Automatic weight determination in nonlinear model predictive control of wind turbines using swarm optimization technique

Elham Tofighi¹, Amin Mahdizadeh²

¹School of Electrical Engineering and Computer Science, University of Newcastle, NSW 2308, Australia
²Department of Electrical and Electronic Engineering, University of Melbourne, Victoria 3010, Australia

E-mail: elham.mohammadalipourtofighi@newcastle.edu.au
a.mahdizadeh@students.unimelb.edu.au

Abstract.
This paper addresses the problem of automatic tuning of weighting coefficients for the nonlinear model predictive control (NMPC) of wind turbines. The choice of weighting coefficients in NMPC is critical due to their explicit impact on efficiency of the wind turbine control. Classically, these weights are selected based on intuitive understanding of the system dynamics and control objectives. The empirical methods, however, may not yield optimal solutions especially when the number of parameters to be tuned and the nonlinearity of the system increase. In this paper, the problem of determining weighting coefficients for the cost function of the NMPC controller is formulated as a two-level optimization process in which the upper-level PSO-based optimization computes the weighting coefficients for the lower-level NMPC controller which generates control signals for the wind turbine. The proposed method is implemented to tune the weighting coefficients of a NMPC controller which drives the NREL 5-MW wind turbine. The results are compared with similar simulations for a manually tuned NMPC controller. Comparison verify the improved performance of the controller for weights computed with the PSO-based technique.

1. Introduction
Wind turbines are complex and nonlinear systems operating against variable speed wind speeds which yield aerodynamic and mechanical loads on the turbine structure and components. To fulfill feasibility concerns, it is crucial for wind turbines to operate at their maximum capacity while mechanical damages to the turbine structure are kept minimal [1, 2]. To achieve these goals, different control techniques have been implemented to control wind turbines. Among these approaches, model predictive control (MPC) has received special attention due to its ability to handle multi-input multi-output systems and include actuator and operation constraints in the control problem formulation [3, 4]. MPC utilizes an internal model of the system along with measurements and estimations of current states and disturbances to predict the future behavior of the system over a given period of time. The control signals are computed via optimization of a performance and/or operation criteria (i.e., cost function) in an open-loop optimal control problem (OCP) fashion. From these control sequences, only the first segment is applied to the system at the current instant. In the next time step, the prediction horizon is shifted one step into the future, measurements are re-acquired and the whole procedure is repeated (i.e., receding horizon policy) [5].
Two critical concerns that affect the application of MPC controllers are first development of accurate process model and second appropriate tuning of MPC parameters [6]. The MPC parameters may be selected intuitively and through trial and error. However, it is not always straightforward to comprehend the relationship between the qualitative value of parameters and their physical significance. Hence, the empirical methods might not yield the optimum results especially when the number of parameters to be considered and the nonlinearity of the system increase. Moreover, several evaluations may be required in order to figure our the most appropriate settings. Several studies have investigated different theory-based strategies and heuristic/industrial methods in tuning of MPC controllers. It can be shown that the infinite prediction horizon guarantees stability of MPC while the finite prediction horizons should be carefully tuned to ensure closed-loop stability of the controller. The problem of tuning the prediction horizon length in MPC has been investigated in [7]. Furthermore, the MPC cost function contains mathematical translation of control objectives that includes one or several terms penalizing and/or regulating input, output, and state variables. These terms are tuned using weighting coefficients. The absolute value assigned to each of these weights determines the relative importance of the corresponding term in influencing the NMPC process and specifies how strongly the terms would effect the overall turbine operation. The weights on the system outputs facilitate the scaling of control variables such that more effort is directed towards more important manipulated outputs for more accurate control. Tuning of the weights on system outputs has been investigated in [8, 9]. The rate of change of input signals affects the robustness of the controller. Hence, it should be tuned carefully [10, 11]. The penalization of magnitude of inputs allows removal of constraints from the optimization problem which would reduce the computational efforts in MPC [12]. The reference tracking terms in MPC ensure a smooth transition from measured/estimated outputs to the desired set-points. The tuning problem for these parameters are investigated in [13, 14]. Finally, the constraints in MPC problem should be carefully determined to guarantee the feasibility of the optimization problem. The constraint tuning methods are discussed in [15].

