15th Triennial World Congress, Barcelona, Spain

ROBUST STRICTLY POSITIVE REAL SYNTHESIS BASED ON GENETIC ALGORITHM

Liangjun Xie[1] **Long Wang**[2] **Wensheng Yu**[1] **Yuhuang Qiu**[1]

1 Laboratory for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, P. R. China
2 Center for Systems and Control, Department of Mechanics and Engineering Science, Peking University, Beijing 10087, P. R. China

Abstract: In this paper, a new numerical method based on Genetic Algorithm (GA) for robust Strictly Positive Real (SPR) synthesis is presented. The algorithm works well in coefficient space of continuous-time systems and is computationally efficient for some types of polynomial families, such as polynomial segments, interval polynomials and polytopic polynomials et al.. The method can be easily extended to the discrete-time systems. Illustrative examples are provided showing that the method is rather effective for arbitrary given high-order systems. *Copyright ©2002 IFAC*

Key words: Robustness, Strict Positive Realness (SPR), Genetic Algorithm (GA), Robust Analysis and Synthesis.

1. INTRODUCTION

The robust Strict Positive Realness (SPR) of transfer functions is an important performance of dynamic systems. It plays a crucial role in absolute stability and hyperstability (Popov, 1973), passivity analysis, and adaptive system theory (Landau, 1979). Since there always exist uncertainties in the real systems, it is imperative to study the robust strict positive realness. In recent years, stimulated by the parameterization approach in the robust stability analysis (Bhattacharyya, et al., 1994; Barmish, 1994), much attention has been paid to the study of robust strict positive realness of dynamic systems, and much progress has been made (Anderson, et al., 1990; Betser and Zeheb, 1993; Chapellat, et al., 1991; Hollot, et al., 1989; Wang and Huang, 1991; Wang and Yu, 2000; Yu, 1998; Yu and Huang, 1998; Yu and Wang, 2001a; Yu and Wang, 2001b). It was proved by Wang and Huang (1991), Chapellat et al. (1991) that the strict positive realness of an entire family of interval transfer functions can be ascertained by the same property of prescribed eight vertex transfer functions. However, most available results belong to the category of robust strictly positive real analysis. There are very few valuable results on the robust strictly positive real synthesis. Synthesis problems are of more practical significance from the engineering application's viewpoint. Much work remains to be done in robust strictly positive real synthesis.

The basic statement of the robust strictly positive real synthesis is as follows: Given an *n-th* order robustly

The project is supported by National Natural Science Foundation of China (69925307, 60075018) and National Lab. of Intelligent Technology and Systems of Tsinghua University

stable polynomial set F, does there exist, and how can we construct a (fixed) polynomial $b(s)$ such that, $\forall a(s) \in F$, $b(s)/a(s)$ is strictly positive real?

For the robust strictly positive real synthesis problem above, existing results show that: If the elements of F have the same even (or odd) parts, such a polynomial $b(s)$ always exists (Hollot, et al., 1989); If F is a low order (n $\leq$4) stable interval polynomial set (Anderson, et al., 1990; Betser and Zeheb, 1993; Hollot, et al., 1989; Wang and Yu, 2000; Yu, 1998; Yu and Wang, 2001a), such a polynomial $b(s)$ always exists; If F is a low order (n $\leq$5) stable polynomial segment (Wang and Yu, 2000; Yu, 1998; Yu and Huang, 1998; Yu and Wang, 2001b), such a polynomial $b(s)$ always exists. Some sufficient conditions for robust synthesis are presented in (Anderson, et al., 1990; Betser and Zeheb, 1993; Yu,1998; Wang and Yu, 2000), especially, the proposed design method in (Wang and Yu, 2000), by introducing the concept of Weak Strict Positive Realness (WSPR) and giving a complete characterization of the (weak) SPR regions for transfer functions in coefficient space, is numerically efficient for high order systems and the derived conditions are necessary and sufficient conditions for low order stable interval polynomials (n$\leq$4) and segment polynomials (n $\leq$5). However, the method in (Wang and Yu, 2000) had not provided any convex programming algorithm for the intersection of weak strictly positive real regions. This subject deserves further investigation. Betser and Zeheb (1993) and Yu (1998) dealt with this problem using Matrix Equations (MEs) or Linear Matrix Inequalities (LMIs). However, the order of involved MEs or LMIs may be high, many variables must be introduced, and there is no theoretic result of the feasible conditions for the MEs or LMIs.

