Intelligent Predictive Control Methods for Synchronous Power System

Muhammad S. Yousuf
Electrical Engineering Department
King Fahd University of
Petroleum & Minerals
Dhahran 31261, Saudi Arabia
Email: syousuf@kfupm.edu.sa

S.Z. Rizvi
Electrical Engineering Department
King Fahd University of
Petroleum & Minerals
Dhahran 31261, Saudi Arabia
Email: srizvi@kfupm.edu.sa

Hussain N. Al-Duwaish
Electrical Engineering Department
King Fahd University of
Petroleum & Minerals
Dhahran 31261, Saudi Arabia
Email: hduwaish@kfupm.edu.sa

***Abstract*—In this paper, an intelligent Model Predictive Controller (MPC) for a Synchronous Power Machine on Infinite Bus (SMIB) is proposed. Owing to the nonlinear and multivariable nature of the SMIB system, calculating optimal control signals can be difficult. To solve this problem, a novel scheme of predictive controller in tandem with heuristic optimization algorithms is proposed. Numerical simulations are carried out and performance of the controller under different conditions and in combination with different optimizers is analysed in detail. Comparison is made with the performance of existing SMIB controllers present in the literature and improvements are observed.**

I. Introduction

The optimal and efficient usage of a Single synchronous Machine on Infinite Bus (SMIB) has been one of the most important problems for power system designers. Thus, various methods have been developed in order to control this system as efficiently as possible, preferably from a perturbed or unstable state, to a stable desired set-point.

The SMIB system is complex and highly nonlinear. Hence, most approaches toward controller design for this system involve some kind of linearization [1]. Complex nonlinear transformations are also used in order to reduce the order of the system as in [2], [3] and [4].

One of the earliest method involving classical control is given by Demello [5] which concerned more with the stabilization of the system than with the control of the system in case of perturbations. Optimal control theory for stabilizing SMIB power systems was developed by Anderson [6] as well as by Yu [7]. These optimal controllers were linear. Adaptive control techniques have also been proposed for SMIB, most of which involve linearization or model approximation. Pahalawaththa et al. [8] proposed an adaptive multi-input multi-output (MIMO) self-tuning Power System Stabilizer (PSS). Although the actual system is complex and nonlinear, the system was approximately identified using recursive least mean square (LMS). This reduced-order-model was then used to control the system. Tabu-search based robust PSS for single machine as well as multi-machine on infinite bus power systems was proposed by Abdel-Magid et al. [9]. Matthews et al. developed a Variable Structure Controller (VSC) for SMIB [10]. The paper presented linear VSC controller using nonlinear transformation which showed good performance in bringing the system from a perturbed state to the equilibrium state. However this control technique was slow and inaccurate resulting in large overshoots in the mechanical power generated. Moreover, the control signals are not used effectively. These issues arise due to the proposed design procedure of the sliding VSC as it involves transforming the state-space system into a Luenberg canonical form and then constructing a suitable sliding surface. This procedure, especially the first step, is complicated and involves many manipulations while also sacrificing the precision of control. Al-Musabi [11] proposed a newly designed VSC for SMIB system utilizing iterative heuristic optimization techniques like Genetic Algorithms (GA) and Particle Swarm Optimization (PSO) to provide a simpler, more systematic method with no need for complex approximations. This enabled direct application of the VSC design to the nonlinear model without undergoing bothersome transformations. It was successfully applied to the model given in [10] and showed significant improvements compared to previous work on this subject. The overshoots and response time of the system improved. However, this technique suffered from the same problems as of Matthews and Cao, in which the control signal was not able to reach to its maximum allowable values and the values of the state variables of the system drastically shifted from their equilibrium values.

A. Paper contribution and organization

In this paper, intelligent Model Predictive Control (MPC) schemes are proposed. The proposed techniques are directly applied to the nonlinear SMIB model. The primary objective is to drive the states of the complex and nonlinear SMIB system from a perturbed state to a desired set-point without the need of any approximation, linearizations, or model reduction, and while taking into consideration the constraints on the states and inputs. Evolutionary Programming (EP), GA and PSO are used to find the best optimum control signals to drive the SMIB plant from one operating point to other. The combination of Model Predictive Control (MPC) and evolutionary techniques will give obvious advantages with regards to optimal control and constraints handling.

Notations in this paper are used in the following manner. Variables in lower case represent scalar quantities while lower case bold variables represent vector quantities. Upper case bold variables are used to represent matrices. The only exceptions to this convention are in the choice of conventional *J* for the cost function, and where notations are defined otherwise, as in the plant model.

