Improved Bernstein Optimization Based Nonlinear Model Predictive Control Scheme for Power Systems

Bhagyesh V. Patil * **Jan Maciejowski** ** **K. V. Ling** ***

* *Cambrdige Centre for Advanced Research and Education, 50 Nanyang Ave, Singapore (e-mail: bhagyesh.patil@gmail.com)*
** *Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, United Kingdom (e-mail: jmm@eng.cam.ac.uk)*
*** *School of Electrical and Electronic Engineering, NTU, 50 Nanyang Ave, Singapore (e-mail: ekvling@ntu.edu.sg)*

Abstract: This paper presents a improved Bernstein global optimization algorithm based model predictive control (MPC) scheme for the nonlinear systems. A new improvement in the Bernstein algorithm is the introduction of a box pruning operator, which during a branch-and-bound search, discard portions of the solution search space that do not contain global solution, thereby speeding up the algorithm. The applicability of this MPC scheme is demonstrated with a simulation studies on a nonlinear single machine infinite bus power system over a wide range of operating conditions. The simulation results show improvement in the system damping and settling time compared with the classical power system stabilizer and partial feedback linearization control schemes.

Keywords: Bernstein polynomials, Global optimization, Nonlinear model predictive control, Single machine infinite bus, Synchronous Generators, Excitation control.

1. INTRODUCTION

Model predictive control (MPC) is an established advanced control methodology for multivariable control systems. It is also known as receding horizon control, which generates control actions by optimizing specific performance index over a finite-time moving window within system constraints, and based on a dynamic model of the system to be controlled [Maciejowski (2002), Camacho and Bordons (2004)]. The applications of MPC on a large scale is found in process industries, especially petrochemical plants [Darby and Nikolaou (2012)]. In the last decade, MPC has also gained a good success in the other sectors, such as power systems, robotics and automotive and aerospace industries (see, for instance Qin and Badgwell (2003), and reference therein). A more recent survey on the current trends in MPC can be found in Badgwell and Qin (2014).

In practice, many MPC applications prefer linear models, due to there simplicity and facilitating the use of convex optimization techniques for the online solution of the optimization problems. Such MPC applications are also known as 'linear MPC' scheme (cf. Rao and Rawlings (2000)). However, some applications (like power systems) has a nonlinear behavior, and for such applications, linear MPC scheme may not yield good close-loop performance. Hence, to alleviate problems arising from the system nonlinearities, gain scheduling and switching between multiple-linear models based on the operating region are possible solution approaches reported in the literature [Lawrence and Rugh (1995), Aufderheide and Bequette (2003)]. Another interesting approach followed by many researchers is to use a nonlinear system model. This approach may come with attractive benefits, such as, tighter regulation of system parameters, and the possibility of operating the system (with a good control authority) in different operating regimes. The model predictive control using nonlinear system models, usually called 'nonlinear MPC' (or NMPC), hence has attracted many researchers over the past decade [Martinsen et al. (2004), Findeisen et al. (2007), Hedengren et al. (2014)].

We note that, an NMPC formulation requires the solution of a (usually *nonconvex*) nonlinear optimization problem at each sampling instant. As such, NMPC is a challenging field, and is dependent on a good optimization procedures. Concerning this fact, in the present work we introduce one such (global) optimization procedure for NMPC applications. This procedure is based on the well-known Bernstein form of polynomials [Ratschek and Rokne (1988)], and uses several nice 'geometrical' properties associated with this Bernstein form. Optimization procedures based on this Bernstein form, also called *Bernstein global optimization algorithms*, have shown good promise to solve hard nonconvex optimization problems (see, for instance, Patil et al. (2012b), and references therein). Recently, authors in Patil et al. (2012a) reported some encouraging preliminary findings using one such Bernstein global optimization algorithm for predictive control of nonlinear hybrid systems. Therefore, more investigations using these algorithms seem to be a promising research direction for NMPC applications.

The current scope of the work involve experimentation of the Bernstein global optimization algorithm reported in Patil and Nataraj (2016) to solve a nonlinear optimization problem at each NMPC iteration. Specifically, we use the box consistency feature for the Bernstein algorithm, which aids in pruning (discarding) regions from a solution search space that surely do not contain the global solution. As this feature provide the means to narrow the search region (in our case box) for the optimization problem, we term it as a box pruning (or narrowing) operator. The applicability of the Bernstein algorithm with this box pruning feature (henceforth, referred as the improve Bernstein algorithm) is demonstrated by simulating a predictive control scheme for a classical nonlinear single machine infinite bus (SMIB) power system [Kundur (1994)]. The findings of the NMPC scheme based on a improved Bernstein algorithm are compared with respect to a two well established control schemes, namely, power system stabilizer (PSS) [Kundur (1994)] and partial feedback linearization (PFL) [Mahmud et al. (2014)].

In the rest of the paper, we first introduce a nonlinear MPC formulation (Section 2). Next, we briefly describe the Bernstein form, the box pruning operator followed by the presentation of the improved Bernstein global optimization algorithm (Section 3), and report the simulation studies on a nonlinear SMIB power system (Section 4). Finally, some concluding remarks are given in Section 5.

2. REVIEW: NMPC

In this section, we outline one variant of NMPC reported in the literature (see, for instance, Maciejowski (2002), Mayne and Rawlings (2000)).

