2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Selection of Weighing Functions in H∞ Controller Design using PBIL

P. Munawa, KA Folly

***Abstract*—H∞ optimal control technique is seen as a promising robust control technique that can effectively deal with the problems of model uncertainties. However, for H∞ optimal control design to be successful one must be able to choose adequate performance and uncertainty weights. Until now, there is no a systematic way of choosing these weighting functions; they are generally selected based on trial and error. This approach not only is ineffective but also time consuming. In this paper, a systematic way of selecting the weighting functions in H∞ optimal control is proposed. The selection of adequate weighting function is formulated as an optimization problem and solved using Population Based Incremental Learning (PBIL) Algorithm.**

I. Introduction

POWER system stabilizers (PSSs) are local controllers that are widely used in the power industry for supplementary control of generator excitation control systems [1]. They are used to damp low frequency electromechanical oscillations in the frequencies range of 0.2– 2.5 Hz . The oscillations of this nature arise due to changes in system loads, transmission network, and varying operating conditions [2].

The conventional methods for PSS design involve eigenvalue techniques, root locus, pole placement and linear optimal control [3]-[4]. Although these techniques are useful, they do not directly address the issues of model uncertainties. As a result, controller designed using the above techniques are not robust. The robustness requirement in this context means that the PSSs should be able to provide sufficient damping even when subjected to a wide range of operating conditions.

In the last two decades, researchers have developed advanced control such robust control (H∞ optimal control) in order to address the limitations associated with the conventional design methods [7]-[10]. Although H∞ optimal control can address directly the issue related to model uncertainties, there is no systematic way to select the performance and uncertainty weighting functions upon which a successful design relies. In general, the weighting functions are selected based on trial and error. This approach is not only ineffective but also time consuming [11]-[17].

In this paper, a systematic way of selecting the weighting functions in H∞ optimal control is proposed. The selection of adequate weighting function is formulated as an optimization problem based on H∞ loop-shaping approach and solved using Population Based Incremental Learning (PBIL). PBIL is a relatively new Evolutionary Algorithm that is simpler and more compact and effective than Genetic Algorithms (GAs) [19]-[21]. Simulation results shown that PBIL-based controller gives a slightly better performance than GA-based controller. The conventional controller gives the worst performance.

K A. Folly is with the Electrical Engineering Department, University of Cape Town, South Africa (corresponding author to provide phone: 27216504490; fax: 27216503465; e-mail: Komla.Folly@ uct.ac.za).

P. Munawa, was the Electrical Engineering Department, University of Cape Town, South Africa (e-mail: MNWPRO001@myuct.ac.za).

This work is based on the research supported in part by the National Research Foundation of South Africa (UID: 83977 and UID: 71967).

II. Overview of H∞ Optimal Control

A. Mixed Sensitivity Approach

H_∞ optimal control is a powerful tool for control design, since it explicitly takes robustness into account. It is a frequency domain synthesis technique that seeks to minimize the worst case disturbance and does offer a possibility to incorporate directly into the design practical requirements such as bandwidth limitation, robustness and disturbance attenuation [1]-[8].

The H_∞ design method accounts for the model uncertainties at the design stage using uncertainty representation methods such as numerator-denominator uncertainty representation or the standard additive/multiplicative uncertainty. The common H_∞ design formulation involves the standard mixed sensitivity weighting strategy where frequency dependent functions are used to shape various sensitivity functions [9]-[10]. Until now, the weighting functions are generally selected based on trial and error. This approach is not only ineffective but also time consuming [11]-[17]. The main difficulty is how to systematically select the weighing functions to achieve the desired performance and robustness requirements. This issue will be discussed in section 5.

The standard H_∞ optimal control employs the mixed sensitivity approach where the peak value minimization of the frequency response functions such as the sensitivity S and complementary sensitivity T functions is achieved [6]. Both S and T are associated with the plant disturbance attenuation at low frequencies and noise attenuation at high frequencies respectively [6] – [7].

The standard H_∞ approach thus offers the controller that is able to account for model uncertainties and performance specifications. The model uncertainty representation in the standard H_∞ approach is either additive or multiplicative with associated control sensitivity $R=KS$ and complementary sensitivity functions $T=1-S$, respectively. Note that the design requirements for S and T are contradictory [5]-[10].

That is., making S smaller will result in T being large and vice-versa.