The aforementioned approaches are off-line methods and demand the control engineer to not only have sufficient knowledge of the MPC but to be familiar with dynamics of the system under control. In automatic parameter determination methods, on the other hand, the tuning parameters get updated along with the optimization algorithm which means they are set to optimal values [16, 17]. Evolutionary computation algorithms such as genetic algorithm and particle swarm optimization (PSO) have been used to determine the MPC parameters automatically in [18] and [19]. In this paper, we propose a novel automatic parameter determination method that can be implemented to optimal tuning of weighting coefficients in the cost function of the nonlinear model predictive controller (NMPC) of wind turbines.

The cost function terms in NMPC problem of wind turbines are generally selected with respect to control objectives and aim to ensure the feasibility of the turbine operation. Generally, there are no analytic or numerical methods to directly determine the weighting coefficients for the NMPC problem. With intuitive understanding of the turbine operation, these weights have been chosen empirically in previous studies [20–22]. In this paper, we introduce a novel parameter identification method that employs the PSO technique to identify optimal weights for the NMPC controller of wind turbines automatically. Here, the tuning problem for NMPC is formulated as an optimization problem and a customized PSO algorithm is applied as the optimization tool. To verify the performance of the proposed method, simulation are performed based on the NREL 5-MW wind turbine model [23]. The weights are computed and minimized through a quadratic objective function. The proposed method is, according to the best knowledge of the authors, the only systematic method implemented to the automatic determination of weighting coefficients for the NMPC controller of wind turbines.

The remainder of this paper is organized in six sections. In Section 2 the mathematical model of the NREL 5-MW reference wind turbine is provided. The NMPC controller for this wind turbine is defined and discussed in this section as well. A brief introduction to PSO technique is provided in section 3. The problem of automatic tuning of NMPC weights is detailed in section 4. The analysis and simulation results are presented in section 5 and ultimately, the conclusion is summarized in section 6.
2. System Configuration

2.1. Simplified model of the wind turbine for the controller design

Wind turbines are complex machines built of several mechanical, electrical, and aerodynamic subsystems. Therefore, it is generally infeasible to exploit full nonlinear model of wind turbines in model-based control design approaches. Hence, a low-order simplified model is obtained and adapted for controller design purposes [3]. In this research, we have used the NREL 5-MW wind turbine as the reference wind turbine. Details on the turbine design and control parameters are available from [23].

As shown in Figure 1, the turbine tower and rotor are considered as flexible components in the turbine structure while other moving parts are assumed to be rigid. Servo-elastic characteristics of the wind turbine consist of structural subsystem (i.e., drivetrain and tower dynamics) and actuator subsystem (i.e., generator torque and blade pitch control signals).

\[
J \dot{\Omega} + \frac{M_g}{i_{gb}} = M_a(x_T, \Omega, \theta, v_0) \tag{1a}
\]
\[
m_{Te} \ddot{x}_T + c_{Te} \dot{x}_T + k_{Te}(x_T - x_{T0}) = F_a(x_T, \Omega, \theta, v_0), \tag{1b}
\]
\[
\dot{\theta} + 2\zeta \omega \theta = \omega^2(\theta_c - \theta) \tag{1c}
\]

where (1a) is the equation of motion of the rotor with \( J \) being the sum of total moment of inertia of the drivetrain about the low-speed shaft, \( \dot{\Omega} \) the rotor speed rate, and \( i_{gb} \) the gearbox ratio. The aerodynamic torque, \( M_a \), accelerates the rotor while the generator torque, \( M_g \), decelerates it.

The turbine tower is subject to aerodynamic thrust forces, \( F_a \). The tower top fore-aft dynamics are described as a mass-spring-damper system given in (1b). The constant \( x_{T0} \) represents the static tower-top position in the absence of thrust forces [20]. Other constants \( m_{Te} \), \( c_{Te} \), and \( k_{Te} \) are tower equivalent modal mass, and tower damping and stiffness coefficients respectively. The blade pitch actuator, (1c), is modeled as a second-order linear system with \( \zeta \) being the damping factor, \( \omega \) the undamped natural frequency of the tower, and \( \theta_c \) the collective blade pitch reference signal.