In this paper, a new algorithm based on Genetic Algorithm (GA) is proposed to synthesize the robust SPR for a family of polynomials. GA is a parallel problem solver, which uses ideas and gets inspira -tions from natural evolutionary processes. Due to its intrinsic parallelism and some intelligent properties, GA has been applied successfully to problems where heuristic solutions are not available or generally lead to unsatisfactory results. In recent years, GA has proved to be efficient in a large number of optimi -zation and search tasks (Fernandez-anaya, et al., 1997; Goldberg, 1989; Li, et al., 1996; Rechenberg, 1989). Moreover, GA using real-valued encoding deals directly with real number, thus saving more computation of mapping from the solution space to bit strings, contrary to what simple genetic algorithm does.

The problem under consideration involves a non -empty, bounded convex set in R^{n-1} (Wang and Yu, 2000), thus fulfilling the applicability criteria of GA (Li, et al., 1996). Furthermore, there is no method that leads to satisfactory results, especially for the condition with many vertices and high order. Thereby, given these characteristics of GA, it is selected as a heuristic method and suits for constructing a polynomial, the ratios of which to a finite collection of polynomials could be SPR (WSPR).

An outline of the paper is as follows. In Section 2, preliminaries about SPR are introduced. The main frame of our algorithm and details are presented in Section 3. Numerical examples are provided to illustrate the effectiveness of this method in Section 4. Then conclusion appears in Section 5.

2. PRELIMINARIES

The concept of strict positive realness stems from different areas such as control systems and network analysis. There is some slightly difference in definitions in the literature (Yu, 1998; Wang and Yu, 2000). In this paper we employ the following definitions from (Yu, 1998; Wang and Yu, 2000):

Denote P^n as the set of all $n-th$ order polynomials of s with real coefficients, R^n as n-dimensional real field. Denote $H^n \subset P^n$ as the set of all $n-th$ order Hurwitz stable polynomials.

In the following definitions, $b(\cdot)$, $a(\cdot) \in P^n$, $p(s) = b(s)/a(s)$ is a rational function.

Definition 1. $p(s)$ is said to be strictly positive real (SPR), denoted as $p(s) \in SPR$, if $b(s) \in P^n$, $a(s) \in H^n$, and $\forall w \in R$, Re[$p(jw)$]>0.

Definition 2. $p(s)$ is said to be weak strictly positive real (WSPR), denoted as $p(s) \in WSPR$, if $b(s) \in P^{n-1}$, $a(s) \in H^n$, and $\forall w \in R$, Re[$p(jw)$]>0.

Definition 3. Given $a(s) \in H^n$, the set of the coefficients (in R^{n+1}) of all the $b(s)$'s in P^n such that $p(s) := b(s)/a(s) \in$ SPR is said to be the SPR region associated with $a(s)$, denoted as Ω_a.

Definition 4. Given $a(s) \in H^n$, the set of the coefficients (in R^n) of all the $b(s)$'s in P^{n-1} such that $p(s) := b(s)/a(s) \in$ WSPR is said to be the WSPR region associated with $a(s)$, denoted as Ω_a^W.

For notational convenience, $\Omega_a(\Omega_a^W)$ sometimes also stands for the set of all the polynomials $b(s)$ in $P^n(P^{n-1})$ such that $p(s) := b(s)/a(s) \in$ SPR (WSPR).

From the definitions above, it is easy to get the following properties:

Property 1. (Hollot, et al., 1989) Given $a(s) \in H^n$, Ω_a is a non-empty, open, convex cone in R^{n+1}.

