Optimised Aerodynamics and Control by Nonlinear Model based Predictive Control

J. Schuurmans
DotX Control Solutions
j.schuurmans@dotxcontrol.com

E.A. Nederkoorn
DotX Control Solutions
e.nederkoorn@dotxcontrol.com

S. Kanev
Energy Centre Netherlands
kanev@ecn.nl

R. Rutteman
XEMC Darwind
r.rutteman@xemc-darwind.com

E. Nguyen
Mitsubishi Power Systems
eric_nguyen@mhips.com

Abstract:

Reducing the cost of wind energy requires integrating improvements in all relevant fields among which aerodynamics and control.

If Nonlinear Model Predictive Control (NMPC) can successfully be applied in practice, it could be an ideal tool in optimising the aerodynamic design of a wind turbine. This paper reports on the development of an NMPC controller for wind turbines, and on an evaluation thereof. The details of the controller are presented, including methods to deal with limited computation times, robust stability requirements and sensor noise.

The NMPC controller is tested in aero-elastic simulations in load cases as defined in IEC 61400-1 edition 3, on a commercial wind turbine. Its results are compared with two alternative controllers. The results demonstrate that NMPC is able to result in more reduced fatigue loads AND ultimate loads. The latter is due the ability of NMPC to handle both input and output constraints.

Keywords: wind turbine control, NMPC, Nonlinear Leunberger observer, sensor noise, robust stability.

1. Introduction

There is an ongoing need for reducing the cost of wind energy and increasing the energy extraction from wind. These goals can be maximised by integrating improvements in all relevant fields among which aerodynamics and control. If the aerodynamics of the wind turbine design is to be optimised, the design of the aerodynamics and the control system need to be optimised jointly.

Wind turbine controllers, applied to commercial wind turbine designs, use a more and more elaborate scheme of PID controllers to achieve this. They include active damping loops and Individual Pitch Control loops, see e.g. [1], [2]. Local PID controllers cannot optimally handle interactions, nonlinearities and constraints, which led researchers to investigate the potential benefits of MIMO model based control. Selvam et al. [3] and Balas [4] used LQR to achieve, among others, individual pitch control of the wind turbine. LQR assumes linear time invariant system's behaviour, and therefore requires some kind of gain scheduling. This led other researchers, including [5], and [6], to apply linear parameter varying control (LPV). In all aforementioned control methods, the system is first approximated by a linear system, but feedback linearization can handle nonlinear systems and does not require this approximation. This approach has been applied to wind turbine control in [7] and [8].

However, the best control performance can be expected from control techniques that can optimise the control of the nonlinear wind turbine dynamics on-line, while taking constraints into account. Nonlinear Model Predictive Control (NMPC) is currently one of the best options to do just that. Therefore, it could be an ideal tool in optimising the aerodynamic design of a wind turbine.

Several researchers have investigated the potential benefits of applying NMPC to wind turbine control. One of the first studies on the application of NMPC to wind turbine control was presented in [9], where the author pointed out the potential benefits of NMPC during an extreme operating gust. Santos [10] and Koerber at al [11] showed reduced loads when including the knowledge of future wind speeds, obtained with e.g. a LIDAR, into the NMPC algorithm. Koerber and Henriksen [12] compare NMPC to classical wind turbine control and gain-scheduled MPC and concluded that performance improved only slightly when using NMPC. Henriksen stated that it would be hard to realise NMPC in practice due to excessive computational burdens; Koerber did not agree. Hence, literature on the application of NMPC to wind turbines is not only positive. None of the aforementioned papers tested NMPC on aero-elastic code, as used in the design of commercial wind turbines.

Therefore, the question that still remains, and which is the main topic of this paper, is whether there can be advantages in applying NMPC to wind turbines. In the firm belief that this question can be answered positively, a consortium of companies and one research institute started a project to develop and test advanced wind turbine control software, among which NMPC. This paper reports on the developments with NMPC within that project.

Structure of the paper
Section 2 presents the NMPC controller as developed by the consortium. Section 3 presents simulation results of this controller, tested in aero-elastic code; the NMPC controller is compared to two alternatives. Section 4 presents the conclusions.

2. The NMPC controller

Figure 1 shows a generic block-diagram of the NMPC controller that is used within the project. It consists of various blocks, each of which will be described in detail in the following subsections.

2.1 The internal model and cost function

Referring to Figure 1, the Internal Model provides the NMPC controller with the response of all relevant outputs, when called upon. The relevant outputs are those that the NMPC controller must optimize and/or limit. In this project, NMPC:

- maintains the generator speed (x_1) at target and below the overspeed limit,
- maintains generated power output at target,
- minimises tower top fore-aft motions and maintains tower top position within limits.