II. Model Predictive Control

The Model Predictive Control (MPC) is one of the most well-known and successful control methodologies that can incorporate and handle nonlinearities and constraints in a structured way for any process model [12]. In these techniques, an explicit dynamic model of a plant is used to predict the effect of future actions of the manipulated variables on the output, thus providing the name *Model Predictive Control*. The future moves of the manipulated variables are determined by minimizing the predicted error subject to necessary constraints. The optimization is repeated at each sampling time based on updated information i.e. measurements from the plant. Good literature reviews of MPC can be found in [13], [14], [15], [16] and the references therein.

A. The Intelligent Predictive Controller iMPC

The intelligent MPC concept is explained as follows. In a discrete-time space with a sampling period *T*, the input and output of every system will be denoted by $\mathbf{u}[k] := \mathbf{u}(kT)$ and $\mathbf{y}[k] := \mathbf{y}(kT)$ respectively, where k is an integer from $-\infty$ to $+\infty$. Any nonlinear lumped system in this space can be described by the following sets of equations:

$$\mathbf{x}(k+1) = h(\mathbf{x}(k), \mathbf{u}(k), k), \quad (1)$$

$$\mathbf{y}(k) = f(\mathbf{x}(k), \mathbf{u}(k), k), \quad (2)$$

where h and f are nonlinear functions. Variables $\mathbf{u}(k) \in \Re^{n_u}$ denote control efforts, $\mathbf{x}(k) \in \Re^{n_x}$ denote system states and $\mathbf{y}(k) \in \Re^{n_y}$ denote process output at discrete-time instant k.

The future outputs of the system are predicted for a finite number of future time-samples called Prediction Horizon H_p. Considering a system having multiple inputs and outputs (MIMO), these predicted outputs, denoted by $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}(k+1) \cdots \hat{\mathbf{y}}(k+H_p)]^T$ are dependent on future control moves $\mathbf{U} = [\mathbf{u}(k) \cdots \mathbf{u}(k+H_p-1)]^T$. These future control moves need to be determined so as to minimize a cost function J based on predicted error. The objective is to keep the process as closed as possible to the set of reference trajectories $\mathbf{W} = [\mathbf{w}(k+1) \cdots \mathbf{w}(k+H_p)]^T$ for all outputs. Cost function J is given as

$$J = \sum_{i=1}^{H_p} \mathbf{e}(k+i)^T \mathbf{Q} \mathbf{e}(k+i) + \sum_{i=1}^{H_c} \Delta\mathbf{u}(k+i)^T \mathbf{R} \Delta\mathbf{u}(k+i) + \sum_{i=1}^{H_p} \mathbf{u}(k+i)^T \mathbf{S} \mathbf{u}(k+i), \quad (3)$$

where H_c is the control horizon and ***e(k)*** is the error between the desired output and the predicted output.

$$\mathbf{e}(k) = \mathbf{w}(k) - \hat{\mathbf{y}}(k). \quad (4)$$

Q, **R** and **S** are the weighting matrices for the error **e**, control effort **u** and change in control effort $\Delta\mathbf{u}$ respectively. Their values are assigned according to the process model and constraints. Optimization of J results in an optimal control sequence, $\mathbf{u}(k) \in \Re^{H_p}$. The first control signal in the sequence is applied for process control, system states are updated and the routine is repeated at the next sample $k+1$ using the latest measured information. This is called the *receding horizon* principle [17].

The algorithm can be summarized to generally have the following three steps.

1) Set time-sample $k = 0$.
2) Set $\mathbf{x}(k)$ = initial conditions.
3) $\mathbf{U} = \begin{bmatrix} \mathbf{u}(k) \\ \mathbf{u}(k+1) \\ \vdots \\ \mathbf{u}(k+H_p-1) \end{bmatrix}$ = minimize $J(\mathbf{U})$.
4) Apply $\mathbf{u}(k)$ to the plant.
5) $\mathbf{x}(k+1)$ = $h(\mathbf{x}(k), \mathbf{u}(k), k)$.
6) $k = k+1$.
7) repeat steps 3 - 6.

The minimization of cost J in step 3 is a crucial task and requires strong optimization capabilities. In this work, optimal control-efforts are calculated using different evolutionary techniques discussed below. Their performance is compared and analysed in later sections.