We consider a class of discrete-time systems described by the following nonlinear model

$$x_{k+1} = f_k(x_k, u_k) \tag{1}$$

where $x_k \in \mathbb{R}^n$ and $u_k \in \mathbb{R}^m$ are the state of the system and the control input applied at sampling instant k, respectively. The system is subject to the state and input constraints of the following form:

$$x_k^{\min} \leq x_k \leq x_k^{\max} \tag{2}$$

$$u_k^{\min} \leq u_k \leq u_k^{\max} \tag{3}$$

In the present work, we consider the design of an NMPC controller for (1) which brings the system from an arbitrary point back to the equilibrium point (origin), while fulfilling constraints of the form (2)-(3). The general form of NMPC control law can be derived at each sampling instant k, given the initial state x_0, by the solution of the following nonlinear programming (NLP) problem.

$$\min_{u_k} \sum_{k=0}^{N-1} \left(x_k Q x_k^T + u_k R u_k^T \right) \tag{4}$$

$$\text{subject to} \quad (1),\ (2),\ \text{and}\ (3)\ \text{for}\ k = 0, 1, \ldots, N-1 \tag{5}$$

where $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ denote positive definite, symmetric weighting matrices used to penalize state and control movements about the origin, receptively. $N(\geq 1)$ denotes the prediction horizon.

At the outset, the nonlinear model (1) is used for predictions based on the initial state x_0. The predicted control input profile is denoted by $\overline{u}_k$, $k = 0, 1, \ldots, N-1$ Then, assuming that the optimization problem has a feasible solution, an optimizer (in this work, we use improved Bernstein global optimization algorithm) computes an optimal control sequence, based on the NMPC optimization problem formulated in (4), defined as

$$\begin{bmatrix} u_0^* \\ u_1^* \\ \vdots \\ u_{N-1}^* \end{bmatrix}. \tag{6}$$

Only the first step of this optimal control sequence, u_0^* is applied to the system (1) to obtain a new updated state. Then the whole process is repeated, with x_0 obtained from the latest measurements, until the state is steered to the equilibrium point.

3. BERNSTEIN POLYNOMIAL APPROACH FOR GLOBAL OPTIMIZATION

This section briefly introduces some notions about the univariate Bernstein form. A comprehensive background and mathematical treatment for a multivariate case can be found in Patil et al. (2012b).

Let a real bounded and closed interval $\mathbf{x}$ (referred as a box in the multidimensional case) is defined as

$$\mathbf{x} = [\underline{x}, \overline{x}], \quad \underline{x} \leq \overline{x} \text{ and } \underline{x}, \overline{x} \in \mathbb{R}.$$

Without loss of generality, we consider the unit interval case (*i.e.* $\mathbf{x} = [0, 1]$), since any nonempty compact interval can be mapped affinely onto it.

We can write a univariate l-degree polynomial p over the unit interval $\mathbf{x}$ in the form

$$p(x) = \sum_{i=0}^{l} a_i x^i, \quad a_i \in \mathbb{R}\ . \tag{7}$$

Now the polynomial p can be expanded into the Bernstein polynomials of the same degree as below [Ratschek and Rokne (1988)]

$$p\,(x) = \sum_{i=0}^{l} b_i B_i^l\,(x) \tag{8}$$

where $B_i^l(x)$ are the Bernstein basis polynomials and b_i are the Bernstein coefficients:

$$B_i^l(x) = \binom{l}{i} x^i (1-x)^{l-i}\ . \tag{9}$$

$$b_i = \sum_{j=0}^{i} \frac{\binom{i}{j}}{\binom{l}{j}} a_j, \quad i = 0, \ldots, l. \tag{10}$$

Equation (8) is referred as the Bernstein form of (7) and obeys the following range enclosure property [Ratschek and Rokne (1988)]:

$$\overline{p}(\mathbf{x}) \subseteq B(\mathbf{x}) := [\min b_i,\ \max b_i]\ . \tag{11}$$

where $\overline{p}(\mathbf{x})$ denote the range of p on a given interval $\mathbf{x}$.

Remark 1: The above theorem says that the minimum and maximum coefficients of b_i provide lower and upper bounds for the range of p. This forms the Bernstein range enclosure, defined by $B(\mathbf{x})$ in equation (11). Further, this Bernstein range enclosure can successively be sharpened by the continuous domain subdivision procedure (see, for instance Patil et al. (2012b)).

The derivative of a polynomial p with respect to x can be found from the Bernstein coefficients of the original polynomial, using the following relation Ratschek and Rokne (1988).

$$p'(\mathbf{x}) = l\left(\sum_{i=0}^{l-1} b_{i+1}B_i^{l-1}(x) - \sum_{i=0}^{l-1} b_i B_i^{l-1}(x)\right) \tag{12}$$

where $p'(\mathbf{x})$ contains an enclosure of the range of the derivative of p on $\mathbf{x}$.

3.1 Box pruning operator

The box pruning operator is used to discard unwanted regions from the solution search space (in our case from the box $\mathbf{x}$) that surely do not contain the global solution. This pruning is achieved by assessing consistency of left and right bounds ($\underline{x}$ and $\overline{x}$) for the given set of algebraic equations (in our case constraints of the form (5)). Typically, with the help of interval Newton method, leftmost and rightmost 'quasi-zeros' are isolated from the box $\mathbf{x}$ Hansen and Walster (2005). In this work, we shall use the univariate version of the Bernstein Newton contractor given in Patil and Nataraj (2016).

Algorithm box prune: $\mathbf{x}' = \text{box\_prune}((b_g), (b_h), \mathbf{x}, r)$

We below explain the box pruning operator procedure for the equality constraint case. It can also be applied to the inequality constraint $h(\mathbf{x}) \le 0$ by converting it into an equality constraint. Consider an equality constraint polynomial $g(\mathbf{x}) = 0$, and let (b_g) be the Bernstein coefficient array of $g(\mathbf{x})$. Consider any component direction, say the first, with $\mathbf{x}_1 = [a, b]$. Typically, an attempt is made to increase the value of a and decrease the value of b, thus effectively narrowing the width of $\mathbf{x}_1$.