Property 2. (Popov, 1973; Landau, 1979; Huang, 1989) Given $a(s) \in H^n$, we have $\Omega_a \subset H^n$, $\Omega_a^W \subset H^{n-1}$.

Property 3. (Popov, 1973; Landau, 1979; Huang, 1989) Given $a(s) \in H^n$, $b(s) \in H^m$, if $\forall w \in R$, Re[$p(jw)$]>0, then $|m\text{-}n| \leq 1$.

Without loss of generality, let $a(s) = s^n + a_1 s^{n-1} + ... + a_n \in H^n$. Denote Ω_{1a} as the set of the coefficients of all the $b(s) = s^n + x_1 s^{n-1} + ... + x_n \in P^n$, such that $p(s) = b(s)/a(s) \in SPR$. And denote Ω_{1a}^W as the set of the coefficients of all the $b(s) = s^{n-1} + x_1 s^{n-2} + ... + x_{n-1} \in P^{n-1}$, such that $p(s) = b(s)/a(s) \in$ WSPR.

Some results in (Wang and Yu, 2000) will be used in this paper.

Theorem 1. (Wang and Yu, 2000) Given $a(s) \in H^n$, Ω_{1a} is a non-empty, open, unbounded convex set in R^n.

Theorem 2. (Wang and Yu, 2000) Given $a(s) \in H^n$, Ω_{1a}^W is a non-empty, bounded convex set in R^{n-1}.

Theorem 3. (Wang and Yu, 2000) Given $a(s) \in H^n$, if $(x_1, x_2, ..., x_{n-1}) \in \Omega_{1a}^W$, then $\forall (1, a_1, a_2, ..., a_n)$), we can take sufficiently small $e > 0$ such that $(0, 1, x_1, x_2, ..., x_{n-1}) + e\ (1, a_1, a_2, ... a_n) \in \Omega_a$.

From (Wang and Yu, 2000), we also know that the boundary of every entry of x is:
$(x_1, x_2, ..., x_{n-1}) \in \Omega_{1a}^W$ i.e.,
$\Omega_{1a}^W \subset \{(x_1, x_2, \cdots, x_{n-1}) | x_1 \leq a_1, x_2 < a_2, \cdots, x_{n-1} < a_{n-1}\}$.

Since Ω_a and Ω_{1a} are both unbounded sets, when considering the robust SPR synthesis problem, it is hardly tractable operating on unbounded set to check the intersection of SPR regions. On the other hand, from the Theorem 2, 3 and the boundary of Ω_{1a}^W listed above, we can construct the finite search space for this particular problem that is one of the applicability criteria of GA. That's why we first con-sider the problem from Ω_{1a}^W. Furthermore, Theorem 3 reveals the relationship between Ω_{1a}^W and Ω_a and plays an important role in robust SPR synthesis. Moreover, from property 2, there is one variable less in Ω_{1a}^W than Ω_{1a}, thus saving the computation.

3. COMPUTATIONAL METHOD

3.1. Algorithm description

Based on genetic algorithm, the main procedures of the new numerical algorithm are as follows:

Step 1. For the input vertices of polynomials, test the robust stability of the convex hull of F, i.e. $\overline{F}$. If $\overline{F}$ is robustly stable, then go to Step 2; otherwise, print "there does not exist such a b(s)"; (by Definitions 1 and 2)

Step 2. Set the search space;

Step 3. Call the main program of genetic algorithm for WSPR. If succeed in searching, then go to Step 4; Otherwise, go to Step 5;

Step 4. Take a sufficiently small $e > 0$ such that ($(e, 1, x_1, x_2, ..., x_{n-1}) \in \bigcap_{i=1}^{m} \Omega_{ai}$. Hence, the n-th order polynomial $c(s)$ with coefficients $(e, 1, x_1, x_2, ..., x_{n-1})$ satisfies the design requirement; (by Theorem 3)

Step 5. Appropriately enlarge the search space of $(x_1, x_2, ..., x_n)$, and call the main program of genetic algorithm for SPR. End.