B. Genetic Algorithms

Genetic Algorithms (GA) are exploratory search and optimization algorithms that can solve multi-modal and nonlinear optimization problems. The GA algorithm was first introduced by Holland in [18]. The general idea is to maintain a population of chromosomes that represent possible solutions to a problem at hand. With successive generations, the population evolves into one with better solutions, based on the principles of natural selection. The population undergoes transformation and evolves towards optimal solution using operations that imitate the biological process of mutation and crossover.

To solve the MPC problem at hand, an initial solution of n number of chromosomes is generated. Each chromosome $\mathbf{x}_i$ represents a possible solution in m-dimensional space

$$\mathbf{x}_i = [x_{i1} \cdots x_{im}]. \quad (5)$$

Two candidates are then selected as parents to breed children for the next generation. There exists multiple methods of parent selection in the literature [19]. In the present work, tournament-based selection is used to select parents. Each pair of parents undergoes operation of crossover to breed two new children. Out of the several crossover methods possible, the BLX-α crossover is used in this work. The BLX-α crossover ensures minimum repetition of solutions and provides better exploration. If the child $\mathbf{c}_i$ is represented by

$$\mathbf{c}_i = [c_{i1} \cdots c_{im}], \tag{6}$$

then the j^{th} gene of the given child is produced by generating a random number in the interval $[c_{min} - I\alpha, c_{max} + I\alpha]$, where

$$c_{max} = max[c_j^{parent1}, c_j^{parent2}], \tag{7}$$
$$c_{min} = min[c_j^{parent1}, c_j^{parent2}], \tag{8}$$
$$I = c_{max} - c_{min}. \tag{9}$$

Typical value of α is around 0.5. Once, crossover is applied, children are produced. These children become part of the next generation, unless they represent a solution which is worse than their parents. After crossover, chromosomes undergo random mutation in their genes based on a probability-of-mutation that is usually near 0.1. This way, generations evolve moving in the direction of optimal solutions.

C. Evolutionary Programming

Like GA, Evolutionary Programming (EP) is a heuristic population-based search procedure that incorporates random variation and selection. It has been reported by Fogel to perform well with highly epistatic objective functions, i.e. where the parameters being optimized are highly correlated. The EP algorithm makes sure that a parent having an advantage is not lost without transferring its advantageous gene to the child. It combines old and new generation and uses tournament competition amongst them. This ensures that individuals with good capabilities are not lost by mutation. This feature makes EP robust and efficient to epistatic objective functions and on many problems [20]. The convergence analysis of EP is well established and it has been proven to asymptotically converge to the global optimum. Problem constraints can be easily incorporated in EP as well [21].

In EP, an initial population of n number of m-dimensional probable solutions is generated. Each candidate $\mathbf{x}_i$ is represented in m-dimensional space as

$$\mathbf{x}_i = [x_{i1} \cdots x_{im}], \tag{10}$$

where m is the number of optimizable parameters. Initially, each individual in the population is evaluated using the cost function J in equation 3, and best solution is saved as $\mathbf{x}_{best}$. Mutation is then carried out on the individuals and n offsprings are generated from n parents using the following equation.

$$\mathbf{x}_{n+i} = \mathbf{x}_i + [N(0, \sigma_{i1}^2) \cdots N(0, \sigma_{im}^2)], \tag{11}$$

where σ_{ij} is the standard deviation for the j^{th} gene of the i^{th} individual specifying the range for the offspring produced, and is given by.

$$\sigma_{ij} = \beta \frac{J(\mathbf{x}_i)}{J(\mathbf{x}_{max})}(x_j^{max} - x_j^{min}), \tag{12}$$

where β is the scaling factor and $J(\mathbf{x}_i)$ is the objective function of individual $\mathbf{x}_i$. Best solution is then calculated from amongst $2n$ individuals and $\mathbf{x}_{best}$ is updated in case of an improvement.

A tournament is then arranged, and each individual in the $2n$ combined population is then compared with q opponents selected at random such that $q < 2n - 1$. A weighting factor w_i is assigned to every individual based on the following equations.

$$w_i = \sum_{t=1}^{q} = w_t \tag{13}$$

$$w_t = \begin{cases} 1 & ifU > \frac{J(\mathbf{x}_i)}{J(\mathbf{x}_i)+J(\mathbf{x}_t)} \\ 0 & otherwise \end{cases} \tag{14}$$

where U is a uniform random number over [0,1]. After obtaining the competition weights for all $2n$ individuals, the individuals with highest weights are selected to represent the parents of the next generation. In the proposed MPC strategy, EP is used to find optimal control signals $u_{1,2}(k)$ to steer the states of the synchronous machine towards the reference trajectory.