The energy in the moving air (wind) is harvested by the aerodynamic subsystem. The aerodynamic torque, \( M_a \), and the aerodynamic thrust force, \( F_a \), are calculated as:

\[
M_a(\Omega, x_T, \theta, v_0) := \frac{1}{2} \rho \pi R^3 c_p(\lambda, \theta) \frac{\lambda}{\lambda} v_{rel}^2 \tag{2}
\]
\[
F_a(\Omega, x_T, \theta, v_0) := \frac{1}{2} \rho \pi R^2 c_T(\lambda, \theta) v_{rel}^2 \tag{3}
\]

where \( \rho \) is the air density passing through the blades plain and \( \lambda \) is the tip speed ratio (TSR) given as

\[
\lambda := \frac{R \Omega}{v_{rel}} \tag{4}
\]
Figure 2: (left) power coefficient $c_P(\lambda, \theta)$, (right) thrust coefficient $c_T(\lambda, \theta)$ for the NREL 5MW wind turbine.

The hub-height relative wind speed is obtained as $v_{rel} = v_0 - \dot{x}_T$ with $v_0$ and $\dot{x}_T$ being the averaged longitudinal wind speed in front of the turbine and tower-top oscillation rate respectively. The power coefficient, $c_P(\lambda, \theta)$, and the thrust coefficient, $c_T(\lambda, \theta)$, are computed from steady-state simulations [20]. Figure 2 illustrates the power and thrust coefficients for the NREL 5-MW wind turbine.

The simplified model of the wind turbine defined by (1) to (3) may also be shown in nonlinear state-space form as:

$$\dot{x} = f(x, u, d), \quad x(0) = x_0$$
$$y = h(x, u, d)$$

where $u$, $x$, $d$, and $y$ denote the input, state, disturbance, and output vectors and

$$u = [\dot{M}_g \quad \theta_c]^T$$
$$x = [\Omega \quad x_T \quad \dot{x}_T \quad \theta \quad \dot{\theta} \quad M_g]^T$$
$$d = v_0$$
$$y = [\Omega \quad \dot{x}_T \quad \theta \quad \dot{\theta}]^T$$

2.2. Nonlinear model predictive control design for wind turbines

Nonlinear model predictive control (NMPC) [24] is based on the solution of a finite horizon open-loop optimal control problem (OCP). At every sampling instant, $t_k = k\delta$, $k \in \mathbb{N}$ and $\delta > 0$, an OCP consisting of a model of the system dynamics and the input and state constraints is solved over a given prediction horizon, $T_p$. The mathematical formulation of the OCP formulation is shown as,

$$\min_{u(\cdot)} \int_{t_k}^{t_k + T_p} \ell(\hat{x}(\tau), u(\tau), \hat{d}(\tau)) d\tau$$

subject to $\forall \tau \in [t_k, t_k + T_p]$:

$$\dot{\hat{x}}(\tau) = f(\hat{x}(\tau), u(\tau), \hat{d}(\tau)), \quad \hat{x}(t_k) = x(t_k)$$
$$0 \geq g_i(\hat{x}(\tau), u(\tau), \hat{d}(\tau)), \quad i = 1, \ldots, n_g$$
$$u(\tau) \in \mathcal{U}$$

where $\ell(\hat{x}(\tau), u(\tau), \hat{d}(\tau)) : \mathbb{R}^{n_u} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}^+$ is the cost function that defines the system performance criteria with respect to control objectives to increase the energy harvest from the wind and alleviate
the fatigue and extreme loads on the turbine structure. The function \( f(\hat{x}(\tau), u(\tau), \hat{d}(\tau)) \) forms the equality constraints while \( g_i(\hat{x}(\tau), u(\tau), \hat{d}(\tau)) \) defines the nonlinear inequality state constraints and \( \mathcal{U} := \{ u \in \mathbb{R}^m | u_{\text{min}} \leq u \leq u_{\text{max}} \} \) is a box constraint saturating the input control signals. The optimal solution to the OCP problem in (7) is computed repeatedly at the sampling instances \( t_j = j\delta, \ j = 0, 1, ... \) yielding sequences of control trajectories \( u^* (\cdot; x(t_k)) : [t_k, t_k + T_p] \rightarrow \mathcal{U} \).