Remark 1. By the Theorem 2 and 3, we can first get the search space as follows: D: $= \{(x_1, x_2, ..., x_{n-1}) \mid 0 < x_l < a_l, l = 1, 2...n$, $a_l = \min\ \{ a_l^{(i)}, \ i = 1, 2, \ldots, m\}$, $l = 1, 2, \ldots, n-1$.

Remark 2. In order to get the sufficiently small positive number e, noting that the *n-th* polynomial $(e, 1, x_1, x_2, ..., x_{n-1})$, i.e. $c(s)$, must be Hurwitz stable (by Property 2). Employing a necessary condition of *n-th* Hurwitz polynomials (Huang, 1989), we can at first set the domain of e in the concept of stability, i.e. $e < x_1 x_2 / x_3$. And then get the SPR domain of e by using a simple bisection algorithm.

Remark 3. The main programs for WSPR and SPR are the same except for the differences in the number of variables and their boundaries of the search space.

Remark 4. Our method can be easily extended to the discrete-time systems. In fact, by using linear fractional transformation, we can transform discrete-time systems into continuous-time systems.

3.2. The main frame of GA

The algorithm's efficiency depends on the adequate strategy selection for the problem under consideration. In this particular case the selected strategy, which constructs the main frame of the main GA, is the following.

After encoding, N individuals (parents) of the population are selected, crossed over, mutated, thus producing a new population (offspring). Then each individual of this population is assigned its corresponding value. The procedures above are repeated until the pre-set number of iterations (M) is completed or the task is solved. Details are discussed as follows:

Encoding: In robust SPR (WSPR) synthesis for n-th order polynomial set, what we need is to find an n-th or (n-1)-th order monic polynomial in coefficient space, so the particular codification assumes the n or (n-1) coefficients as an individual in the form of a vector, i.e. $x=(x_1,x_2,...,x_n)$ or $x=(x_1,x_2,...,x_{n-1})$.

In order to improve the performance, a number of Hurwitz stable polynomials (randomly selected from the search space.) are pre-given as a part of initial population and others are generated randomly from the search space.

Selection: In order to improve the GA's performance, "Elitism mold" is adopted which forces the GA to retain some number of the best individuals at each generation (in this paper, denote the number as Ne). Those best individuals are directly selected to form a part of the offspring. By using the fitness -proportionate selection with "Roulette wheel" and "Stochastic Universal" sampling (Goldberg, 1989), the N individuals (parents) of the population are selected, producing one population of offspring. From which we choose (N–Ne) individuals and treat them with crossover and mutation and after that they form the other part of the offspring population.

Crossover: By randomly mating, pairs of individuals are selected by the probability of p_c and crossed over with each other. In this particular case, "global recombination" is used just as follows:

$$z_i = a_i x_i + (1-a_i)y_i\,,\quad i=1,2,\cdots,(N-Ne)/2$$

where a_i is uniform random distribution selected from[0,1], x_i, y_i are parents and z_i offspring.

Mutation: For GA with real-valued encoding, muta -tion has played one of the most important parts in genetic operators, thus intensified mutation has been adopted. The strategies presented by Rechenberg (1989) have been modified. In this particular case we use "BGA" strategy as follows.

Let $x=(x_1,x_2,...,x_n)$ as a parent's vector, then the entry x_i is selected to mutate by the probability of p_m. The offspring vector is $x'=(x_1,x_2,...,x_k',...,x_n)$. After that, x_k' is assigned to be $x_k+0.1\Delta_k*d$ or $x_k-0.1\Delta_k*d$ with the same probability. Here

$$d=\sum_{j=0}^{t}a_j 10^{-j}\,,\quad a_j\in\{0,1\}\,,\quad \Delta_k=u_k-l_k\,,$$

l_k and u_k are the lower and upper bounds respectively, and t, is a positive integer, chosen by the precision of mutation needed. If x_k' is beyond the search space $[l_k,u_k]$, then transform it with the following formula:

$$x_k'=\begin{cases}(x_k'-l_k+0.1(u_k-l_k)\cdot r+l_k & x_k'<l_k\\ u_k-(u_k-x_k'+0.1(u_k-l_k)\cdot r & x_k'>u_k\end{cases}$$

where r is selected randomly from (0,1).