D. Particle Swarm Optimization

PSO is one of the best known and most widely used optimization methods. It was introduced by Kennedy and Eberhart [22] and is inspired by human or animal social behavior. Compared to other Evolutionary Algorithms (EAs), PSO is more robust and faster. Since PSO can generate a high-quality solution quickly with most stable convergence characteristics, it has been effective in solving problems relevant to a wide variety of scientific fields [23].

The PSO algorithm starts with a swarm of particles $\mathbf{X}(k) \epsilon \Re^{n \times m}$ at iteration $k = 0$, where n denotes the size of the population in which each particle $\mathbf{x}_i(k)$ is represented by an m-dimensional vector

$$\mathbf{x}_i(k) = [x_{i1}(k) \cdots x_{im}(k)], \tag{15}$$

where m represents the number of parameters that need to be optimized.. The particles change their positions by flying around in a multi-dimensional search space until a relatively unchanging position has been encountered. The velocity for the i^{th} particle is represented by an m-dimensional vector

$$\mathbf{v}_i(k) = [v_{i1}(k) \cdots v_{im}(k)]. \tag{16}$$

An inertia weight, w is used to control the impact of the previous velocities on the current velocity. A large initial inertia weight is recommended for global exploration and vice versa. As a particle moves through the search space, it compares its fitness value at the current position to the best fitness value it has ever attained at any time up to the current time. The best position that is associated with the best fitness encountered so far is the individual or local best $\mathbf{x}_j^*(k)$. The global best $\mathbf{x}^{**}(k)$ is the best position among all individual best positions achieved so far.

The j^{th} parameter of every particle is generated within the range of the j^{th} optimized parameter $[x_j^{max}, x_j^{min}]$. For the problem at hand, each particle is evaluated using the objective function in equation 3. As the iterations progress, each particle

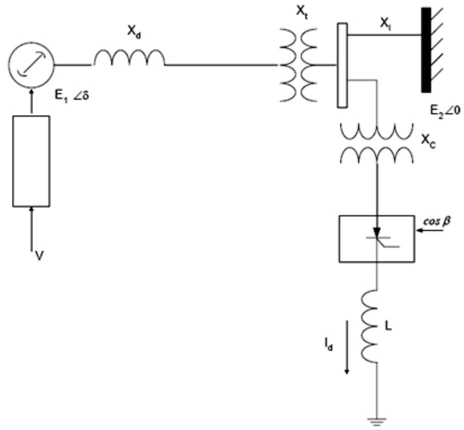

Fig. 1. Diagram of Synchronous Machine on Infinite Bus (SMIB) Power System

is compared with its local best and local best is updated. Inertia weight is updated according to $w = \alpha w$, where α is smaller than but close to 1. Finally velocity and position of every particle is updated. Velocity update of i^{th} particle is given by

$$\mathbf{v}_i(k+1) = w\mathbf{v}_i(k) + c_1 r_{i1}(k)\{\mathbf{x}_i^*(k) - \mathbf{x}_i(k)\} + c_2 r_{i2}(k)\{\mathbf{x}^{**}(k) - \mathbf{x}_i(k)\}], \tag{17}$$

$$\mathbf{x}_i(k+1) = \mathbf{v}_i(k) + \mathbf{x}_i(k). \tag{18}$$

where c_1 and c_2 are cognitive and social parameters and represent orientation of velocity update towards local and global best respectively.

III. The SMIB System

A. Nonlinear Model of the SMIB System

The nonlinear model of the SMIB system taken here is given in [10] and the block diagram is shown in Figure 1.

The dominant dynamics of the nonlinear system can be simplified using the following assumptions:

- The voltage behind the transient reactance of the machine is constant.
- Governor/turbine dynamics are represented by a slow first-order system
- Swing equations are used to describe the mechanical motion of the synchronous machine.

