laarhoven and E. H. L. Aarts,

Simulated Annealing: Theory and Applications, Kluwer Academic Publishers, 1987.

[5] E. H. L. Aarts, Simulated Annealing and Baltzmann Machines: a Stochastic Approach to Combinatorial Optimization and Neural Computing, Wiley, 1989.

[6] R. B. Hollstien, "Artificial Genetic Adaptation in Computer Control Systems", Doctoral Dissertation, University of Michigan, 1971.

[7] Z. Michalewicz, J. B. Krawczyk, M. Kazemi, C. Z. Janikow, "Genetic Algorithms and Optimal Control Problems", Proc. of IEEE Int. Conf. on Decision and Control, Vol.3, 1990, pp.1664-1666.

[8] C. Karr, "Genetic Algorithms for Fuzzy Controllers", AI Expert, Vol.6, pp.26-33, Feb., 1991.

[9] C. Karr, "Applying genetics", AI Expert, Vol.6, pp.38-43, Mar., 1991.

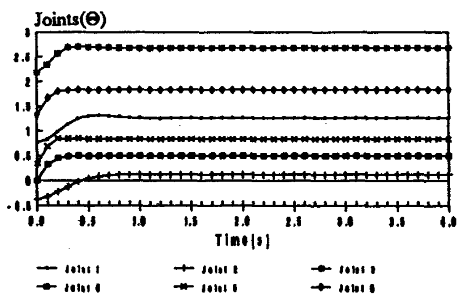
[10] P. Y. Kokate, M. Das and R. N. K. Loh, "Rule-based Control of Slowly Varying Systems Using Compensator Segmentation Determined by Simulated Annealing", IEEE Transactions on Industrial Electronics, Vol. 39, No. 4, pp.359-367, August 1992.

[11] H. D. Chiang, J. S. Thorp, J. C. Wang, J. Lu and B. Aubert, "Optimal Controller Placements in Large-scale Linear Systems", IEE Proceedings, Part D: Control Theory and Applications, Vol.139, No.1, pp.79-87, January 1992.

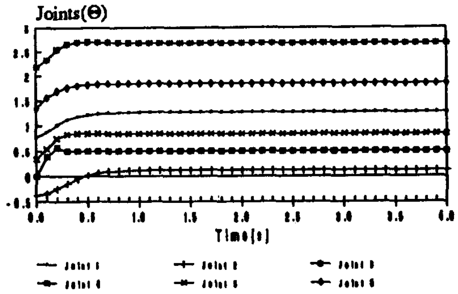
[12] J. J. Grefenstette, "Optimization of Control Parameters for Genetic Algorithms", IEEE Trans. on System, Man, and Cybernetics, Vol. Smc-16, no1, pp.122-128, Jan./Feb., 1986.

[13] Zengqi Sun and Xien Fan "A Simulation System for Dynamic Analysis and Control Design of Robots", Proc. of Beijing Int. Conf. on System Simulation and Scientific Computing, Oct. 1989, pp.23-26.

[14]T. J. Tarn, Dynamic Equations for Six-link PUMA560 Robot Arm, Report SSM-RL-86-05, Washington University,USA,1986.

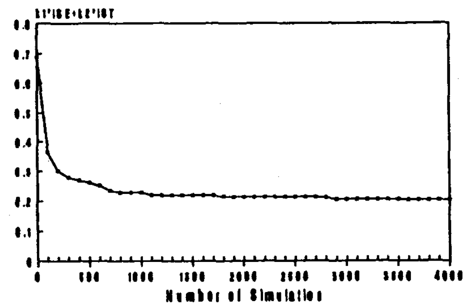


(a) GA

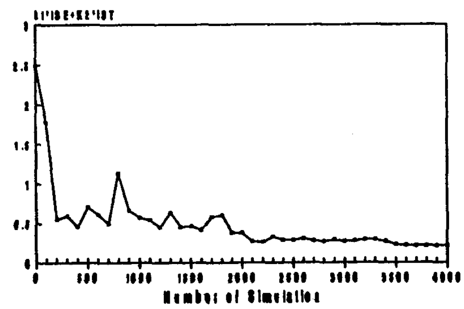


(b) SA

Fig.2 Step Responses of six joints of PUMA 560 based on (a) GA, (b) SA.