Ultimately, the closed-loop control input is selected as the first segment of this control sequence and is applied to the system at current sampling interval as

\[
u(\tau) := u^*(\tau; x(t_k)), \quad \forall \tau \in [t_k, t_{k+1})\] (8)

The wind turbine control goal is to mitigate extreme and fatigue loads on the turbine structure while the generated electrical power is maximized with respect to different wind speeds and operational conditions. With this objective in mind, a quadratic cost function is designed

\[
\ell(\hat{x}(\tau), u(\tau), \hat{d}(\tau), \tau) = q_1 (u_0(\tau)) (\Omega(\tau) - \Omega_{\text{ref}}(\tau))^2 \\
+ q_2 (\hat{x}(\tau))^2 \\
+ q_3 (u_0(\tau)) (M_g(\tau) - M_{g,\text{ref}}(\tau))^2 \\
+ q_4 (u_0(\tau)) (\dot{\theta}(\tau))^2 \\
+ q_5 (M_g(\tau))^2 \\
+ q_6 (u_0(\tau)) (\theta(\tau))^2
\] (9)

where \( q_k > 0 \) with \( k \in \{1, 2, ..., 6\} \) denotes the set of weights that are chosen to penalize and/or regulate the wind turbine behavior against variable wind speed and with respect to the desired control objectives.

It can be realized that except for the rotor speed, \( \Omega \), and the electromagnetic torque, \( M_g \), the cost function terms are regulated to the equilibrium point in the origin. The choice of the reference rotor speed trajectory, \( \Omega_{\text{ref}} \), depends on the wind speed, \( v_0 \). Traditionally, for below rated wind speeds, the controller is aims to maximize the power capture. Hence, the rotor speed is regulated such that the optimal tip speed ratio, \( \lambda_{\text{opt}} \), is tracked and the maximum power coefficient, \( c_{P,\text{max}} \), is maintained. The optimal rotor speed for below rated wind speeds is given as:

\[
\Omega_{\text{opt}}(\tau) = \lambda_{\text{opt}} \frac{v_0(\tau)}{R}
\] (10)

For above rated wind speeds, the controller attempts to maintain the rated rotor speed, \( \Omega_{\text{rated}} \). The overall reference rotor speed trajectory may be shown as:

\[
\Omega_{\text{ref}}(\tau) = \begin{cases} 
\Omega_{\text{opt}}(\tau) & \text{if } v_{\text{in}} < v_0(\tau) \leq v_{\text{rated}} \\
\Omega_{\text{rated}} & \text{if } v_{\text{rated}} < v_0(\tau) \leq v_{\text{out}}
\end{cases}
\] (11)

Similarly, to minimize the difference between the actual generated torque, \( M_g \), and the desired torque, the reference generator torque trajectory, \( M_{g,\text{ref}} \), is defined with respect to the wind speed:

\[
M_{g,\text{ref}}(\tau) = \begin{cases} 
k \left( \frac{\Omega_{\text{opt}}(\tau)}{i_{gb}} \right)^2 & \text{if } v_{\text{in}} < v_0(\tau) \leq R \frac{\Omega_{\text{rated}}}{\lambda_{\text{opt}}} \\
M_g(\tau) & \text{if } R \frac{\Omega_{\text{rated}}}{\lambda_{\text{opt}}} < v_0(\tau) \leq v_{\text{rated}} \\
M_{g,\text{rated}} & \text{if } v_{\text{rated}} < v_0(\tau) \leq v_{\text{out}}
\end{cases}
\] (12)

1 Here, we assume, for sake of simplicity, that for all \( x(t_k) \), an optimal solution (7) exists and is actually attained.
where \( k = \frac{1}{2} \rho \pi R^5 \frac{c P_{\text{max}}}{\lambda_{\text{opt}}} \frac{1}{g_b} \). The reference aerodynamic torque trajectory, \( M_a(\tau) \), is calculated from (2) with \( \Omega = \Omega_{\text{rated}} \), \( \theta = 0 \), and \( \dot{x}_{\text{r}} = 0 \).