The possibility of handling (input and) output limits is one of the major advantages of (N)MPC over conventional control. An upper limit on generator speed allows us to reduce the aggressiveness of the pitch actuator to act on generator speed deviations. Maintaining generator speed tightly at target is in principle not necessary; the only reason that, in conventional control, generator speed control is often tuned tightly, is to avoid overspeed. With (N)MPC, we can avoid overspeed directly, by setting a limit on the generator speed. By reducing the activity of the NMPC controller to generator speed deviations, more pitch speed is available for load reducing control activities, such as tower top motion damping, and Individual Pitch Control (IPC).

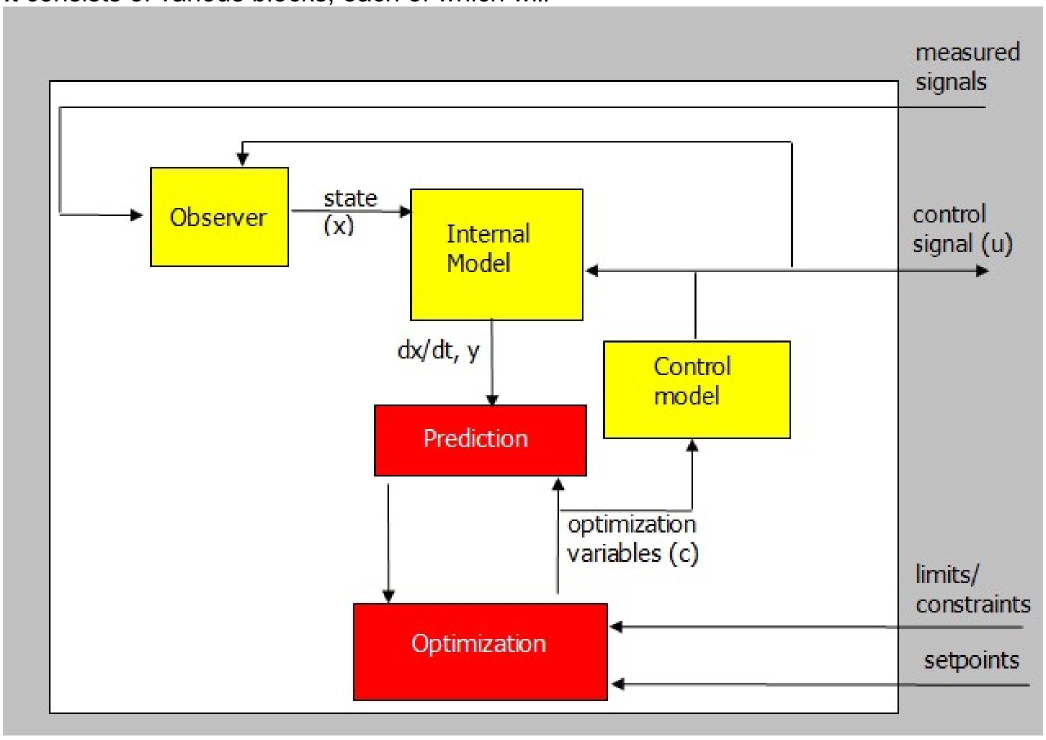

Figure 1 Control scheme of NMPC

The reason for setting a limit on the tower top position is as follows. Assuming a simple linear relation between tower top position and tower bottom load, the limits on tower top position form a "translation" of the requirement "maintain tower bottom load within limits". By tightening these limits, we can reduce the ultimate tower bottom loads in an efficient way. Reducing this load is, for some wind turbines (mainly on-shore), important, since that load is design driving for the tower.

Because of the aforementioned control objectives, an Internal Model was created that could simulate generator speed and tower top fore-aft motions. The dynamics were based on the control-oriented model that was presented in [1]:

$$\begin{aligned}
\frac{dx_1}{dt} &= \frac{i_{tr}}{J_e}\left(T_a(u_1,V) - i_{tr}x_2\right) \\
\frac{dx_2}{dt} &= \frac{1}{\tau_g}\left(u_2 - x_2\right) \\
\frac{dx_3}{dt} &= \frac{1}{m_t}\left(F_{ax}(u_1,V) - d_t x_3 - c_t x_4\right) \\
\frac{dx_4}{dt} &= x_3
\end{aligned} \tag{1}$$