The dynamics of the system are described by the following equations:

$$\dot{\delta} = \omega \tag{19}$$

$$\dot{\omega} = \frac{\omega_B}{2H}[P_m - P_{ac} - KP_{dc}] - D.\omega \tag{20}$$

$$P_{dc} = (\cos(\beta) - R_c I_d) I_d \tag{21}$$

$$\dot{I_d} = \frac{1}{L}(\cos(\beta) - R_c I_d) \tag{22}$$

$$\dot{P_m} = -\alpha P_m + v \tag{23}$$

where δ is the rotor angle of machine in electrical rad relative to the center of mass, ω is the rotor angular velocity in rad/s with respect to synchronous speed, H is inertia constant in sec, D is the damping coefficient in sec^{-1}, P_m is per unit mechanical power, P_{ac} is per unit AC power, P_{dc} is the per unit power stored in the converter. ω_B = 377 rad/s. ω_b = 75.399 rad/s.K = 1, α is the time constant of governor/turbine or mechanical power actuator. v is the corresponding input, I_d is the Direct Current through the converter and R_c is the Commutating resistance per unit.

$$X = X_d + X_t + X_l \tag{24}$$

$$P_{ac} = (E_1 E_2/X)\sin\delta \tag{25}$$

Based on the dynamic model above, the states are defined as follows:

x_1 = δ; x_2 = I_d; x_3 = ω; and x_4 = P_m

And the control inputs are:

u_1 = $\cos(\beta)$ and u_2 = v

The system can be represented in state-space form as follows:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} = \begin{bmatrix} x_3 \\ -k_1 x_2 \\ -k_2 sin(x_1) + k_3 x_2^2 - Dx_3 + k_5 x_4 \\ \alpha x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ k_4 & 0 \\ -k_5 x_2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{26}$$

where,

k_1 = $\frac{R_c}{L}$, k_2 = $\frac{\omega_B E_1 E_2}{2HX}$, k_3 = $\frac{\omega_b R_c}{2HX}$, k_4 = $\frac{1}{L}$, k_5 = $\frac{\omega_B}{2H}$

The DC Converter is rated at 80 MW. The system is 230 kV and the machine rating is 800 MVA. On this rating base, the system parameters are [10]: X = 0.2 pu, R_c = 0.3 pu, L = 0.015 pu, H = 7.0 s, D = 0.5 s^{-1}, and α = -0.1 s^{-1}. This corresponds to k_1 = 20, k_2 = 177.72857, k_3 = 8.078571, k_4 = 66.667, and k_5 = 26.928561.

B. Control Objectives

The primary control objective is to drive the system from a perturbed, possibly unstable state to a desired equilibrium point and to maintain it there.

The controller achieves this by posing the SMIB system as an optimization problem in which the error is predicted beforehand using MPC and is minimized using the intelligent heuristics. The cost function proposed is the following:

$$J = \sum_{i=1}^{H_p} \Delta I_d^2 + \Delta\omega^2 \tag{27}$$

where ΔI_d is the error in the DC current through the converter and $\Delta\omega$ is the error in the rotor angular velocity in rad s^{-1} with respect to the synchronous speed of the rotor.

The control objectives involve these subgoals:

1) The machine must be operated at the rated frequency, i.e. change in frequency, x_3 must be zero at equilibrium.

TABLE I
PARAMETER VALUES FOR OPTIMIZER

Parameters	GA	EP	PSO
Population size	70	150	50
Number of genes or particles	10	10	10
Number of elite chromosomes	4		
Number of generations or iterations	200	200	200
Tournament size for parents selection	15	100	
Crossover type	BLX-α		
α	0.5		
Probability of mutation	0.1		
Initial value of inertia weight			0.9
Final value of inertia weight			0.4
Cognitive parameter c_1			2
Social parameter c_2			2

2) The DC current through the converter, I_d, x_2 must be zero at equilibrium.
3) A specified amount of AC power is required to be delivered to the bus. This defines the desired load angle, γ, of x_1.

The control inputs are constrained as follows for all cases:

$$-0.95 \leq u_1 \leq 0.985 \tag{28}$$

$$|u_2| \leq 3.5 \tag{29}$$

Due to the rating of the converter, limit is also imposed on x_2 (I_d) as: $0 \leq x_2 \leq 0.1$ pu. And since $x_4 = P_m$, it is required that $x_4 \geq 0$.

IV. SIMULATION RESULTS

The SMIB system described in Section III has been simulated in a benchmark test. The control inputs are constrained according to Equations 28 and 29. The initial conditions are defined as [10]:

$$x_1 = 0.0522, \quad x_2 = 0.1, \quad x_3 = 0.1,$$
$$x_4 = 6.6\sin(x_1(0)) = 0.3444$$

Here, the initial states x_2 (I_d) and x_3 (ω) are perturbed from equilibrium and the control objective is to converge them to 0 using the inputs $u_1 = \cos(\beta)$ and $u_2 = v$.