To prevent saturation of the system actuators, hardware constraints are applied on input signals. Hence, the generator torque actuation rate, \( \dot{M}_g \), is limited by

\[
\dot{M}_g \in [-\dot{M}_{\text{g,max}}, +\dot{M}_{\text{g,max}}]
\]

The nonlinear inequality state constraints on the rotor speed, \( \Omega \), the blade pitch angle, \( \theta \), and the blade pitch rate, \( \dot{\theta} \), are given as

\[
\Omega \leq 1.2\Omega_{\text{rated}} \\
\theta_{\min} \leq \theta \leq \theta_{\max} \\
\dot{\theta} \in [-\dot{\theta}_{\max}, +\dot{\theta}_{\max}]
\]

The optimal control problem is solved using multiple direct shooting and sequential quadratic programming (SQP)/trust-region method in MATLAB®. The integration is performed using fourth-order Runge-Kutta method.

3. Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is an evolutionary population-based optimization technique [25]. The underlying concept is inspired by social behavior of animals and their team effort to capitalize and integrate collective knowledge to fulfill their survival needs. In this algorithm, potential solutions to the optimization problem (i.e., particles) move across the multi-dimensional search space utilizing a simple movement protocol which is based on semi-stochastic exploration behavior. The local and/or global optima is located through information exchange amongst particles during the process.

3.1. Principles of the PSO Algorithm

Let \( N \) be the number of particles in the swarm. Each particle is defined with its current position, \( P^j(t) \in \mathbb{R}^n \), and current velocity, \( v^j(t) \), where \( j \in \{1,...,N\} \) denotes the particle number and \( t \in \{0,...,t_{\text{max}}\} \) is the iteration number with \( t_{\text{max}} \) defining the final iteration. Additonally, every particle keeps the track of its best position.

Classically, the PSO algorithm is determined to locate the optimum of a given objective function, \( \mathcal{F} : \mathbb{R}^n \rightarrow \mathbb{R} \), via a minimization process. To initiate the PSO algorithm, particles are distributed uniformly randomly in the swarm. The better position is computed by assigning the current position of each particle in the cost function, \( \mathcal{F} \), and examining the output for higher fitness value. The position and velocity of each particle are then updated:

\[
P^j(t + 1) = P^j(t) + v^j(t + 1)
\]

\[
v^j(t + 1) = w^j(t)v^j(t) + c^j(t)r_1(t)[P^j_{\text{best}}(t) - P^j(t)]
+ s^j(t)r_2(t)[G_{\text{best}}(t) - P^j(t)]
\]

where,

- \( 0 \leq w(t) \leq 1 \) denotes the inertial bias and defines the tendency of each particle to maintain their direction of motion. In other words, it determines the trade-off between exploration and convergence behavior of the particles.
- \( 0 < c(t) < 2 \) defines the cognitive bias and regulates the particles movement step-size in the direction of the personal best position.
- \( 0 < s(t) < 2 \) represents the social bias and regulates the particles movement step-size in the direction of the global best position.
• $r_1(t), r_2(t) \sim U(0,1)$ are random and independent sequences employed to enhance the stochastic nature of the PSO algorithm.

• The personal best position of each particle is derived from

$$ P_{\text{best}}^j(t) := \arg\min \{ \mathcal{F}(P_j(t)) \mid t = 0, \ldots, t_{\text{max}}, \ j \in \{1, \ldots, N\} \} \quad (17) $$

• The global best position experienced by all particles in the swarm is given as:

$$ G_{\text{best}}(t) := \arg\min \{ \mathcal{F}(P_{\text{best}}^j(t)) \mid j \in \{1, \ldots, N\} \} \quad (18) $$

3.2. Convergence and termination criteria

Different strategies may be applied as the termination criterion of the PSO algorithm. In this work, the procedure is terminated once the iterations reach a pre-set iteration number, $t_{\text{max}}$.