with

x_1 = generator speed (rad/s), x_2 = generator torque (Nm), x_3 = tower top speed in fore-aft direction (m/s), x_4 = tower top position in fore-aft direction (m), u_1 = pitch angle demand (rad), u_2 = generator torque demand (N), i_{tr} = drive train transmission ratio (-), T_a = aerodynamic torque (Nm), J_e= effective rotor&generator inertia (kg m^2), τ_g = time constant (s), m_t = effective tower top mass (kg), d_t = damping coefficient (kg/s), c_t = stiffness (N/m), F_{ax} = thrust force (N). The generator power (P_g), aerodynamic torque and the thrust force are given by:

$$\begin{aligned}
P_g &= x_2 x_1 \\
T_a &= R^3 \frac{1}{2}\rho\pi V^2 C_q(\lambda,u_1) \\
F_{ax} &= \frac{1}{2}\rho\pi R^2 V^2 C_t(\lambda,u_1) \\
\lambda &= \frac{x_1 R}{i_{tr} V}
\end{aligned} \tag{2}$$

with *R* the rotor radius, ρ = density of air, and C_q and C_t the torque and thrust coefficient, respectively. On-line, these coefficients are not computed from a lookup-table, but they are approximated by bi-cubic splines; this allows fast computation of these coefficients, and avoids non-smooth partial derivatives.

At every sample period, NMPC minimises a cost function *J*, subject to constraints. In this paper, the cost function *J* is given by:

$$J = \left(\sum_{k=0}^{N-1} y(t_k)^T Q\, y(t_k) + \Delta u(t_k)^T R \Delta\, u(t_k)\right) + x(t_N)^T P\, x(t_N)$$

with *y* the cost vector, t_k = time at discrete sample intervals k (0, 1, 2, ...), Q, R = positive definite weight matrices, *P* = end weight matrix, $\Delta u(t_k) = u(t_k) - u(t_{k-1})$ =change in control input, x = state vector.

The cost vector y is given by:

$$y = \begin{bmatrix} x_1 - x_{1s} \\ P_g - P_{gs} \\ x_3 \\ s(x_1) \\ s(x_4) \end{bmatrix} \tag{3}$$

with s(.) a soft limit function. The subscript s refers to the setpoint value.

The soft limit function s(.) is a function that maps real inputs smoothly onto real positive outputs. It is chosen such it outputs 0 if the input is within pre-defined limits, but increases with increasing excess of limits. Figure 2 shows an example for $s(x_1)$ with x_{1max} the overspeed limit.

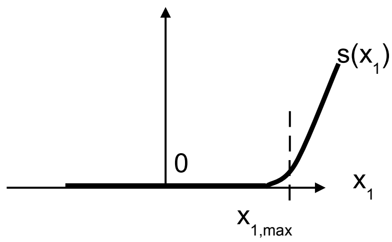

Figure 2 Example of soft limit function

The setpoint values for generator speed (x_{1s}) and power (P_{gs}) depend on the mode of the controller. In normal power production mode they are chosen as the (off-line pre-calculated) optimal steady state values, which depend on the wind speed V. However, during other operating conditions, the setpoints are chosen differently. For instance, during a shut down, the setpoints are ramped down.

The end weight matrix P in the cost function J is based on the idea that $x(t_N)^T P\, x(t_N)$ should form an upper bound on the 'tail' (J_t) of the cost function with J_t given by:

$$J_t = \left(\sum_{k=N}^{\infty} y(t_k)^T Q y(t_k) + \Delta u(t_k)^T R \Delta u(t_k)\right) \tag{4}$$

If it is (an upper bound), then it can be shown that *J* acts a Lyapunov function and stability is guaranteed, see e.g. [13], [14]. We compute *P* by minimising J_t subject to the linearised model given by equations (1) – (3) (linearised around the desired steady state). This computation (of P) requires the solution of Ricatti equations.

Hence, the matrix P is recomputed on-line, every sample period, and depends on the desired steady state.

The weight matrices Q and R are diagonal matrices, where the elements on the diagonal are used to tune the controller. These weights directly affect the deviations of the cost vector (y).

The minimisation of J is subject to absolute and rate constraints on the input u.

2.2 Observer

Not all states and inputs of the Internal Model can be measured on-line. Only the generator speed (x_1) and the tower top acceleration (dx_3/dt) are measured on-line, but the wind speed V, the tower top position (x_4) and velocity (x_3) must be estimated on-line. This is done by the Observer (referring to Figure 1). The Observer that we use is almost similar to the one described in [1], except that we use a Nonlinear Luenberger estimator to find the wind speed V, similar to the one presented in [2]. As a result, the Observer has guaranteed convergence at an adjustable exponential speed.