To increase the value of a, first find all those Bernstein coefficients of (b_g) corresponding to $x_1 = a$. The minimum to maximum of these coefficients gives an interval denoted by $\mathbf{g}(a)$. If $0 \notin \mathbf{g}(a)$, then the constraint is infeasible at this endpoint a, and we search starting from a, along $x_1 = [a, b]$ for the first point at which constraint becomes just feasible, that is, we try to find a zero of $g(x)$. Let us denote this zero as a'. Clearly, $g(x)$ is infeasible over $[a, a')$, and so it can discarded to get a contracted interval $[a', b]$. On the other hand, if $0 \in \mathbf{g}(a)$ then we abandon the process to increase a and instead switch over to the other endpoint b and make an attempt to decrease it in the same way as we did to increase a.

To find a zero of $\mathbf{g}$ in $[a, b]$, one iteration of the univariate version of the Bernstein Newton contractor given in Patil and Nataraj (2016) is used. It is as follows

$$\mathbf{N}(\mathbf{x}_1) = a - (\mathbf{g}(a)/\mathbf{g}'_{x_1})$$
$$\mathbf{x}'_1 = \mathbf{x}_1 \cap \mathbf{N}(\mathbf{x}_1)$$

where $\mathbf{g}(a)$ is the minimum to maximum of the Bernstein coefficients array $(b(\mathbf{x}))$ at $x_1 = a$, $\mathbf{g}'_{x_1}$ denotes an interval enclosure for the derivative of $\mathbf{g}$ on $\mathbf{x}_1$, and $\mathbf{x}'_1$ gives a new narrowed interval. A similar process is carried out from the other endpoint b.

We now illustrate the above box pruning operator idea with a help of an simple example given below.

Consider the following equality constraint polynomial

$$g(x) = 3x_1^2 - x_2 = 0$$
$$x_1 \in \mathbb{R},\ x_2 \in \mathbb{Z}$$

with $\mathbf{x}_1 = [0.2, 1]$, and $\mathbf{x}_2 = [0, 1]$.

The Bernstein coefficient array of $g(x)$ (calculated from (10)) is

$$(b_g) = \begin{pmatrix} 0.12 & -0.88 \\ 0.6 & -0.4 \\ 3 & 2 \end{pmatrix}$$

Consider the application of above box pruning idea along the first component direction, that is along x_1. Along the direction x_1, the first row corresponds to $x_1 = a = 0.2$, and the third row corresponds to $x_1 = b = 1$. Along the first row, the minimum and maximum values of the Bernstein coefficients, respectively are -0.88 and 0.12 giving $\mathbf{g}(a) = [-0.88, 0.12]$. Along the third row, the minimum and maximum values of the Bernstein coefficients, respectively are 2 and 3 giving $\mathbf{g}(b) = [2, 3]$. Since $0 \in \mathbf{g}(a)$ the left end-point cannot be increased. However, $0 \notin \mathbf{g}(b)$, hence the right end-point can be decreased.

The partial derivative in the direction x_1 (calculated from (12)), that is, $\mathbf{g}'_{x_1}$ is obtained by computing the difference of the Bernstein coefficients across the three rows and multiplying them with the ratio of degree of x_1, and width of $\mathbf{x}_1$. Here, $\mathbf{g}'_{x_1} = [1.2, 6]$.

Now, we can perform one iteration of the Bernstein Newton contractor as

$$\begin{aligned}
\mathbf{N}(\mathbf{x}_1) &= b - (\mathbf{g}(b)/\mathbf{g}'_{x_1}) \\
&= 1 - \frac{[2,3]}{[1.2,6]} \\
&= 1 - [2,3] * \frac{1}{[1.2,6]} \\
&= 1 - [2,3] * \left[\frac{1}{6}, \frac{1}{1.2}\right], \left(\text{if } 0 \notin \left[\frac{1}{6}, \frac{1}{1.2}\right]\right) \\
&= 1 - [2,3] * [0.16, 0.83] \\
&= 1 - [0.33, 2.5] \\
&= [-1.5, 0.66]
\end{aligned}$$

Therefore, the updated value of $\mathbf{x}_1$ is

$$\begin{aligned}
\mathbf{x}'_1 &= \mathbf{N}(\mathbf{x}_1) \cap \mathbf{x}_1 \\
&= [-1.5, 0.66] \cap [0.2, 1] \\
&= [0.2, 0.66]
\end{aligned}$$

It may be noted that, a right end-point of $\mathbf{x}_1$ (*i.e.* $b = 1$) has been reduced to $b = 0.66$ with one iteration of box pruning operator. Similar process can be carried out in the other direction x_2.

3.2 Improved Bernstein global optimization algorithm

We now present our main Bernstein global optimization algorithm. This algorithm uses the Bernstein range enclosing property (equation (11)), followed by a domain subdivision, to correctly locate the global solution (global minimum and global minimizers) for a given NLP problem. Briefly, the improved Bernstein algorithm is similar to a interval branch-and-prune procedure, but with following enhancements.

- This algorithm use the Bernstein form as a inclusion function for the global optimization.
- This algorithm use the box pruning operator based on the Bernstein form (presented in Section 3.1).
- Following the Remark 1, this Bernstein algorithm always converge to the global minimum. Interested readers can refer Patil et al. (2012b) for the detailed mathematical treatment on the topic.