4. Automatic configuration of NMPC parameters using PSO algorithm

There are several parameters in formulation of NMPC problem of wind turbines that should be tuned with respect to control objectives and the wind turbine performance criteria. Among these parameter, the prediction horizon, $N_p$, and the system and control constraints, as given in (13)-(14), are defined based on manufacturing characteristics of the wind turbine. The main concern, therefore, would be to determine the set of weighting coefficients, $q_k > 0$ and $k \in \{1,2,\ldots,6\}$, for the NMPC cost function, ($\mathcal{F}$), such that the turbine performance is optimized with respect to the control objectives. To this end, we have developed a two-level optimization process in which the weighting coefficients for the NMPC cost function are computed automatically in the upper-level PSO algorithm and then sent to the lower-level NMPC problem. The NMPC uses these weights to calculate optimal control signals for the wind turbine.

4.1. Objective function of the PSO process

The upper-level PSO algorithm is designed to allocate the optimal weighting coefficients for the NMPC cost function in (9). The set of admissible weighting coefficients is defined as the position of every particle, $P^j$, and is given as

$$ P^j = [q_1 \ q_2 \ q_3 \ q_4 \ q_5 \ q_6]^T \quad (19) $$

Generally, wind turbines are designed to operate at their nominal capacity in full-load operation region. However, due to wind speed fluctuations, it is very common for the turbine to switch between full- and partial-load operation regions and operate in transition region. Therefore, the weighting coefficients can be functions of the wind speed prioritizing the control objectives for below and above rated operation conditions. Consequently, the upper-level PSO optimization should be adjusted based on the wind speed and therefore, the wind turbine operation region. To this end, the PSO objective function, $\mathcal{F} : \mathbb{R}^n \rightarrow \mathbb{R}$, is developed to address the wind turbine control goals directly as

$$ \mathcal{F}(P) = \frac{1}{E_{\text{total}}^2} + M_{yT,\text{max}}^2 + \Delta\Omega_{\text{max}}^2 \quad (20) $$

where $E_{\text{total}}, M_{yT,\text{max}},$ and $\Delta\Omega_{\text{max}}$ are normalized terms with respect to their nominal values and denote the total electrical energy produced by the turbine, maximum mechanical loads on the turbine tower, and maximum deviation of the rotor speed from the reference rotor speed receptively.

4.2. PSO dynamics and evolution

The initial position of particles, $P_j(0)$, are uniformly distributed throughout the search space based on a random uniform distribution. At every iteration, the position and velocity of each particle, $P^j(t)$, is updated using the PSO dynamics introduced in (15)–(18). The stochastic property in each particle’s
movement is biased through random variables $r_1$ and $r_2$ while the bias coefficients, $w$, $c$, and $s$ influence this random behavior. We have developed a success-based approach that ensures the particles move in the direction of convergence at every iteration [26]:

$$w_j(t) = \begin{cases} 0.1 & \text{if } \hat{F}_j(t) < \hat{F}_j(t-1) \\ 0 & \text{otherwise} \end{cases}$$

(21)

where, for any successful step, the particles maintain their direction of motion during the next iteration and for any unsuccessful step, the particles avoid moving towards the same direction in favor of exploring behavior. The cognitive and social bias parameters are set $c_i(t) = 1.9$ and $s_i(t) = 1.6$ respectively.

4.3. PSO Constraints
To prevent particles from leaving the search area, we have employed a velocity clamping policy where the upper amplitude of the velocity of each particle is bounded as follows:

$$v_j(t) = \begin{cases} v_j(t) & \text{if } \|v_j(t)\| < v_{\text{max}} \\ v_{\text{max}} & \text{otherwise} \end{cases}$$

(22)

Moreover, to limit the search space, the position of particles are confined as $0 \leq \|P_j\|_\infty \leq 5$.

5. Analysis and simulation results
In order to verify the performance of the proposed two-level PSO-based technique in automatic determination of weighting coefficients for the NMPC cost function, the NREL 5MW wind turbine, [23], is utilized as the simulation model. With the wind turbine control goals being maximizing the energy harvest whilst reducing the aerodynamic loads on the turbine tower, the cost function of the NMPC controller in (9) is tuned through weighting coefficients $q_k > 0$ with $k \in \{1, 2, \ldots, 6\}$. To compute optimum weighting coefficients, the PSO algorithm is customized (as explained in section 4) and implemented as the upper-level optimization in combination of the lower-level NMPC.