As explained in [13], the use of observers with exponential convergence allows obtaining guaranteed stability and reliability of the NMPC controller.

2.3 Control Model

The Control Model (referring to Figure 1) maps the optimization variables c to the control signal u. One of the most widely used mappings in MPC is where each element of c defines the control input at regular time intervals $t_0, t_1, \ldots,$ for instance:
$u_1(t_0) = c_0$, $u_1(t_1) = c_1$, etc.
In between the time points, the control input is assumed to remain constant.
The disadvantage of this parameterisation is that it requires a large amount of optimization variables that slow down the calculations. Instead, we use a more efficient parameterisation, according to:

$$u_1(t) = \sum_{n=0}^{N-1} c_n e^{-a_n t} + c_N t$$
$$u_2(t) = \sum_{m=N+1}^{M+N} c_m e^{-a_m t} \tag{5}$$

with N and M fixed integers and $a_1, a_2, \ldots, a_{M+N}$ fixed constants which are selected in such a way that the control profiles can form a close approximation to those of the closed loop optimal LQR (in the absence of constraints). Note that u_1, the pitch demand, has the ability to pitch to feather at maximum speed, due to the term $c_N t$.

2.4 Prediction

Whenever called by the Optimisation module, the Prediction module simulates the Internal Model forward, using the state of the Observer as the initial state. During normal operation, the wind speed prediction is assumed to remain constant in future. However, if better wind speed predictions would be available (from e.g. a wind gust detector as presented in [2], or LIDAR), they can be incorporated directly.

2.5 Optimization

Each sample period, the Optimization module (referring to Figure 1) searches the vector of optimization variables (c) that minimises the cost function J, subject to constraints.

To solve the NMPC optimization problem, we use the method as presented in [13] and [14]: each sample period a Sequential Quadratic Program (SQP) is solved which is set up in such a way that it allows to break-off the optimization anytime after at least one sequential iteration, and still ensures that the cost J reduces at each sample period. This method ensures (1) stability of the closed-loop in the nominal case (when the model matches the system to be controlled perfectly), and (2) a guaranteed upper limit on the optimization time .
The QP problem, solved in each SQP iteration, is solved by the Primal/Dual Interior Point method, as presented in [15].

The NMPC algorithm, presented so far, works well when tested on the Internal Model, and in the absence of 'noise' in the sensor signals. However, without any modifications, it did not perform well in aero-elastic simulations (and therefore it would probably also not work in practice). In the next section, we explain how we resolved this problem.

2.6 Adjustments required in practice

In order to be able to apply any wind turbine controller in practice, there must be ways to:

- filter the measured signals before they enter the controller
- robustify the controller, i.e. ensure stability over the full operating range

The possibility to filter measured signals is an absolute requirement in wind turbine control, since many of these signals contain 'periodic noise' that needs be removed before entering the controller. If this is not done, the pitch speed is mostly consumed by responding to “periodic noise” and there is no pitch speed left for the essential control tasks, such as generator speed control, active tower damping and IPC. With conventional control, adding (notch) filters is easy; it only requires some retuning of the PID controllers due to phase lag changes by the filter(s).

However, with model based control concepts, such as (N)MPC, this is, in principle, not easy at all. Unlike PID control, there is no simple “button” to adjust. The simplest approach would be to include filtering of the measured signals, and ‘detune’ the NMPC controller, by adjusting the weight matrices Q and R. However, this rude method may only be satisfactory in case of ‘mild’ filtering. A better approach might be to integrate filters into the Observer (design). This approach is possible if observers are used that allow to take measurement noise into account. An example is the (Extended) Kalman observer. However, even then, it is relatively complicated to realise the required (notch) filtering. Moreover, obtaining closed-loop stability of the NMPC algorithm over the full operating range can be challenging, certainly if if the Extended Kalman filter is used [13].

Robust stability is easy to obtain with conventional PID control, but many advanced model based controllers, including the NMPC controller described in this paper, are based on the assumption of “no model mismatch”, i.e. they assume that the system to be controlled behaves exactly the same as the model used in control design. There are robust NMPC algorithms, such as open-loop min-max NMPC [17], and the H_∞-NMPC [16], in which model mismatch can be taken into account explicitly. However, we fear that, currently, these method are computationally (too) demanding for wind turbine applications.

In this work, we have used a novel method that allows us to resolve the two aforementioned issues in a relatively simple way and without additional computational costs. This method allows us:

- to freely filter sensor signals without causing stability problems
- to robustify a model based controller, i.e. make it robust stable over the entire operating range.