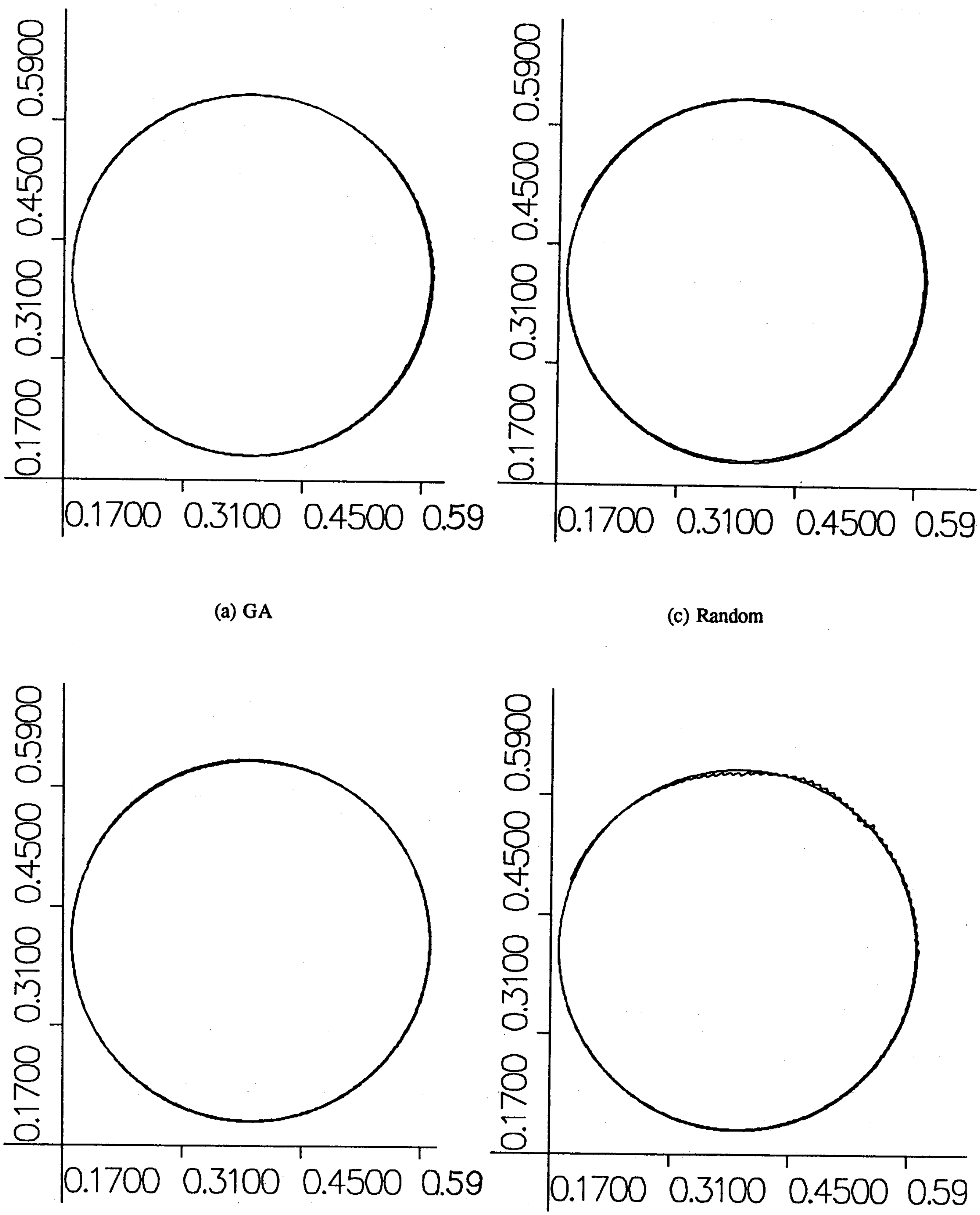


(a) GA



(b) SA

Fig.3 Convergent tendency of K_1*ISE+K_2*IST based on (a) GA, (b) SA.



(a) GA

(c) Random

(b) SA

(d) Empirical

Fig.4 Trajectories for tracking a circle based on (a) GA, (b) SA, (c) random and (d) empirical.