As a means of avoiding local convergence, a more great probability of mutation, denoted by p_{mc}, is considered and it is controlled by the parameter e that takes effect when the best fitness of the population is maintained constant during e iterations (Fernandez -anaya, et al., 1997; Rechenberg, 1989).

Fitness function: Fitness function is used to evaluate the status of each individual. In this particular case, the value is given by the efficacy of $b(s)$ belonging to Ω_{1a}^{W} (or Ω_a). For a given polynomial $a(s)$, the GA iteratively optimizes the population depending on the value of fitness function. Since WSPR (or SPR) requires Re[$p(jw)$]>0 and $b(s)$ Hurwitz stable, for m polynomials $a^i(s)$, $i=1,2,...,m$, if the algorithm finds a polynomial $b(s)$ that simultaneously to be WSPR (SPR) with $a(s)$, then the value of fitness function of the feasible $b(s)$ is located in (2, 3).

The fitness function F of the GA is proposed as follows:

$$F(a,b)=\sum_{i=1}^{m}\frac{fit(i)}{m}\ \mathrm{sgn}(hur(b))$$

where

$$fit(i)=(\frac{y(i)}{fd}+2)\mathrm{sgn}(y(i))+e^{y(i)}(1-\mathrm{sgn}(y(i)))\,,$$

$$\mathrm{sgn}(x)=\begin{cases}1 & x>0\\ 0 & x\le 0\end{cases}\,,$$

$$\mathrm{hur}(b)=\begin{cases}1 & b(s)\in H^{n-1}\\ 0 & b(s)\notin H^{n-1}\end{cases}\,,$$

$$fd=\max(\max(y),\min(y),\max(y))$$

$$y(i)=\inf_{w\in[0,+\infty)}[\frac{1}{|a^i(jw)|^2}\cdot\sum_{l=0}^{n}c_l w^{2(n-l)}]$$

where $c_l=\sum_{k=0}^{n}a_k x_{2l-k}(-1)^{l+k}$, $a_0=1$, and if $i<0$ or $i>n$, let $a_i=0, x_i=0$, $l=0,1,..,n$, When all sgn($y(i)$)>0, this means that $b(s)$ is a solution, and then jump from GA and using Theorem 3, we ultimately get the $b(s)$ for SPR synthesis.

4. EXAMPLES

In what follows, some examples are provided to show the advantages of the GA developed in the previous section. Here, for convenience, we represent the polynomial $a(s) = s^n + a_1 s^{n-1} + ... + a_n$ as a vector form $[1, a_1, a_2, ..., a_n]$. Every row of a matrix represents a polynomial when there are many polynomials in question.

Example 1. Consider a family of 4-th order polynomial set F=[1 89 56 88 1; 1 11 56 88 50; 1 89 56 88 50; 1 11 56 88 1]. It is easy to see that the convex hull $\overline{F}$ is robustly stable. Using our GA method, we easily get the feasible WSPR $b(s)$ =[1 9.0021 4.0386 3.1104] with only 22 iterations. The initial parameters are: N=40, M=90, e=3, $p_m = 0.1$, $p_c = 1$, p_{mc}=0.7. The following figure of Example 1 shows the value evolution of the best individual of each population. Thus let $c(s) := es^n + b(s)$, e >0, by Step 4, it is easy to find that when $e \leq 0.5562$, $c(s)$ meets the design requirement.

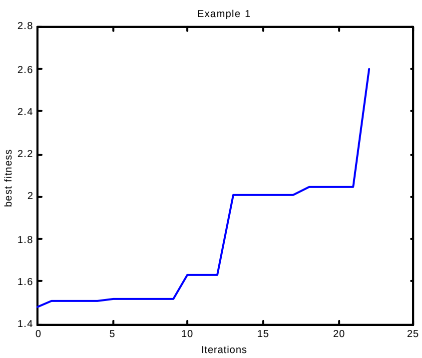

Fig. 1. Example 1.