Currently, a patent application is being written on this novel method, and therefore, we regrettably cannot disclose the details of this method at this moment in time. We can only mention that we used this method to achieve the results as presented in section 3.

Combination of NMPC with conventional control loops

The NMPC controller in this project does not perform individual pitch control (IPC). There are two ‘remedies’ for this:

1. extend the Internal Model and the cost vector so that NMPC does act as an IPC controller
2. combine the NMPC controller, as presented in this paper, with conventional IPC loops, such as described in e.g. [1].

A good reason to apply the first option would be that the resulting NMPC would be able to distribute (limited) pitch speed control optimally, and, NMPC could reduce extreme loads during, for instance, a stuck blade event.

At the time of writing, option 1 had not yet been tested, but option 2 had, and worked well.

Code

All NMPC code has been programmed in both Matlab code and ANSI C. The C code is compiled into a DLL that is called by PHATAS when needed. Both source codes can be obtained from DotX.

3. Case study

The NMPC controller, as described in section 2, has been simulated on the aero-elastic PHATAS model of the 2MW XEMC-Darwind XV90 wind turbine. We simulated a selection of the full load set that was used in the design of this machine; this selection was established by XEMC-Darwind and includes Design Load Case (DLC) 1.2 and 1.5, as defined in accordance with IEC 61400-1 edition 3. Three controllers were evaluated: (1) Conventional control (as used in the original design of the wind turbine), (2) the Improved Baseline controller, as presented in [5], and (3) the NMPC controller as presented in this paper. The Conventional controller uses (gain scheduled) PID controllers, extended with linear filters, to control generator speed, power and tower fore-aft damping. The Improved Baseline controller also uses (gain scheduled) PID controller with linear filters, but has improved tuning methods, and more feedback loops (among which tower side-to-side damping and IPC).

Table 1 summarises the results. We have compared NMPC to the Conventional controller (as used in the original design of the wind turbine) and to the Improved Baseline controller, as presented in [5].

Controller	DEL (MyTB), DLC 1.2	FL (MyTB), DLC 1.2 for $V > V_{rat}$	Power , DLC 1.2	$\|MyTB\|_{max}$, DLC 1.5	Computational Load
Conventional	100	100	100	100	100
Improved Baseline [5]	90	87	102	90	100
NMPC	98	81	106	74	140

DEL = Design Equivalent fatigue Load, FL = Fatigue Load, MyTB = tower bottom moment in fore-aft direction, V_{rat} = rated wind speed.

Table 1 Summary of results in % (Conventional = 100%)

The results indicate that NMPC is able to produce more power while maintaining the design equivalent fatigue load on the tower bottom slightly lower than that of the Conventional controller. Furthermore, NMPC is able to reduce the ultimate tower bottom bending moment. The Computational Load, i.e. the computation time that NMPC needs has only increased by 40%, and therefore, it should be possible to run NMPC in standard wind turbine computers.

We shall now clarify these results.

Figure 3 shows simulation results of all three controllers in one of the DLC 1.2 load cases. Clearly, the generated power in partial load conditions (for t > 435 s) is highest with NMPC. This is mainly due to the fact that both the Conventional controller and Baseline controller use a fine pitch schedule, i.e. the minimum pitch angle is scheduled with generated power; this technique was probably used in the Conventional controller to reduce some critical load, or to avoid overspeed. The Baseline controller has adopted the same schedule to allow a better comparison with the Conventional controller, but this schedule is not used in NMPC. Applying a fine pitch schedule has a reducing effect on the tower bottom fatigue load, but NMPC is able to better reduce tower fore-aft motions, thanks to its model based approach, and therefore, the DEFL of the tower bottom load has not increased (when compared to the Conventional controller).

The Improved Baseline controller reduces the DEFL of the tower bottom load more, but at the price of reduced power output (when compared with NMPC). In above rated conditions, the fatigue loads with NMPC are the lowest of all controllers (see Table 1).

The ultimate load is reduced by 26% when using NMPC. To understand how this is possible, figure 4 shows the simulation results of NMPC and the Conventional controller (the results of the Baseline controller are similar to that of the Conventional one and are therefore left out). The upper graph of figure 4 shows how the wind speed (uniform wind) varies during simulation. At t= 12 seconds, grid loss occurs. As soon as the upper tower bottom load limit is predicted to be exceeded, which is before grid loss, NMPC pitches to feather to avoid this ‘as much as possible’. After grid loss, it computes pitch actions that avoid a violation of the negative tower bottom load limit.