It is observed that all of the proposed iMPC techniques succeed in controlling the perturbed system to equilibrium quickly, as seen in Figures 2 to 5. However, there are marked differences in the dynamic responses among the various heuristics used. It is observed the PSO gives the best responses by far, followed closely by GA. EP delivers the results, but with fluctuations and delays. A look at Table I shows that PSO delivers best results with the smallest initial population size. While GA delivers comparable results, it requires slightly larger population to achieve these results. A further reduction in GA population size adversely affects the performance of the controller. The EP algorithm requires the largest initial population. The dynamic response of the controlled outputs, I_d and ω is seen in Figure 3 and Figure 4 respectively. The convertor current, I_d, takes only 0.03 seconds to reach the required equilibrium state of 0 for PSO. More importantly, the change in frequency also converges end reaches the required equilibrium state after 0.1s. Practically, this means that the

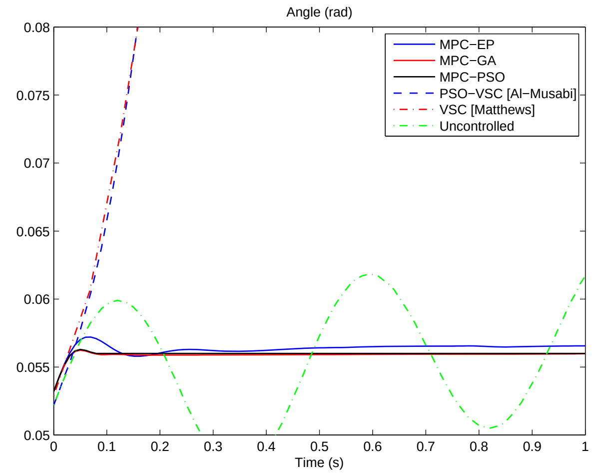

Fig. 2. Angle (rad) for Perturbed System

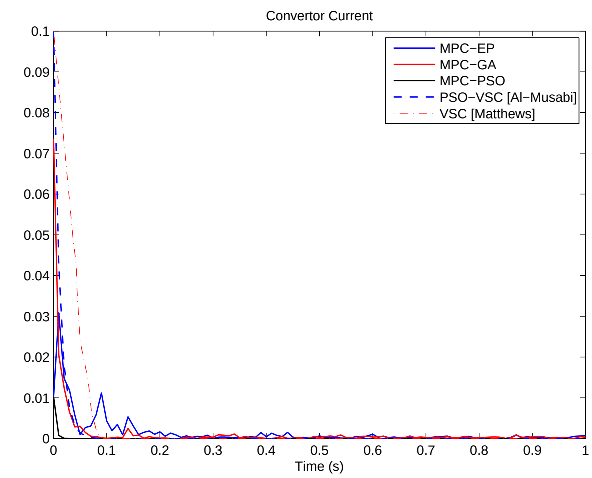

Fig. 3. Convertor Current for Perturbed System

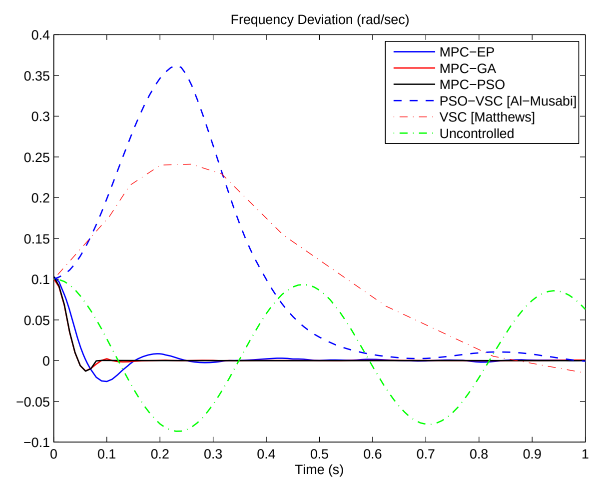

Fig. 4. Frequency Deviation (rad/s) for Perturbed System

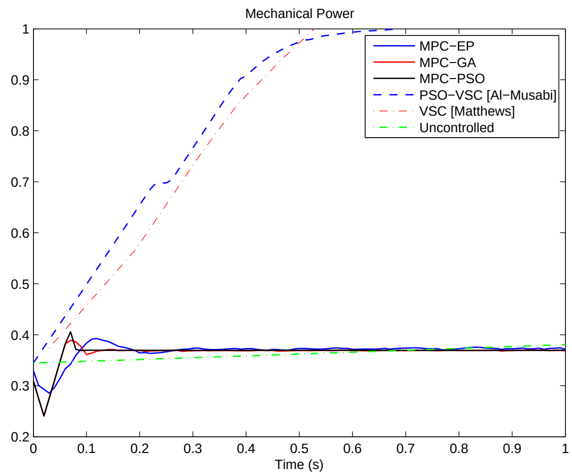

Fig. 5. Mechanical Power for Perturbed System

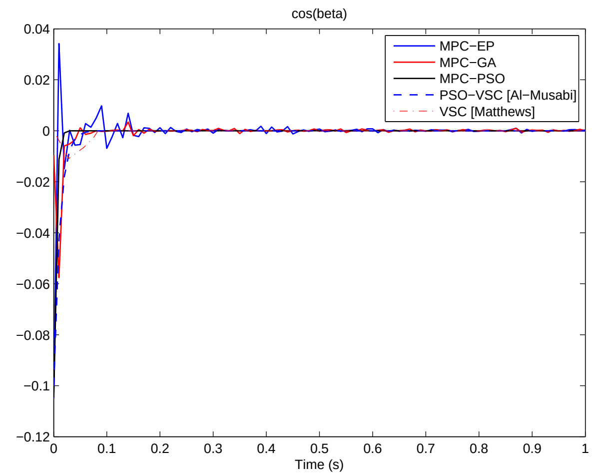

Fig. 6. Control Input 1, cos(β) for Perturbed System

system frequency is brought to 60Hz after starting from an error of 0.1 p.u. The other states of the system, $x_1 = \delta$ and $x_4 = P_m$ settle at slightly different equilibrium points from the initial values after the perturbed system is brought to equilibrium.