In general, a trade-off is needed between S and T such that S is minimized at low frequencies, while T is minimized at high frequencies where uncertainties and noise are of concern [9]-[10]. This is achieved by using the weighting functions W_p, W_a and W_m to penalize the various sensitivity functions. It is suggested in the literature that a high gain low-pass filter (weighting function) W_p be used to penalize S, whereas a high-pass filter W_m is used to penalize T [6]-[17]. For the weighting function W_a (associated with the additive uncertainty representation) a compromise should be found such that the control input is minimized without a deterioration of the performance [10]. It is generally selected as a high pass filter [6]-[10].

The standard H∞ controller is thus found by solving equation (1).

$$\underset{K}{Min}\left\|\begin{matrix} W_p(j\omega)S(j\omega) \\ W_a(j\omega)R(j\omega) \\ W_m(j\omega)T(j\omega) \end{matrix}\right\|_\infty \tag{1}$$

Where, $\|.\|_\infty$ denotes the maximum magnitude of the vector over all frequencies.

One drawback that is associated with standard H∞ is that of pole-zero cancellation. This limitation prevents the H∞ controller from increasing the damping of lightly damped poles which may compromise the system's robustness and performance. This issue has been addressed in [7]-[10].

A simpler method than the mixed sensitivity approach H∞ optimal control is to use the loop-shaping approach [6], [11], [17] as described below.

B. Loop-shaping Approach

The loop-shaping approach is based on H∞ robust stabilization combined with classical loop shaping as proposed in [6]. It is essentially a two-stage design process in which the selection of the weighting function is simplified. The first step in loop-shaping approach is to augment the open-loop plant by pre- and post-compensators to give a desired shape to the singular values of the open-loop frequency response. In the second step, the augmented open-loop plant is robustly stabilized with respect to the general class of coprime factor uncertainty using H∞ optimization. An important advantage is that no problem-dependent uncertainty modeling or weighting function selection is required in the second step. In addition, unlike the mixed sensitivity approach, the loop-shaping approach does not require γ–iteration for its solution. Explicit formulae for the corresponding controllers are available.

One of the limitations with additive and/or multiplicative uncertainty representations (or similar representations) is that the plant and the perturbed plants are restricted to have either the same number of unstable poles or the same number of unstable zeros. To overcome this, the numerator-denominator uncertainty representation (also known as Normalized co-prime Factorization) representation) advocated in [6], [10], [14], [17] can be used. This uncertainty representation is much more general than additive and/or multiplicative uncertainty representations.

Let the nominal be represented as:

$$G = D^{-1}N \tag{2}$$

where, N is the numerator and D is the denominator

The perturbed plant can be written as:

$$G_p = \{\ [D_0 + \Delta D]^{-1}[N_0 + \Delta N] : \left\|\begin{bmatrix}\Delta_N & \Delta_D\end{bmatrix}\right\|_\infty < \varepsilon\ \} \tag{3}$$

where, Δ_D and Δ_N are stable unknown transfer functions representing the uncertainties in the denominator and the numerator of the nominal plant, respectively. Also, $\varepsilon>0$ is the stability margin.

To maximize the stability margin is the problem of robust stabilization of normalized coprime factor plant descriptions. For the perturbed system given in (3), the system is robust, if and only if the nominal feedback system (assuming negative feedback) is stable and

$$\gamma_K = \left\|\begin{bmatrix}K \\ I\end{bmatrix}(I+GK)^{-1}D^{-1}\right\|_\infty \le \frac{1}{\varepsilon} \tag{4}$$

where, γ_K is the infinite norm for the closed-loop augmented plant, and $(1+GK)^{-1}$ is the sensitivity function.

The lowest achievable value of γ_K and the corresponding maximum stability margin ε are given as:

$$\gamma_{\min} = \varepsilon_{\max}^{-1} = \left\{1 - \left\|N \quad D\right\|_H^2\right\}^{-\frac{1}{2}} = (1+\rho(XZ))^{\frac{1}{2}} \tag{5}$$

where, $\|.\|_H$ denotes Hankel norm and ρ denotes the spectral radius (maximum eigenvalue).