Figure 5 demonstrates how the soft limits in NMPC contribute to this result. The blue lines show the (same) simulation with NMPC, but in this case without soft limits (hence, no limits on generator speed, and tower bottom load), while the red lines show the simulation when NMPC with soft limits is used. In the absence of soft limits (blue lines), NMPC responds only after after grid loss by pitching to feather at maximum pitch speed (after grid loss, the setpoints for generator speed and power are immediately ramped down). When NMPC with soft limits is used, NMPC responds earlier.

4. Conclusions

The main question of this paper was whether there can be advantages in applying NMPC, instead of state-of-the art controllers.

In order to be able to answer this question, an NMPC controller, that is applicable in practice, had to be developed. Such a controller must overcome the issues of computational speed limits, robust stability requirements and the possibility to filter sensor signals (before they enter the controller). All of these issues have been tackled. Subsequently, we have tested the controller on aero-elastic code and compared its performance with two other controllers. Thereby, we focussed on power output maximisation and tower bottom load minimisation.

The results indicate that NMPC is better able to maximise power output, while minimising fatigue and ultimate loads of the tower bottom load. This is due to mainly two factors:

1. NMPC is better able to reduce fatigue loads (thanks to its model based approach)
2. NMPC can handle both hard and soft limits; ultimate loads can be reduced by tightening the soft limits on those loads

In future work we intend to test NMPC on a prototype wind turbine, and to apply it in the design of a commercial wind turbine.

Acknowledgement

The work described in this paper was executed in the project "Development of new control method for wind turbines", under grant KTOT02056.

The Dutch Ministry of Economic affairs (Agentschap NL) is greatly acknowledged for their financial support to this project.