It is assumed that the perfect wind field measurements are available using a LIDAR device. The extreme operational conditions are considered and the wind turbine is exposed to the rotor-effective Extreme Operation Gust (EOG). Due to space limitations, only the simulation and analysis results for the critical transition region are discussed here. Hence, the EOG wind with a mean wind speed of $\bar{v}_0 = 9.2 m/s$ is considered. The number of particles and maximum iterations for the PSO are set to $N = 30$ and $t_{\text{max}} = 200$ while the prediction horizon for the NMPC problem is set $N_p = 24$. The OCP is solved every $\delta = 0.25$ seconds and the whole simulation time is set to 50 seconds. The parallel processing toolbox of MATLAB® 2015b is used as the simulation environment for modeling the wind turbine, NMPC controller and the PSO algorithm.

In the beginning of the process, the PSO is initialized with random distribution of particles throughout the swarm and the position, $P_j$, and velocity $v_j$ of each particle is stored. In the next step, the better position of each particle is examined by assigning the position of each particle in the objective function of the PSO. The objective function of the PSO, calls the lower-level NMPC optimization and evaluates the value of the NMPC cost function. The position and velocity of each particle are updated based on this evaluation. The process is repeated for maximum $t_{\text{max}} = 200$ iterations when the optimum particle containing the optimal weighting coefficients for the NMPC cost function are achieved as: $q_1 = 2.4752$, $q_2 = 2.7169$, $q_3 = 3.8307$, $q_4 = 0.2297$, $q_5 = 0.0143$, and $q_6 = 3.5846$.

To examine the performance of these weighting coefficients, we implemented them to tune the NMPC cost function which is integrated to drive the simplified model of the wind turbine. The simulation results are shown in Figure 3(left). The results are then compared with simulation outcomes from implementing the same NMPC controller that is tuned using empirical weighting coefficients selected manually through...
trial and error (Figure 3(left)) [27]. It can be realized that the NMPC controller tuned with the weighting coefficients derived by the PSO-based optimization improves the performance of the controller with weighting coefficients, computed by the proposed two-level PSO-based approach yield much efficient performance of the wind turbine.

For more comprehensive study, a second scenario is considered. In this case, the wind turbine is exposed to an EOG with higher mean wind speed, $\bar{v}_0 = 11\text{m/s}$, which drives the turbine to operate just below rated wind speed. Here, the NMPC cost function is tuned using the same weighting coefficients computed by the proposed two-level PSO-based optimization. The results are drawn in Figure 3(right). Similar to the first case, the NMPC was also tuned with empirical weighting coefficients from [27] and the performance of the wind turbine was assessed. A summary of comparison between two methods of tuning the NMPC are summarized in Table 1. The comparison results verify that the accuracy of the proposed PSO-based optimization technique in tuning the NMPC parameters in transitional operation region.

### 6. Conclusion

The problem of automatically determining weighting coefficients of the cost function in NMPC controller of wind turbines is investigated and a novel method, utilizing particle swarm optimization (PSO), was introduced. Unlike the traditional empirical methods of tuning NMPC weights that are based on trial and error and require well-established knowledge of the system under control, the proposed PSO-based technique employs a systematic method and solves an optimization problem to allocate a mix of weights that yield the most efficient performance of the wind turbine with respect to the desired control objectives.
Table 1: Performance comparison: Peak values from Figure 3

|                      | EOG 9m/sec | EOG 11m/sec |
|----------------------|------------|-------------|
|                      | \(M_{\gamma T,\text{max}}\) | \(\Delta \Omega_{\text{max}}\) | \(M_{\gamma T,\text{max}}\) | \(\Delta \Omega_{\text{max}}\) |
| PSO-based-NMPC       | 47.63      | 0.26        | 60.27      | 0.80        |
| Manually-Tuned-NMPC  | 59.52      | 0.65        | 74.98      | 1.31        |

Performance improvement

PSO-based NMPC/Manually-tuned-NMPC 20.3% 60% 19.61% 38.9%

Several simulations were performed and the results verified the efficiency of the proposed technique. Future development would focus on assessing the performance and robustness of the proposed method for stochastic turbulent wind speeds in partial- and full-load operation conditions.