Example 2. Consider a family of 9-th order poly -nomial set F=[1 11 52 145 266 331 280 155 49 6; 1 11 52 146 265.5 332 278.5 151 48 2]. It is easy to see that the convex hull $\overline{F}$ is robustly stable. Using our GA method, we easily get the feasible WSPR $b(s)$ = [1 5.243 34.296 64.2615 150.495 109.9287 130.8705 39.8864 18.084] with only 11 iterations. The initial parameters are: N=30, M=90, e=3, $p_m = 0.1$, $p_c = 1$, p_{mc}=0.8. Thus let $c(s) := es^n +$ $b(s)$, e >0, by Step 4, it is easy to find that when e $\leq$0.051, $c(s)$ meets the design requirement.

The examples above are taken from (Wang and Yu, 2000) where robust synthesis is done by grid.

As far as we know, there is not any example of order higher than 9 in the literature for robust strict positive real synthesis.

In what follows, we will provide some more exam -ples for high order polynomial families.

Example 3. Consider a family of 13-th order poly -nomial set F=[1 12.431 70.1468 237.5119 537.3627 855.8573 984.3782 824.8223 501.1529 216.7529 64.3521 12.2922 1.3374 0.062; 1 11.1952 57.0195 174.8476 359.9367 524.9087 557.4587 436.1066 251.034 104.871 30.8559 6.0438 0.7048 0.0369; 1 12.1901 67.373 223.0708 492.3984 762.7923 849.83 685.9540 398.4889 163.0798 45.1236 7.8441 0.7448 0.0277]. It is easy to see that the convex hull $\overline{F}$ is robustly stable. Choose the search space as follows: [10.0122 45.8284 126.9678 237.53 316.8444 310.016 225.2749 121.5694 48.1199 13.5797 2.5875 0.2984]; Using our GA method, we easily get the feasible WSPR $b(s)$ = [1 7.4164 41.7132 110.8354 231.8102 302.558 301.8614 219.7843 120.0124 46.405 12.513 2.021 0.179] with only 20 iterations. The initial para -meters are: N=30, M=120, e=3, $p_m = 0.2$, $p_c = 1$, p_{mc}=0.9. Thus let $c(s) := es^n + b(s)$, e >0, by Step 4, it is easy to find that when $e \leq 0.0117$, $c(s)$ meets the design requirement.

Example 4. Consider a family of 15-th order poly -nomial set F=[1 15 105 455 1365 3003 5005 6435 6435 5005 3003 1365 455 105 15 1; 1 14.9 103.5 445 1323.2 2883.4 4756.4 6048.4 5977.9 4592 2719.2 1219 400.4 91 12.8 0.8]. It is easy to see that the convex hull $\overline{F}$ is robustly stable. Choose the search space as follows: [14.5 98 410.2 1187.1 2517.8 4042.9 5004.7 4815.5 3601.5 2076.6 906.5 290 64.2 8.8]; Using our GA method, we easily get the feasible WSPR $b(s)$ = [1 13.7 93.5 396.1 1161.4 2474.7 4015.5 4970.3 4761 3576.8 2050.9 896.5 283.9 59.6 7.6] with only 26 iterations. The initial parameters are: N=30, M=120, e=3, $p_m = 0.2$, $p_c = 1$, p_{mc}=1. Thus let $c(s) := es^n + b(s)$, e >0, by Step 4, it is easy to find that when $e \leq 0.009$, $c(s)$ meets the design requirement.

Note that, for the vertex set of a general polytopic polynomial family F, even if $\overline{F}$ is robustly stable, it is still possible not to exist a polynomial *b(s)* for WSPR, i.e. there is no intersection for WSPR regions. Thus Step 3 fails, but it is effective for the Step 5 to solve this problem.