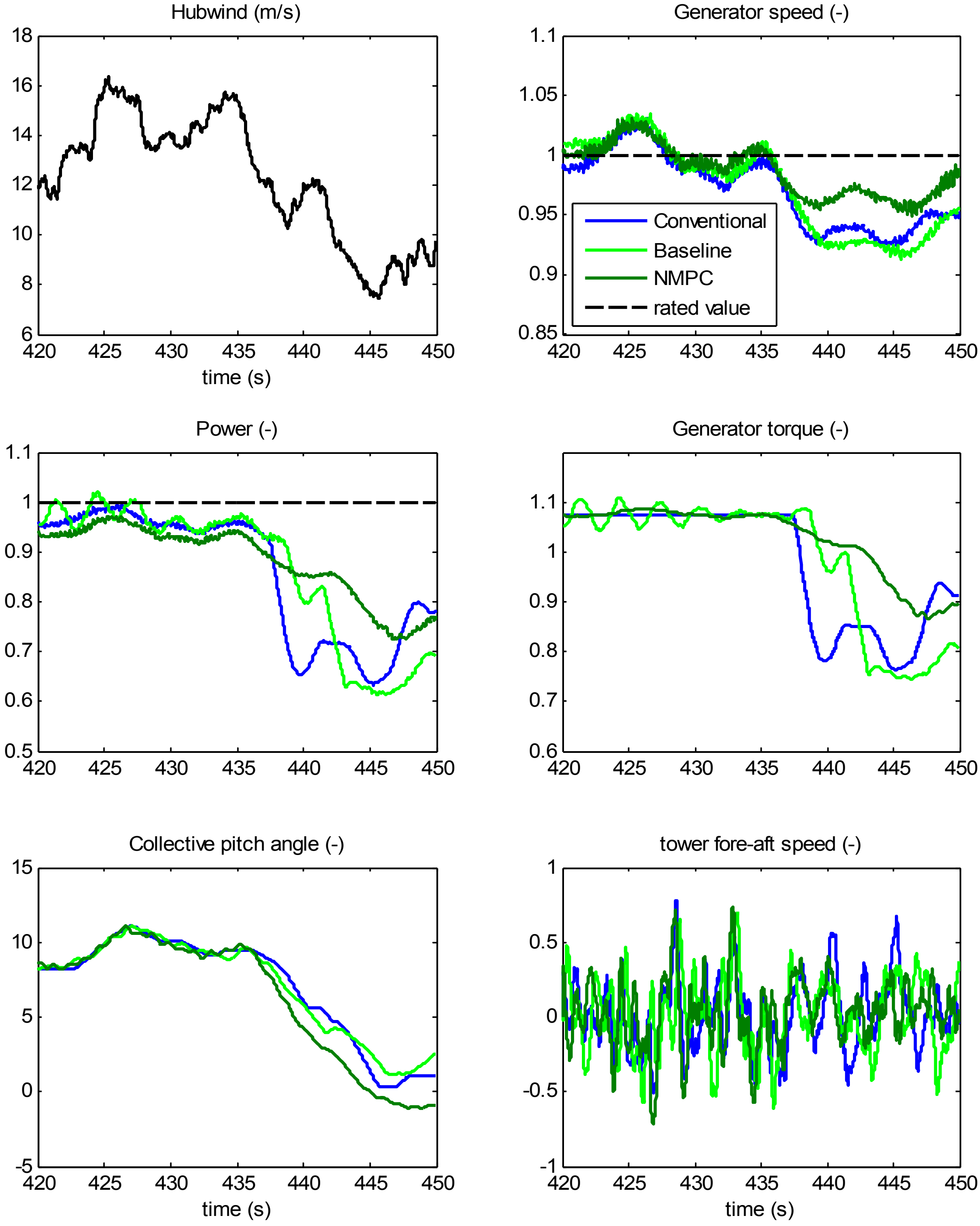

Figure 3 Simulation results for DLC 1.2 with three different controllers.

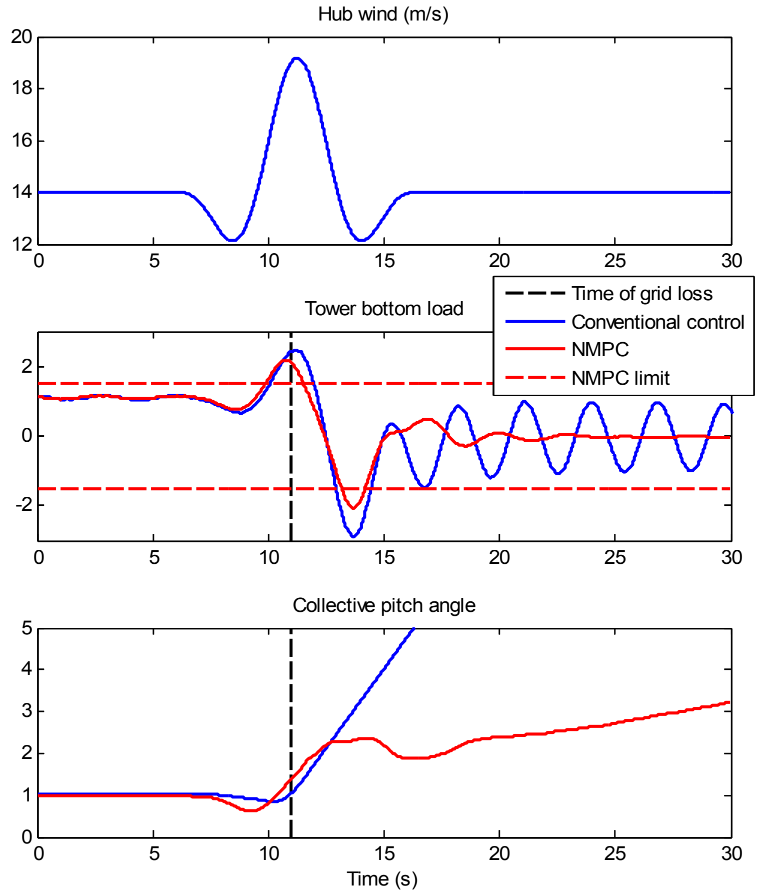

Figure 4 PHATAS simulation results of a DLC 1.5 load case with two different controllers (Conventional versus NMPC). All data, except for the hub wind, has been normalised.