The maximum eigenvalue can be computed from the product XZ whereas X and Z are unique positive definite solutions that can be computed from the following algebraic Riccati equations (6) and (7):

$$(A - BS^{-1}D^TC)^T X + X(A - BS^{-1}D^TC) - XBS^{-1}X + C^TR^{-1}C = 0 \tag{6}$$

$$(A - BS^{-1}D^TC)^T Z + Z(A - BS^{-1}D^TC) - ZC^TR^{-1}CZ + BS^{-1}B^T = 0 \tag{7}$$

where, A,B,C and D are the state matrices of the shaped plant Gs and $R= I+DD^T$ and $S=I+D^TD$.

For the loop shaping approach in this paper, let G_s be the augmented open-loop plant such that $W_1$and W_2 are the pre-compensator and post-compensator, respectively. Then G_s is given as:

$$G_s = W_2GW_1 \tag{8}$$

The function of these compensators is to shape the open loop to meet performance and robustness requirements [6].
This process ensures that the open loop gain is high at low frequencies and roll-off at high frequencies.
Fig. 1 shows the shaped plant and the overall robust controller $K(s) = W_1(s)K_\infty W_2(s)$, which consists of the H_∞ controller (K_∞) and the compensators as shown in the dashed border. K_∞ is synthesized by solving the robust stabilization problem as described above.

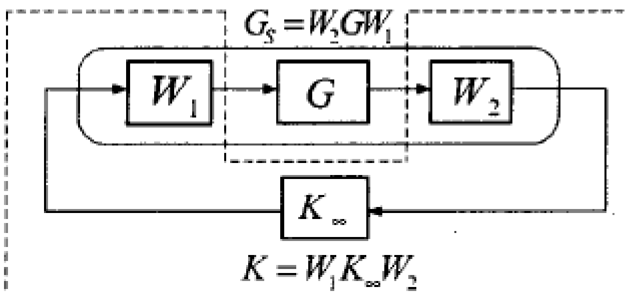

Figure 1. Blocks for the Shaped planted and Controller

III. POWER SYSTEM MODEL

The power system considered is single machine infinite bus (SMIB). The generator is connected to the infinite bus through a double transmission line. The non-linear differential equations of the system are linearized around the nominal operating condition to form a set of linear equations [1]. The generator is modeled using a 6^{th} order machine model, whereas the AVR was represented by a simple exciter of first order differential equation [2], [4].

The system is represented by a set of linear equations as follows [3]:

$$\begin{cases} \dfrac{dx}{dt} = Ax + Bu + w \\ y = Cx + Du \end{cases} \tag{9}$$

where:
A is the system state matrix
B is the system input matrix
C is the system output matrix
D is the feed forward matrix
x is the vector of the system states
u is the vector of the system inputs
w is the disturbance
y is the vector of the system outputs

To design the controller several operating conditions are considered. Selected operating conditions are shown in Table 1. The parameters P_e and X_e in Table 1 are included in matrix A. Matrix B includes the control parameters such as AVR and PSS gains and time constants and are not shown in Table 1. The matrix C contains the output signal, in this case the rotor speed. Matrix D is set to zero.

TABLE I
SELECTED OPERATING CONDITIONS WITH OPEN-LOOP EIGENVALUES

Case	Active Power Pe (p.u.)	Line Reactance Xe (p.u)	Eigenvalues (ζ%)
1	0.300	0.5000	-0.52 ± 4.69i (11.02)
2	0.800	0.4000	-0.03 ± 3.02i (0.99)
3	1.000	1.0000	-0.07 ± 4.23i (1.65)
4	1.250	0.6000	-0.02 ± 4.07i (0.49)

IV. OVERVIEW OF PBIL

PBIL is a technique that combines aspects of Genetic Algorithms and simple competitive learning derived from Artificial Neural Networks. PBIL has the following features [18]-[20]:

- It has no crossover and fitness proportional operators.
- It works with probability vector (number in range 0-1). This probability vector controls the random bitstrings generated by PBIL and is used to create other individuals through learning.
- In PBIL, there is no need to store all solutions in the population. Only two solutions are stored: the current best solution and the solution being evaluated.

The three main operators of PBIL used in this paper are: probability vector (PV), Learning rate (LR) and mutation. Unlike the mechanisms inherent to GAs, where operations defined on the population, in BPIL, the operations take place directly on the probability vector. During the search the values in the probability vector are updated to represent those in high evaluation vectors. The probability vector also guides the search, which produces the next sample point from which learning take place. The learning rate determines the speed at which the probability vector is shifted to resemble the best solution vector. In other words, a higher learning rate would ensure a faster convergence to an optimal solution; however, the whole function space will not be search. This could result in premature convergence. The role of the mutation is to maintain the diversity in the trial solutions.
The individuals are evaluated according to the objective function. The "best" individual is used to update the probability vector so as to produce solutions similar to the current best individuals. Initially, the values of the probability vector are set to 0.5 to ensure that the probability of generating 0 or 1 is equal. As the search progresses, the values in the probability vector are moved away from 0.5, towards either 0.0 or 1.0.