The Figures 2 to 5 also show the comparison of these results with Al-Musabi's [11] and Matthew's [10] work. The proposed controller excels by bringing the system to the equilibrium states considerably quicker and keeping the deviation in the angle and mechanical power of the system minimal. This is especially true for the MPC-PSO example. The frequency deviation reached a maximum value of only -0.013 p.u. while for the previous work, the deviation reached a maximum of 0.35p.u. at 0.25s. The frequency settling time thus shows a 10-fold improvement. The convertor current, I_d is also observed to reach the required equilibrium state in a shorter duration.

The control effort applied is seen in Figures 6 and 7. For the MPC-PSO example, the first control input, cos(β) is needed for only 0.03s. After that, I_d settles to zero. The second control effort, v is in effect for 0.1s. u_1 and u_2 are also found to be within the constraints imposed by the system in Equations 28 and 29.

It is duly noted that the proposed iMPC controllers do not cause large changes in the angle and mechanical power of the SMIB system during the dynamic behavior. This is quite in contrast with the previous work where huge deviations from the equilibrium states are observed.

Another important point to note is that the control effort, v is in effect for at least 1 sec in previous work. However, during the whole duration, it is unable to reach the maximum allowable control limits defined in Equation 29, attaining a maximum of ±1.5. Using the proposed controller MPC-PSO and MPC-GA techniques, it is noted that the whole range of control input is utilized and the control input does reach the maximum allowable values of ±3.5. This explains the improved results achieved by the proposed controllers, since complete range of possible control efforts is properly explored,

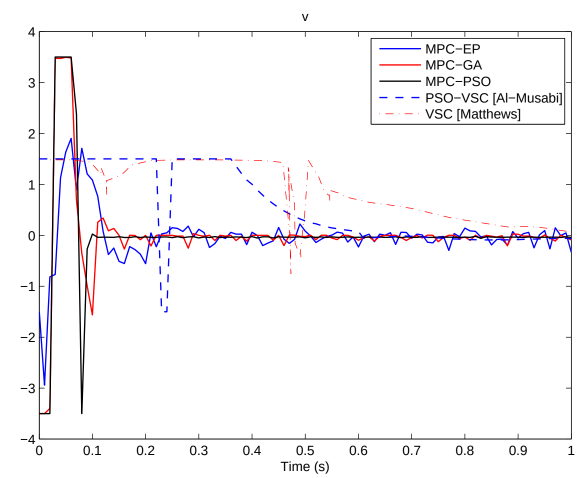

Fig. 7. Control Input 2, v for Perturbed System

thus enabling the system to reach equilibrium quickly.