References

[1] Laks J, Pao L Y, Simley E, Wright A, Kelley N and Jonkman B 2011 Model predictive control using preview measurements from lidar Proc. 49th AIAA Aerospace Sciences Meeting, Orlando, FL
[2] Santos R A 2007 Damage mitigating control for wind turbines Ph.D. thesis University of Colorado
[3] Bottasso C L, Croce A, Savini B, Sirchi W and Trainelli L 2006 Multibody System Dynamics 16 291–308
[4] Henriksen L C 2011 Model predictive control of wind turbines Ph.D. thesis Technical University of Denmark, DTU, DK-2800 Kgs. Lyngby, Denmark
[5] Maciejowski J M 2002 Predictive control: with constraints (Pearson education)
[6] Santos R A 2007 Damage mitigating control for wind turbines Ph.D. thesis University of Colorado
[7] Bottasso C L, Croce A, Savini B, Sirchi W and Trainelli L 2006 Multibody System Dynamics 16 291–308
[8] Rowe C and Maciejowski J 2000 Tuning mpc using h loop shaping American Control Conference, 2000. Proceedings of the 2000 vol 2 (IEEE) pp 429–434
[9] Trierweiler J O and Farina L A 2003 Journal of Process Control 13 591–598
[10] Maurath P R, Laub A J, Seborg D E and Mellichamp D A 1988 Industrial & engineering chemistry research 27 1204–1212
[11] McIntosh A R, Shah S L and Fisher D G 1991 The Canadian Journal of Chemical Engineering 69 97–110
[12] Garriga J L and Soroursh M 2008 Model predictive controller tuning via eigenvalue placement American Control Conference, 2008 (IEEE) pp 429–434
[13] Al-Ghazzawi A, Ali E, Nouh A and Zafiriou E 2001 Journal of Process Control 11 265–284
[14] Clarke D W, Mohtadi C and Tuffs P 1987 Automatica 23 137–148
[15] Rawlings J B and Muske K R 1993 Automatic Control, IEEE Transactions on 38 1512–1516
[16] Barić M, Baotić M and Morari M 2005 On-line tuning of controllers for systems with constraints Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC’05. 44th IEEE Conference on (IEEE) pp 8288–8293
[17] Lee J and Yu Z 1994 Computers & chemical engineering 18 15–37
[18] Van der Lee J, Svreck W and Young B 2008 ISA transactions 47 53–59
[19] Suzuki R, Kawai F, Ito H, Nakazawa C, Fukuyama Y and Aiyoshi E 2007 Automatic tuning of model predictive control using particle swarm optimization Swarm Intelligence Symposium, 2007. SIS 2007. IEEE (IEEE) pp 221–226
[20] Schlipf D, Schlipf J and Kühn M 2013 Wind Energy 16 1107–1129
[21] Koerber A and King R 2011 Proc. EWEA
[22] Tofghi E, Faulwasser T and Kellett C M 2015 Nonlinear model predictive control approach for structural load mitigation of wind turbines in presence of wind measurement uncertainties Control Conference (AUCC), 2015 5th Australian (IEEE) pp 210–214
[23] Jonkman J M, Butterfield S, Musial W and Scott G 2009 Definition of a 5-MW reference wind turbine for offshore system development (National Renewable Energy Laboratory Golden, CO)
[24] Allgower F, Findeisen R, Nagy Z K et al. 2004 Journal-Chinese Institute Of Chemical Engineers 35 299–316
[25] Eberhart R C, Kennedy J et al. 1995 A new optimizer using particle swarm theory Proceedings of the sixth international symposium on micro machine and human science vol 1 (New York, NY) pp 39–43
[26] Mahdizadeh A and Schmid R 2015 Robust eigenvalue assignment via particle swarm optimization methods Control Conference (AUCC), 2015 5th Australian (IEEE) pp 153–157
[27] Tofghi E, Schlipf D and Kellett C M 2015 Nonlinear model predictive controller design for extreme load mitigation in transition operation region in wind turbines Control Applications (CCA), 2015 IEEE Conference on (IEEE) pp 1167–1172