It has been shown that PBIL outperforms standard GAs approaches on a variety of optimization problems including commonly used benchmark problems [19].

The probability update rule is similar to the weight update rule in a competitive learning of ANN as given in (10). The following probability update rule based on the competitive learning is used:

$$PV(i) = PV(i) \times (1.0 - LR) + (LR \times V(i)) \qquad (10)$$

where,
PV(i): the probability of generating 1 in bit position i.
V(i): the i-th position in the solution vector towards which the probability vector is moved.

The learning rate has a greater effect on PBIL as compared to the standard competitive learning. This is because the probability vector is used to generate future sample solutions. Like in competitive learning, the learning rate affects the speed at which the probability vector is shifted to resemble the best solution vector. It also affects the portion of the search space that will be explored [20].

V. Selections of W_1 and W_2 using PBIL

As stated previously, until now, the weighting functions are generally selected based on trial and error. This approach is not only is ineffective but also time consuming. The main difficulty is how to systematically select the weighing functions to achieve the desired performance and robustness requirements. The selection of adequate weighting function is formulated as an optimization problem and solved using Population Based Incremental Learning (PBIL).

For simplicity, W_2 is set to 1. So the only weighting function to be selected is W_1. This weighting function should ensure that the sensitivity function (S) is minimized at low frequencies and the complimentary sensitivity function (T) is minimized at high frequencies. It is assumed that the weighting function is of first order as given below:

$$W_1 = c\frac{s + 1/T}{s + \alpha T} \qquad (11)$$

where α is a control parameter, c is a constant gain and T a time constant.

In general trial and error approach is used to find the suitable weight. However, there is no guarantee that the values obtained using trial and error method are the best values. Therefore, PBIL is used in this paper to get the best values. In total three parameters α, c and T are to be found using PBIL.

To use PBIL, an objective function should be selected. Since the weighting functions will affect the performance of the controller in providing the damping necessary for stabilizing the system's oscillations, the objective function was defined as the maximum of the minimum damping ratio of the closed-loop poles of the system over all the operating conditions considered in table 1. This objective function was the basis upon which selection of suitable weighting function was based, namely,

$$Obj = max(min(damp)) \qquad (12)$$

where, damp means damping ratio which is given by $\zeta_{ij} = \frac{-\sigma_{i,j}}{\sqrt{\sigma_{i,j}^2 + \omega_{i,j}^2}}$ where the number of eigenvalues i=1,2, …n and the number of operating conditions is j =1,2,…, m. The real and imaginary parts of the i-th eigenvalue in the j-th operating condition is $\sigma_{i,j}$ and $\omega_{i,j}$, respectively.

VI. Simulation Results

A. Values of Parameters α, c and T

Using trial and error approach, the following values 5.14, 49.87 and 9.88 were found for the three parameters α, c and T, respectively. Ideally several values should be tried to obtain the best values, which is time consuming. This has not been pursued here.

The parameters of the PBIL are given as:

Population:50
Generation:100
Learning rate: 0.1
Forgetting factor: 0.001

Previous works show that a learning rate of 0.1 is most suitable [18]-[21].

For comparison purposes, Genetic Algorithms (GAs) is also used for the selection of the weighting functions. For more information on GAs, see [22]-[23]

The parameters used for GAs are as follows:

Population:100
Generation:100
Crossover: Arithmetic
Selection: Normalized geometric distribution
Mutation: non-uniform

It should be mentioned that ranking selection is an alternative method whose purpose is to prevent too-quick convergence. The individuals in the population are ranked according to fitness, and the expected value of each individual depends on its rank rather than absolute fitness. Ranking avoids giving the largest share of offspring to a small group of highly fit individuals, and thus reduces the selection pressure when fitness variance is high.

Since the size of population has a significant effect on the performance of GAs, a larger population was used. In terms of function evaluations to reach their respective solutions, GAs has 10000 function evaluations over one independent run, whereas, PBIL has 5000 function evaluations.

Table 2, shows the values obtained when PBIL and GAs are used.