Algorithm improved Bernstein global:
$[\widetilde{y}, \widetilde{p}, U]$=IBBBC($N, a_I, \mathbf{x}, \epsilon_p, \epsilon_x, \epsilon_{zero}$)

Inputs: Degree N of the variables occurring in the objective and constraint polynomials, the coefficients a_I of the objective and constraint polynomials in the power form, the initial search box $\mathbf{x}$, the tolerance parameters ϵ_p and ϵ_x on the global minimum and global minimizer(s), and the tolerance parameter ϵ_{zero} to which the equality constraints are to be satisfied.

Outputs: A lower bound $\widetilde{y}$ and an upper bound $\widetilde{p}$ on the global minimum f^*, along with a set U containing all the global minimizer(s) $\mathbf{x}^{(i)}$.

BEGIN Algorithm

(1) Set $\mathbf{y} := \mathbf{x}$.
(2) From a_I compute the Bernstein coefficient arrays of the objective and constraint polynomials on the box $\mathbf{y}$ respectively as $(b_o(\mathbf{y})), (b_{gi}(\mathbf{y})), (b_{hj}(\mathbf{y}))$, $i = 1, 2, ..., m$, $j = 1, 2, ..., n$.
(3) Set $\widetilde{p} := \infty$ and $y := \min(b_o(\mathbf{y}))$.
(4) Initialize list $\mathcal{L} := \{(\mathbf{y}, y)\}$, $\mathcal{L}^{sol} := \{\}$.
(5) If $\mathcal{L}$ is empty then go to step 15. Otherwise, pick the first item $(\mathbf{y}, y)$ from $\mathcal{L}$, and delete its entry from $\mathcal{L}$.
(6) Apply the box pruning operator to the item $(\mathbf{y}, y)$ based on the equality and inequality constraints (in our case equation (5)) over $\mathbf{y}$. If the result is empty, then delete item $(\mathbf{y}, y)$ and go to step 5.

$$\mathbf{y}' = \text{box\_prune}((b_g), (b_h), \mathbf{y}, r)$$

where (b_g) and (b_h) is the Bernstein coefficient arrays of the inequality and equality functions, respectively, bound contraction will be applied in the r^{th} direction, and $\mathbf{y}'$ is the new contracted box.
(7) Set $\mathbf{y} := \mathbf{y}'$ and compute the Bernstein coefficient arrays of the objective and constraint polynomials on the box $\mathbf{y}$, respectively as $(b_o(\mathbf{y})), (b_{gi}(\mathbf{y})), (b_{hj}(\mathbf{y}))$, $i = 1, 2, ..., m$, $j = 1, 2, ..., n$. Also set $y := \min(b_o(\mathbf{y}))$.
(8) Choose a coordinate direction λ parallel to which $\mathbf{y}_1 \times \cdots \times \mathbf{y}_l$ has an edge of maximum length, that is $\lambda \in \{i : w(\mathbf{y}) := w(\mathbf{y}_i), i = 1, 2, \ldots, l\}$.
(9) Bisect $\mathbf{y}$ normal to direction λ, getting boxes $\mathbf{v}_1$, $\mathbf{v}_2$ such that $\mathbf{y} = \mathbf{v}_1 \cup \mathbf{v}_2$.
(10) for $k = 1, 2$
  (a) Find the Bernstein coefficient array and the corresponding Bernstein range enclosure of the objective function (f) over $\mathbf{v}_k$ as $(b_0(\mathbf{v}_k))$ and $B_0(\mathbf{v}_k)$, respectively.
  (b) Set $d_k := \min B_o(\mathbf{v}_k)$.
  (c) If $\widetilde{p} < d_k$ then go to substep (h).
  (d) for $i = 1, 2, \ldots, m$
    (i) Find the Bernstein coefficient array and the corresponding Bernstein range enclosure of the inequality constraint polynomial (g_i) over $\mathbf{v}_k$ as $(b_{gi}(\mathbf{v}_k))$ and $B_{gi}(\mathbf{v}_k)$, respectively.
    (ii) If $B_{gi}(\mathbf{v}_k) > 0$ then go to substep (h).
    (iii) If $B_{gi}(\mathbf{v}_k) \leq 0$ then go to substep (e)
  (e) for $j = 1, 2, \ldots, n$
    (i) Find the Bernstein coefficient array and the corresponding Bernstein range enclosure of the equality constraint polynomial (h_j) over $\mathbf{v}_k$ as $(b_{hj}(\mathbf{v}_k))$ and $B_{hj}(\mathbf{v}_k)$, respectively.
    (ii) If $0 \notin B_{hj}(\mathbf{v}_k)$ then go to substep (h).
    (iii) If $B_{hj}(\mathbf{v}_k) \subseteq [-\epsilon_{zero}, \epsilon_{zero}]$ then go to substep (f)
  (f) Set $\widetilde{p} := \min(\widetilde{p}, \max B_o(\mathbf{v}_k))$.
  (g) Enter $(\mathbf{v}_k, d_k)$ into the list $\mathcal{L}$ such that the second members of all items of the list do not decrease.
  (h) end (of $k-$loop).
(11) {Cut-off test} Discard all items $(\mathbf{z}, z)$ in the list $\mathcal{L}$ that satisfy $\widetilde{p} < z$.
(12) Denote the first item of the list $\mathcal{L}$ by $(\mathbf{y}, y)$.
(13) If $(w(\mathbf{y}) < \epsilon_x)$ & $(\max B_o(\mathbf{y}) - \min B_o(\mathbf{y})) < \epsilon_p$ then remove the item from the list $\mathcal{L}$ and enter it into the solution list $\mathcal{L}^{sol}$.
(14) Go to step 5.
(15) {Compute the global minimum} Set the global minimum $\widetilde{y}$ to the minimum of the second entries over all the items in $\mathcal{L}^{sol}$.
(16) {Compute the global minimizers} Find all those items in $\mathcal{L}^{sol}$ for which the second entries are equal to $\widetilde{y}$. The first entries of these items contain the global minimizer(s) $\mathbf{x}^{(i)}$.
(17) Return the lower bound $\widetilde{y}$ and upper bound $\widetilde{p}$ on the global minimum f^*, along with the set U containing all the global minimizer(s) $\mathbf{x}^{(i)}$.