V. Conclusion

A new controller for SMIB system is presented in this paper. The proposed iMPC techniques demonstrate successful control of the plant without linearization or approximation. This demonstrates the ability of the proposed intelligent controllers to effectively control complex nonlinear plants having industrial significance.

From the comparisons, it is clear that the PSO based iMPC performs the best, followed by GA and EP. Comparisons made with VSC based controllers show that iMPC techniques are much more successful and swift in dealing with the perturbed SMIB system. It is also noted that the proposed techniques enable the system to utilize the full range of control inputs which greatly improves the dynamic response and reduces the deviation in the uncontrolled states of the system.

Acknowledgment

The authors would like to thank King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia for providing support and resources to undertake this research.

References

[1] Young KD and Kwatny HG, "Variable structure servo-mechanism design and applications to overspeed protection control," *Automatica*, vol. 18, pp. 385–400, 1982.
[2] Sommer R, "Control design for multivariable nonlinear time varying systems," *International Journal of Control*, vol. 31, no. 5, pp. 833–891, 1980.
[3] Su R, "On the linear equivalents of nonlinear systems," *System Control Letters*, vol. 2, no. 1, pp. 48–52, 1982.
[4] Hunt LR, Su R, and Meyer G, "Global transformations of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 28, no. 2, pp. 704–707, 1983.
[5] Demello FP and Concordia C, "Concepts of synchronous machine stability as affected by excitation control," *IEEE Transactions on Power Apparatus and Systems*, vol. 88, pp. 316–329, 1969.
[6] Angeline JH, "The control of a synchronous machine using optimal control theory," *Proceedings of the IEEE*, vol. 1, pp. 25–35, 1971.
[7] Yu YN and Moussa HAM, "Optimal stabilization of a multimachine system," *IEEE Transactions on Power Apparatus and Systems*, vol. 91, no. 3, pp. 1174–1182, 1972.
[8] Pahalawaththa NC, Hope GC, and Malik OP, "Multivariable self-tunning power system stabilizer: Simulation and implementation studies," *IEEE Transactions on Energy Conversion*, vol. 6, no. 2, pp. 310–319, 1991.
[9] Abdel-Magid YL, Abido MA, and Mantawy AH, "Robust tuning of power system stabilizers in multimachine power systems," *IEEE Transactions on Power Systems*, vol. 15, no. 2, pp. 735–740, 2000.
[10] Matthews GP, DeCarlo RA, Hawley P, and LeFebvre S, "Toward a feasible variable structure control design for a synchronous machine connected to an infinite bus," *IEEE Transactions on Automatic Control*, vol. 31, no. 12, pp. 1159–1163, 1986.
[11] Al-Musabi NA, Al-Hamouz ZM, and Al-Duwaish HN, "Particle swarm based design of variable structure stabilizer for a nonlinear model of smib system," *The Mediterranean Journal of Measurement and Control*, vol. 2, no. 3, 2006.
[12] Prett DM, Garcia CE, and Morari M, "Model predictive control: Theory and practice - a survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.
[13] Clarke D, Ed., *Advances in Model-Based Predictive Control*. Oxford University Press, 1994.
[14] Richalet J, "Industrial applications of model based predictive control," *Automatica*, vol. 29, no. 5, pp. 1251–1274, 1993.
[15] Roberts PD, "A brief overview of model predictive control," *IEE Two-Day Workshop on Model Predictive Control: Techniques and Applications*, pp. 1–4, 1999.
[16] Qin SJ and Badgwell TA, "An overview of industrial model predictive control technology," 1997. Available online at: http://citeseerx.ist.psu.edu/viewdoc/ summary?doi=10.1.1.52.8909.
[17] M. JM, *Predictive Control with Constraints*. Pearson Education Limited, 2002.
[18] Holland JH, *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, 1975.
[19] Harrera F, Lozano M, and Verdegay JL, "Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis," *Artificial Intelligence Review*, vol. 12, no. 4, pp. 265–319, 1998.
[20] Fogel DB, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. IEEE Press Piscataway, NJ, 1995.
[21] F. DB, "An introduction to simulated evolutionary optimization," *IEEE Trans. on Neural Networks*, vol. 5, no. 1, pp. 3–14, 1994.
[22] Kennedy J and Eberhart RC, "Particle swarm optimization," *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, 1995.
[23] Parsopoulos KE and Vrahatis MN, "On the computation of all global minimizers through particle swarm optimization," *IEEE Transactions on Evolutionary Computing*, vol. 8, no. 3, pp. 211–224, 2004.