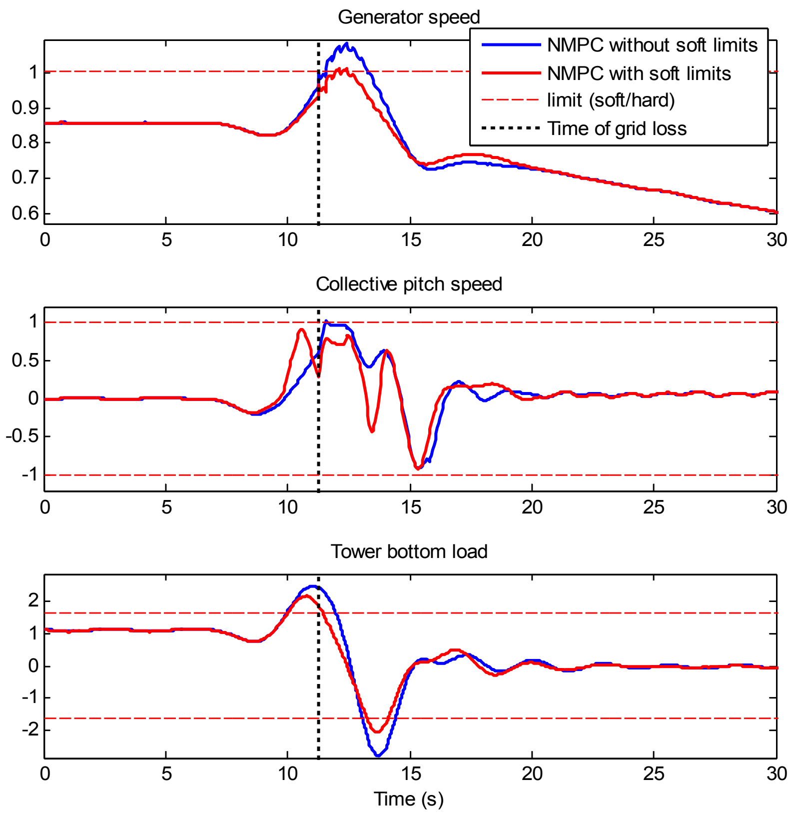

Figure 5

References

[1] Kanev, S.; Schuurmans, J.; Rutteman, R. & Nguyen, E. 'Towards new industrial software for advanced wind turbine control', Proceedings of the EWEA conference, 2011

[2] E. Nederkoorn et al, 'Increased Aerodynamic Performance of Wind Turbines Through Improved Wind Gust Detection, and Extreme Event Control', EWEA 2013.

[3] K. Selvam, S. Kanev, J. W. van Wingerden, T. van Engelen, and M. Verhaegen, 'Feedback/feedforward individual pitch control for wind turbine load reduction', Int. J. Robust Nonlinear Control, 19(1):72{91}, 2009

[4] Karl A. Stol and Mark J. Balas, ' Periodic disturbance accommodating control for blade load mitigation in wind turbines'. J Sol Energy Eng Trans ASME, 125(4):379{385}, 2003.

[5] F. D. Bianchi, R. J. Mantz, and C. F. Christiansen, 'Gain scheduling control of variable-speed wind energy conversion systems using quasi-lpv models.', Control Engineering Practice, 13(2):247 {255}, 2005.

[6] K.Z. Østergaard, J. Stoustrup, and P. Brath, 'Linear parameter varying control of wind turbines covering both partial load and full load conditions.', International Journal of Robust and Nonlinear Control, 19(1):92{116}, 2009.

[7] A. Kumar and K. Stol., 'Simulating feedback linearization control of wind turbines using high-order models.' , Wind Energy 13(5):419{432}, 2010

[8] B. Boukhezzar, H. Siguerdidjane, and M. Maureen Hand, 'Nonlinear control of variable-speed wind turbines for generator torque limiting and power', optimization. Journal of Solar Energy Engineering, 128(4), 2006.

[9] C.L. Bottasso, A. Croce, B. Savini, W. Sirchi, L. Trainelli, 'Aeroelastic modeling and control of wind turbine generators using finite element multibody procedures', multibody dynamics 2005, eccomas Thematic Conference, Madrid, Spain, 2005

[10] R. Santos. 'Damage mitigating control for wind turbines.', PhD thesis, University of Colorado at Boulder, United States – Colorado, 2007.

[11] Arne Koerber, Rudibert King, 'Nonlinear Model Predictive Control for Wind Turbines', EWEA 2011

[12] L. C. Henriksen. Model predictive control of a wind turbine. Master's thesis, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2007.

[13] R. Findeisen, L. Imsland, F. Allgower, B.A. Foss, 'Stability Conditions for Observer Based Output Feedback Stabilization with Nonlinear Model Predictive Control', Proceedings of the 42nd IEEE Conference on Decision and Control Maui, Hawaii USA, December 2003

[14] B. Kouvaritakis, M. Cannon, J. A. Rossiter, 'Non-linear model based predictive control', INT. J. CONTROL, 1999, VOL. 72, NO. 10, 919- 928

[15] J. Nocedal & S. Wright, 'Numerical Optimization', Springer Science+Business Media, LLC. 2nd edition, page 480-485, 2006

[16] L. Magni, H. Nijmeijer, and A.J. van der Schaft, 'A receding-horizon approach to the nonlinear H_∞ control problem.', A*utomatica*, 37(5):429–435, 2001.

[17] H. Chen, C.W. Scherer, and F. Allgöwer, 'A game theoretic approach to nonlinear robust receding horizon control of constrained systems.', *Proc. Amer. Contr. Conf.*, pages 3073–3077, Albuquerque, 1997.