END Algorithm

4. NUMERICAL SIMULATION ON A SMIB POWER SYSTEM

In this section, we study the highly nonlinear model of a single machine infinite bus (SMIB) power system depicted in Fig. 1. We choose this system as it is widely used in the power system control literature and exhibits accurate description of the synchronous generator behavior. We use the following SMIB system dynamical model in a $d-q$ reference frame [Kundur (1994)].

$$\dot{\delta} = \Omega_B(\omega - \omega_r) \tag{13}$$

$$\dot{\omega} = \frac{1}{2H}\left(P_m - P_e - K_d(\omega - \omega_r)\right) \tag{14}$$

$$\dot{e}'_q = \frac{1}{T'_{d0}}\left(-\frac{X_{dr}e'_q}{X'_{dr}} + \frac{(X_{dr} - X'_{dr})}{X'_{dr}}\nu_r\cos(\delta) + e_{fd}\right) \tag{15}$$

$$\dot{e}_{fd} = -\frac{1}{T_A}e_{fd} + \frac{K_A}{T_A}(1 - V_t) + \frac{K_A}{T_A}u \tag{16}$$

where

$$P_e = \frac{e'_q \nu_r sin(\delta)}{X'_{dr}} + \frac{(X_{dr} - X'_{dr})}{X_{qr} X'_{dr}} \nu_r^2 \cos(\delta)\sin(\delta).$$

$$V_t = \sqrt{(e'_q - X'_d I_d)^2 + (X'_q I_q)^2}.$$

$$X_{dr} = X_d + X_r,\ X_{qr} = X_q + X_r,\ X'_{dr} = X'_d + X_r.$$

where δ is the rotor angle of the generator, ω is the rotor speed deviation, H is the inertia constant of the generator, P_m is the mechanical input power to the generator which is assumed to be constant, K_d is the damping constant of the generator, P_e is the electrical power delivered by the generator, e_{fd} is the field voltage of the generator, V_t is the terminal voltage of the generator, and u is the control input from the controller which modulates e_{fd}. All parameters are expressed in per unit (pu) and listed in Table 1.

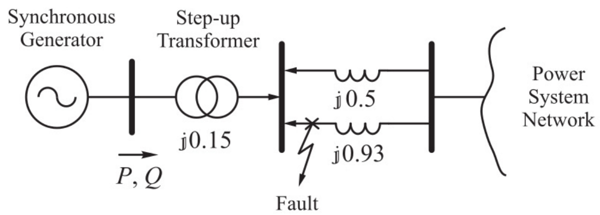

Fig. 1. Classical single machine infinite bus power system network [Kundur (1994)].

For the simulation studies, we consider the following two typical scenarios:
Scenario I: A short-circuit fault occurs at one of the two parallel transmission lines shown in Fig. 1 (mechanical power P_m is assumed to be constant at 0.9 pu).
Scenario II: 5 % and 10 % step changes of the input mechanical power to the generator.

For both scenarios, we simulate an NMPC scheme to maintain SMIB system at its equilibrium point. The nonlinear model in (13)-(16) is discretized using Euler's method for the simulation studies, and the NMPC control law is derived by solving an NLP of the form (4)-(5) using the improved Bernstein algorithm IBBBC. The solution for the updated states is computed based on the set of given initial conditions and first optimal control move derived by a NMPC control law. We adopted the following parameters values for the simulation:

- sampling time of 0.025 seconds
- prediction horizon, $N = 3$
- $Q = \text{diag}(1\ 1\ 1\ 1)^T$ and $R = 1$ as weighting matrices
- initial conditions, $x_0 = [1.225\ 1\ 1.023\ 2.42]^T$ and $u_0 = 0$
- constraints on the control input, $-0.1 \leqslant u \leqslant 0.1$
- tolerances, $\epsilon_p = \epsilon_x = \epsilon_{zero} = 0.001$ in the algorithm IBBBC on the global minimum and minimizers

To compare the performance of an NMPC scheme, we choose the two well established control schemes from the power systems literature, namely, power system stabilizer (PSS) [Kundur (1994)] and partial feedback linearization (PFL) [Mahmud et al. (2014)]. All control schemes were implemented in the MATLAB environment on desktop PC running an Intel®Core i7-5500U CPU processor running at 2.40 GHz with a 4 GB RAM. We below briefly discuss the simulation results of the two scenarios.

Scenario I: In order to validate the effectiveness of an NMPC scheme under a disturbance, a short-circuit fault of a 100 milliseconds is considered between two parallel transmission lines (refer Fig. 1). The fault occurs at $t = 2$ seconds and is cleared at $t = 2.1$ seconds. In practice this event is considered as the perturbation which moves the states (δ, ω, e'_q, e_{fd}) from their equilibrium. The system may become unstable during post-fault period due to insufficient damping. Hence, the controller has two roles: i) to provide some additional damping during post-fault period; and ii) to bring the states back to their equilibrium point.

Figures 2 shows the rotor angle (δ) response of the synchronous generator. The dotted line indicates the system response with the classical PSS; dash-dotted line indicates case with the PFL; whereas solid line shows the system behavior with an NMPC scheme. It can be observed that, the NMPC scheme results in better damping compared to the PSS and slightly better settling time with respect to PFL.