Example 5. Consider a family of 3-th order poly -nomial set F=[1 2.6 37 64; 1 17 83 978; 1 15 28 415]. It is easy to see that the convex hull $\overline{F}$ is robustly stable. However, [14] has proved that the intersection of WSPR regions is empty. Hence, no *b(s)* exists to meet the requirement. By Step 5, we easily get the feasible SPR *b(s)* = [1 4.8372 64.7864 81.4196] meeting the design requirement. The initial para-meters are: N=30, M=90, e=3, $p_m = 0.2$, $p_c = 1$, p_{mc}=0.8. The upper boundary of the enlarged search space is 3*min(a), i.e. 3× [2.6 28 64].

5. CONCLUSIONS

In this paper, a new numerical method using genetic algorithm for the robust SPR synthesis has been presented. The algorithm works well in coefficient space of continuous-time systems and is computa-tionally efficient for some types of polynomial families. Illustrative examples have been provided showing that the method is rather effective for arbitrary given high order robust stable polynomial families.

REFERENCES

Anderson, B. D. O., S. Dasgupta, P. Khargonekar, et.al. (1990). Robust strict positive realness: characterization and construction, *IEEE Trans. Circuit and Systems*, **CAS-37,** 869-876.

Barmish, B. R. (1994). *New tools for robustness of linear systems*. MacMillan publishing company, New York.

Betser, A. and E. Zeheb (1993). Design of robust strictly positive realness transfer functions. *IEEE Trans. Circuits and Systems*, **CAS-40,** 573-580.

Bhattacharyya, S. P., H. Chapellat and L. H. Keel (1995). *Robust control: The parametric approach*, Prentice Hall, New York.

Chapellat, H., M. Dahleh and S. P. Bhattacharyya (1991). On robust nonlinear stability of interval control systems. *IEEE Trans. Automatic Control*, **AC-36**, 59-69.

Fernandez-anaya, G., S. Munoz-gutierrez, R. A. Sanchez-guzman and et.al. (1997). Simultaneous stabilization using evolutionary strategies, *Int. J. of Control*, **68**, 1417-1435.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*, Addison Wesley.

Hollot, C. V., L. Huang and Z. Xu (1989). Designing strictly positive real transfer function families: a necessary and sufficient condition for low degree and structured families. *Proc. of the international conference on mathematical theory of network and systems*, 215-227.

Huang, L. (1989). *Stability theory*. Peking University Press, Beijing.

Landau, I. D. (1979). *Adaptive control: the model reference approach*, Marcel Deker, New York,.

Li, Y. et al. (1996). Genetic algorithm automated approach to the design of sliding mode control systems. *Int. J. of Control*, **63**, 721-739.

Popov, V. M. (1973). *Hyperstability of control systems*, Springer-Verlag, New York.

Rechenberg, I. (1989). Evolution and artificial intelligence. *Machine learning principles and techniques*, (R. Forsyth (ED)).

Wang, L. and L. Huang (1991). Finite verification of strict positive realness of interval transfer function. *Chinese Science Bulletin*, **36**, 262-268.

Wang, L. and W. Yu (2000), Complete characteri-zation of strict positive realness regions and robust strictly positive real synthesis method. *Science in China: E*, **43**, 97-112.

Yu, W. (1998). *Robust SPR synthesis and robust stability analysis.* PhD Thesis, Peking University, Beijing.

Yu, W. and L. Huang (1998). Necessary and sufficient conditions for strictly positive real stabilization of low order systems. *Science Bulletin*, **43**, 2275-2279.

Yu, W. and L. Wang (2001a). Andersons's claim on fourth-order SPR synthesis is true. *IEEE Trans. Circuits Syst.; Part I*, **CAS-48**, 506-509.

Yu, W. and L. Wang (2001b). Robust strictly positive real synthesis for convex combination of fifth-order polynomials, *2001 IEEE Circuits and Systems Conference (ISCAS 2001)*, **1,** 739-742, Sydney, Australia.