TABLE II
PARAMETERS OF THE WEIGHTING FUNCTION W_1

Algorithm	c	α	T
PBIL	19.00	0.37	82.79
GAs	17.92	0.39	83.76

The stopping criterion adopted during the tests is the maximum generation, which is 100 generations for each optimization technique.

B. *Eigenvalue Analysis*

Table 3 shows the eigenvalues for the closed-loop system with the conventional PSS, the PBIL-PSS and the GA-PSS. It can be seen that the PBIL gives the best damping in all the operating conditions considered.

TABLE III
CLOSED-LOOP EIGENVALUES

Case	PBIL	GAs	CPSS
1	-1.59 ± 4.90i (0.31)	-1.50 ± 4.91i (0.29)	-1.17 ± 4.91i (0.23)
2	-3.38 ± 6.35i (0.47)	-3.16 ± 6.39i (0.44)	-1.16 ± 3.56i (0.31)
3	-2.30 ± 5.78i (0.37)	-2.16 ± 5.79i (0.35)	-1.67 ± 5.80i (0.28)
4	-2.35 ± 3.99i (0.50)	-2.15 ± 4.04i (0.47)	-1.48 ± 4.15i (0.49)

C. *Time domain Simulation*

Figs. 2-5 show the rotor speed responses after a step disturbance is applied to V_{ref}. It can be seen that the performances of the PBIL is slightly better than that of GAs. The conventional controller exhibits the worst performance in terms of overshoots and settling time.

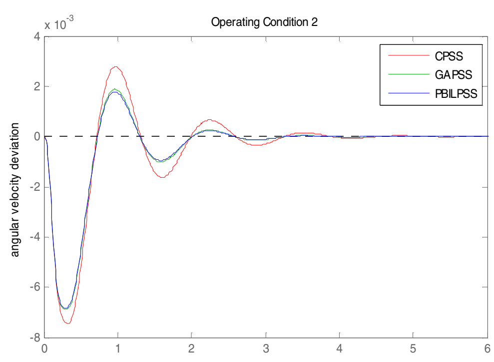

Figure 2: Rotor speed responses (Operating Condition 1)

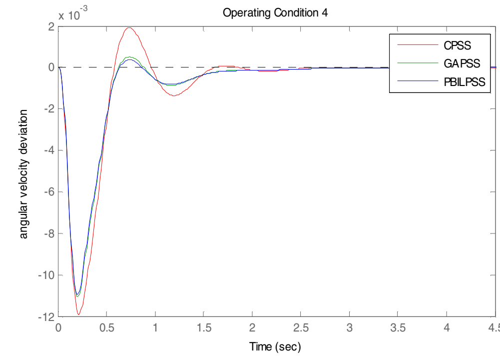

Figure 3: Rotor speed responses (Operating Condition 2)

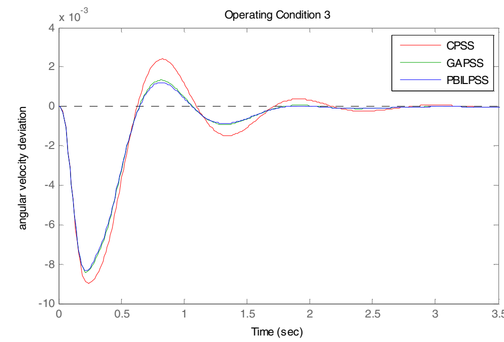

Figure 4: Rotor speed responses (Operating Condition 3)

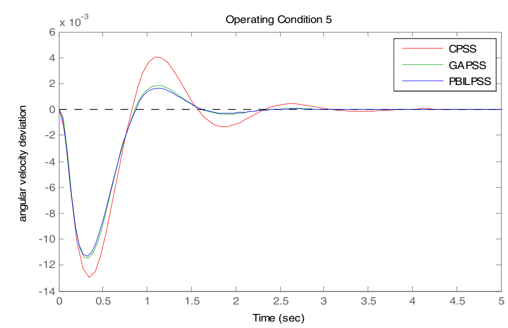

Figure 5: Rotor speed responses (Operating Condition 4)

VII. CONCLUSION

This paper investigated the issue of selecting suitable weight selection in H_∞ Loop shaping controller design. The selection of adequate weighting function is formulated as an optimization problem and solved using PBIL and GAs. The Eigenvalue results show that PBIL gives slightly a better performance in terms of damping ratio compared to GAs and the conventional controller. Time domain simulations also confirm these results.