Similarly, we note that synchronous generator speed deviation ($\nabla\omega$) is zero at equilibrium point. However, as the short-circuit fault occurs, the speed is also disturbed. Fig. 3 shows the speed deviation response of the generator, where the dotted line indicates the system response with the classical PSS; dash-dotted line indicates case with the PFL; whereas solid line shows the system behavior with an NMPC scheme. In this case too NMPC ensures good transient stability compared with the PSS and slightly better settling time when compared with the PFL.

Similar findings can be seen in transient responses in the generator terminal voltage from Fig. 4. Fig. 5 shows the control signal delivered to the generator. We note that both PSS and PFL had a large oscillating damping before settling. On the other hand, NMPC control moves had a small oscillations and was well within the saturation limits (± 0.1).

Scenario II: In this case study, we first simulate the power system with a nominal mechanical power ($P_m = 0.9$). Then we assume a sudden disturbance on the input side of the generator (such as, drop in the steam pressure used to rotate the turbines) which result in a change in the mechanical power (P_m) of the generator. Typically, we consider a 5% step change in P_m from its nominal value (i.e. 0.9 to 0.8548) at 2 seconds and again 10% step change in P_m (i.e. 0.8548 to 0.9403) at 5 seconds. For the first step change in P_m (reduction by 5% from its nominal value) and constant load, the difference between the electrical power generated and the desired load is met by reducing the rotor speed (ω) and as a result the rotor angle (δ) settle down to a lower equilibrium point value. Similarly, vice-versa is observed when P_m is increased by 10% from its nominal value.

The results of the rotor angle (δ) and the its speed deviation ($\nabla\omega$) are shown in Figures 6 and 7, respectively. The dotted line indicates the system response with the classical PSS; dash-dotted line indicates case with the PFL; whereas solid line shows the system behavior with the NMPC scheme. It can be observed that, the NMPC scheme is superior than the PSS and performs slightly better than the PFL in terms of the damping and settling time.

Fig. 8 shows the generator terminal voltage response to the changes in the mechanical power P_m. It can be observed that in this large period of transient, the PSS scheme responds poorly. On the other hand, the PFL and NMPC schemes performs satisfactory keeping terminal voltage close to its nominal value of 1.

Finally, to assess the practical applicability of the Bernstein algorithm (IBBBC) based NMPC scheme, we compare the computational times under the aforementioned Scenarios I and II. Fig. 9 show the computation time taken to compute the control move at each sampling instant (*i.e.* to solve an NLP of the form (4)-(5)) by the algorithm IBBBC. We observed the control move computation time well within the sampling period of 0.025 seconds (on an average 0.016 seconds in both the scenarios).

5. CONCLUSIONS

In this work a model predictive control scheme for the nonlinear systems (termed as NMPC) was presented. The specific highlight of our NMPC scheme was use of the improved Bernstein global optimization procedure to solve the nonlinear programming problems at each sampling instant to derive the control law. The NMPC scheme is expected to be of practical benefit for the power systems, because of its ability to handle constraints efficiently. Further, we believe such NMPC scheme can be benefited from the Bernstein algorithms due to their ability to locate correct global optimal solutions for the online optimization problems. Together this may lead to improved control (better damping and settling time) as demonstrated for a single machine infinite bus power system studied in this work.

Table 1. List of SMIB parameters and data Kundur (1994).

Parameter	Parameter	Value
Base angular frequency	Ω_B	$2\pi \times 50$ rad/sec
Network bus voltage	ν_r	0.90081 pu
d-axis transient time constant	T'_{d0}	8 sec
d-axis reactance	X_d	1.81 pu
d-axis transient reactance	X'_d	0.3 pu
q-axis reactance	X_q	1.76 pu
Transmission line reactance	X_r	0.475 pu
Mechanical power	P_m	0.9 pu
Generator exciter gain	K_A	200
Generator exciter time constant	T_A	0.001 sec
Generator inertia constant	H	3.5 sec
Stable equilibrium point	$[\delta \ \omega \ e'_q \ e_{fd}]$	[1.225 1 1.023 2.42]

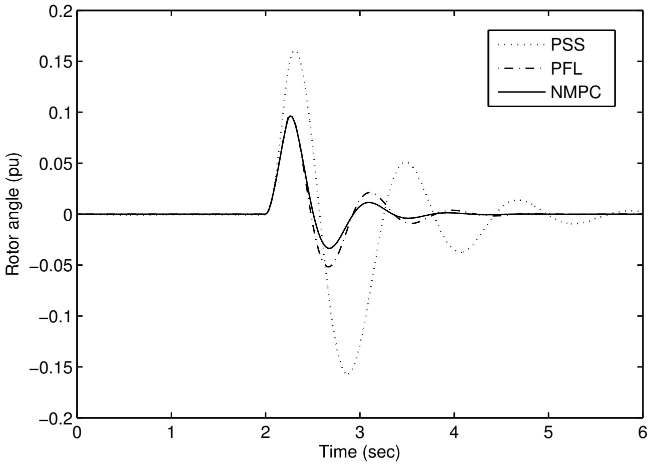

Fig. 2. Rotor angle response to under a 100 ms short circuit fault (Scenario I).

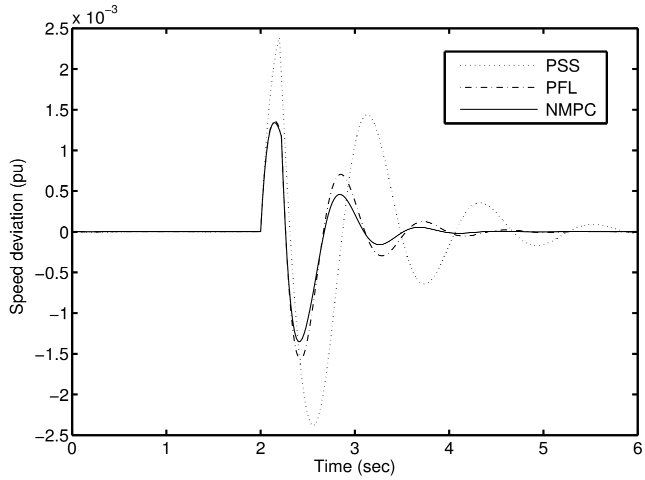

Fig. 3. Rotor speed deviation response under a 100 ms short circuit fault (Scenario I).

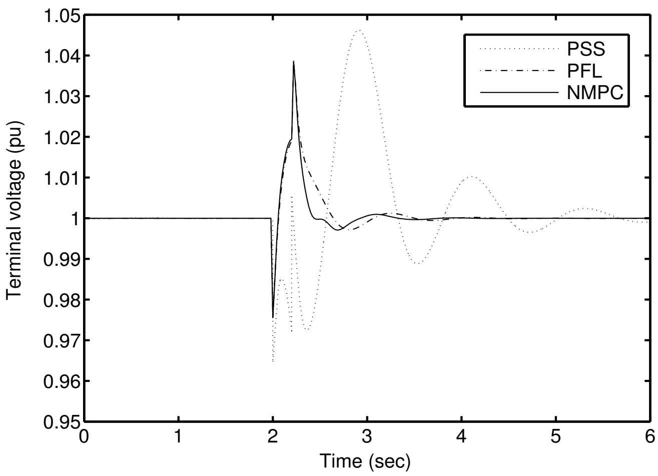

Fig. 4. Generator terminal voltage under a 100 ms short-circuit fault (Scenario I).

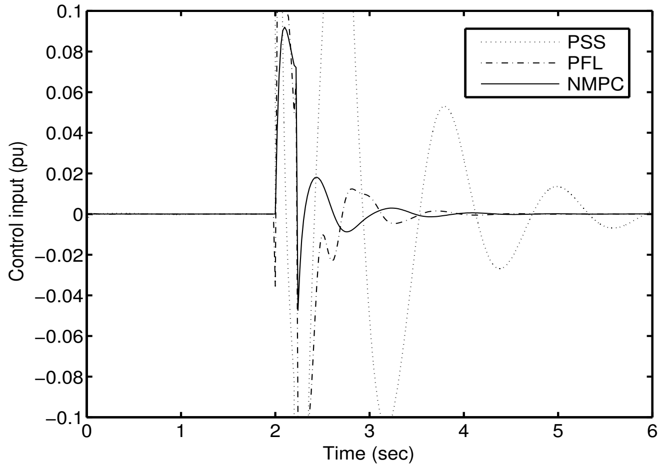

Fig. 5. Control signals under a 100 ms short-circuit fault (Scenario I).

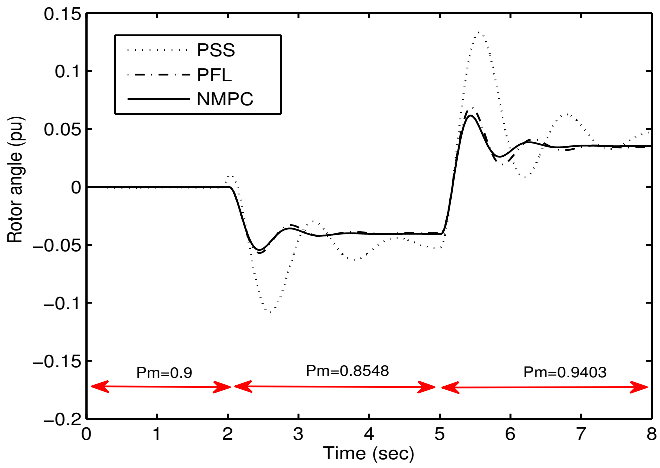

Fig. 6. Rotor angle response with changes in mechanical power input (Scenario II).

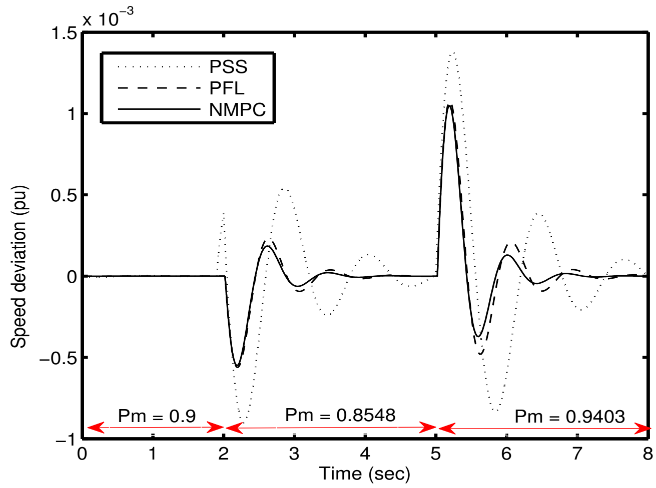

Fig. 7. Rotor speed deviation response with changes in mechanical power input (Scenario II).