REFERENCES

[1] P. Kundur, "Synchronous Machine Representation," in *Power System Stability and Control*, 1st ed., L.Grigsby, Ed. New York, USA: McGraw-Hill Education, 1994, pp. 175-176..

[2] L Chen, "A Novel Method for Power System Stabilizer Design," University of Cape Town, Cape Town, Ph.D Thesis 2003.

[3] P. Kundur, *Power System Stability and Control*, McGraw-Hill, Inc.1994

[4] G. Rogers, *Power System Oscillations*. Boston: Kluwer Academic, 1999

[5] Q.Zhao and J. Jiang, "Robust Controller Design for Generator Excitation Systems," in *IEEE Transactions on Energy Conversion, Vol 10, No 2,* June 1995

[6] S. Skogestad and I. Postlethwaite, "*Multivariable feedback Control: analysis and design.* John Wiley & Sons, Ltd, 2005.

[7] K. Mocwane, "Investigation into H-infinity Power System Controller Design with Pole Placement Constraints, MSc thesis, 2010.

[8] KA Folly, N. Yorino, H. Sasaki, "Design of H-infinity-PSS using numerator-denominator uncertainty representation", *IEEE Trans. On Energy Conversion*, Vol. 12, No.1, 1997, pp. 45-50

[9] KA Folly, N. Yorino, H. Sasaki, "Improving the robustness of H-infinity PSSs using the polynomial approach", *IEEE Trans. On Power Systems,* Vol. 13, No.4, 1998, pp. 1359-1364.

[10] KA Folly, "Robust controller design for small-signal stability enhancement of power systems", *Proc. Of the IEEE Afrion,* 2004.

[11] P. Munawa, A systematic procedure for selecting weighting functions in H-infinity controller design'. Final year project, University of Cape Town 2012.

[12] P. Lundstron, "Weight selection for H-infinity and mu-control methods: Insights and examples from process control", [on line]: http://www.nt.ntnu.no/users/skoge/publications/1991/Weight_Selection/Weight_Selection.pdf

[13] I Gunasinghe and A Lanzon, "A Systematic Weight Synthesis Procedure for Perfomance Optimisation in H infinity Loop Shaping," in *Proceedings of the European Control Conference 2007*, Kros Greece, July 2-5 2007, pp. 223 - 228.

[14] I Ngamroo S Dechanupaprittha, "Enhancing performance through multivariable weighting function design in H∞ loop-shaping: with application to a motion system" *2013 American Control Conference* (ACC), pp. 6051-6056, Washington, DC, USA .

[15] H.I Ali, S.B. M. Noor, M. H. Marhaban, and S.M. Bashi, "Design of –inf. Controller with tuning weights using particle swarm optimization method", *IAENG International Journal of Computer Science*, 38:2, IJCS_38_2_02, 2011

[16] Asgharian, R, Tavakoli, S.A., "A systematic approach to performance weights selection in design of robust H-infinity PSS using genetic algorithms", *IEEE Trans. On Energy Conv*., Vol. 11, No,1, 1996, pp. 111-117.

[17] Zhou, K. Doyle, J, *Essentials of Robust Control, Prentice Hall*, 1998.

[18] KA Folly and GK Venayagamoorthy, "A real time implementation of a PBIL based stabilizing controller for synchronous generator", *Proc. Of Ind. Appl. Society, Annual Meeting*, 2009.

[19] T. Mulumba, K. A. Folly, "Power System Stabilizer Design: Comparison Analysis between Differential Evolution and Population Based Incremental Learning", 20th Southern African Universities' Power Engineering Conference (SAUPEC 2011), July, 2011, Cape Town.

[20] S. Baluja "Population based Incremental Learning: A method for integrating Genetic Search Based Function Optimization" Technical Report CMU-CS-49-163, June 2, 1994.

[21] K. A. Folly, "Design of Power System Stabilizer: A Comparison Between Genetic Algorithms (GAs) and Population-Based Incremental Learning (PBIL)". ISBN 1-4244-0493-2, 06GM1312, PES, General Meeting 2006. Jun 18-22, Montreal, Quebec, Canada

[22] Goldberg, D,E, *Genetic algorithms in search, optimization, and machine learning*, 1989Addison-Wesley.

[23] Mitchell, M., *An introduction to genetic algorithms*, MIT Press, 1999.