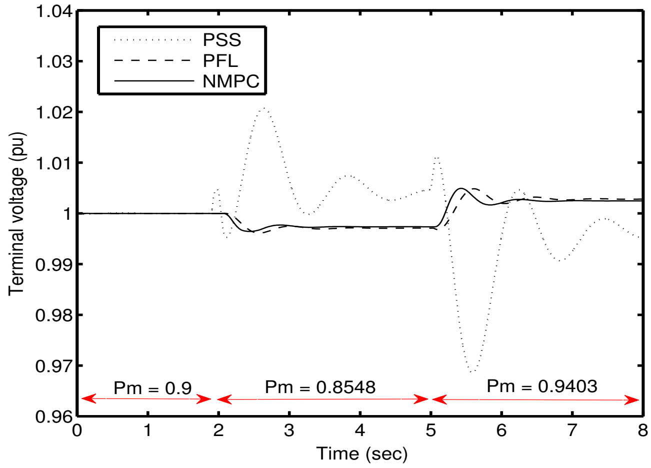

Fig. 8. Generator terminal voltage response with changes in mechanical power input (Scenario II).

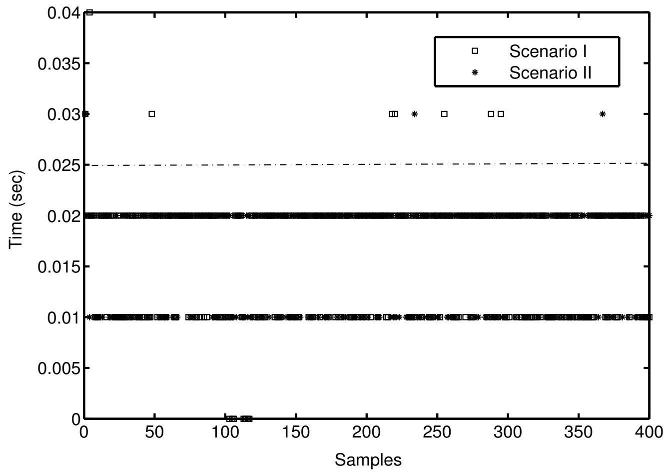

Fig. 9. Comparison of the computation time needed for a solution of an NLP (in NMPC scheme) at each sampling instant with the Bernstein algorithm (IBBBC) for the Scenarios I and II. The dotted line at 0.025 shows the sampling time.

ACKNOWLEDGEMENTS

This research is supported by the National Research Foundation, Prime Ministers Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme.

REFERENCES

Aufderheide, B. and Bequette, B.W. (2003). Extension of dynamic matrix control to multiple models. *Computers and Control Engineering*, 27(8-9), 1079–1096.

Badgwell, T.A. and Qin, S.J. (2014). Model-predictive control in practice. *Encyclopedia of Systems and Control*, 1–6.

Camacho, E.F. and Bordons, C. (2004). *Model predictive control*, 2^{nd} *ed.* Springer-Verlag, London.

Darby, M.L. and Nikolaou, M. (2012). MPC: Current practice and challenges. *Control Engineering Practice*, 20(4), 328–342.

Findeisen, R., Allgöwer, F., and Biegler, L. (2007). Springer-Verlag, Assessment and future directions of nonlinear model predictive control, *Lecture Notes in Control and Information Sciences.*

Hansen, E.R. and Walster, G.W. (2005). *Global optimization using interval analysis, 2^{nd} edition.* Marcel Dekker, New York.

Hedengren, J.D., Shishavana, R.A., Powell, K.M., and Edgar, T.F. (2014). Nonlinear modeling, estimation and predictive control in APMonitor. *Computers and Chemical Engineering*, 70(5), 133–148.

Kundur, P. (1994). *Power system stability and control, 1^{nd} edition.* McGraw-Hill Education, New York.

Lawrence, D.A. and Rugh, W.J. (1995). Gain scheduling dynamic linear controllers for a nonlinear plant. *Automatica*, 31(3), 381–390.

Maciejowski, J.M. (2002). *Predicitve control with constraints.* Prentice Hall, UK, Harlow.

Mahmud, M.A., Pota, H.R., Aldeen, M., and Hossain, M.J. (2014). Partial feedback linearizing excitation controller for multimachine power systems to improve transient stability. *IEEE Transactuions on Power Systems*, 29(2), 561–571.

Martinsen, F., Biegler, L.T., and Fossa, B.A. (2004). A new optimization algorithm with application to nonlinear MPC. *Journal of Process Control*, 14(8), 853–865.

Mayne, D.Q. and Rawlings, J.B. (2000). Constrained model predictive control: stability and optimality. *Automatica*, 36, 789–814.

Patil, B.V., Bhartiya, S., Nataraj, P.S.V., and Nandola, N.N. (2012a). Multiple-model based predictive control of nonlinear hybrid systems based on global optimization using the Bernstein polynomial approach. *Journal of Process Control*, 22(2), 423–435.

Patil, B.V. and Nataraj, P.S.V. (2016). *The Bernstein branch-and-bound unconstrained global optimization algorithm for MINLP problems, in: M. Nehmeier, J. W. Gudenberg, W. Tucker (Eds.), Lectures Notes in Computer Science.* vol. 9553, pp. 180-198, Springer, Switzerland.

Patil, B.V., Nataraj, P.S.V., and Bhartiya, S. (2012b). Global optimization of mixed-integer nonlinear (polynomial) programming problems: the Bernstein polynomial approach. *Computing*, 94(2-4), 325–343.

Qin, S.J. and Badgwell, T.A. (2003). A survey of industrial model predictive control technology. *Control Engineering Practice*, 11, 733–764.

Rao, C.V. and Rawlings, J.B. (2000). Linear programming and model predictive control. *Journal of Process Control*, 10(2-3), 283–289.

Ratschek, H. and Rokne, J. (1988). *New computer methods for global optimization.* Ellis Horwood Publishers, Chichester, England.