Technical report

Supervisory nonlinear MPC for emergency voltage control using pattern search

R.R. Negenborn, S. Leirens, B. De Schutter, J. Hellendoorn

If you want to cite this report, please use the following reference instead:

R.R. Negenborn, S. Leirens, B. De Schutter, J. Hellendoorn. Supervisory nonlinear MPC for emergency voltage control using pattern search. *Control Engineering Practice*, vol. 17, no. 7, pp. 841-848, July 2009.

Delft University of Technology, Delft, The Netherlands

Supervisory nonlinear MPC for emergency voltage control using pattern search

R.R. Negenborn[a,*], S. Leirens[b], B. De Schutter[a,1], J. Hellendoorn[a]

[a] *Delft University of Technology, Delft Center for Systems and Control, Mekelweg 2, 2628 CD Delft, The Netherlands*
[b] *Universidad de Los Andes, Departamento de Ingenería Eléctrica y Electrónica, Carrera 1a Este #18A-70, Bogotá D.C., Colombia*

Abstract

The design of a higher-layer controller using model predictive control (MPC) is considered. The higher-layer controller uses MPC to determine set-points for controllers in a lower control layer. In this paper the use of an object-oriented model of the system for making predictions is proposed. When employing such an object-oriented prediction model the MPC problem is a nonlinear, non-smooth optimization problem, with an objective function that is expensive to evaluate. Multi-start pattern search is proposed as approach to solving this problem, since it deals effectively with the local minima and the non-smoothness of the problem, and does not require expensive estimation of derivatives. Experiments in an emergency voltage control problem on a 9-bus dynamic power network show the superior performance of the proposed multi-start pattern search approach when compared to a gradient-based approach.

Keywords: Power networks, supervisory control, model predictive control, pattern search, object-oriented modeling, voltage control.

---

1. Introduction

1.1. Power network control

General power networks are controlled in a hierarchical way in which control of the physical network is the result of several control layers at local, regional, national, and sometimes international level (Fardanesh, 2002; Hill et al., 2003). As illustrated in Figure 1, at the physical level the power network consists of multiple interconnected subsystems, like generators, loads, transmission lines, etc. This physical level is controlled by one or more control layers in order to control the network in some desired way. The lowest control layer typically consists of decentralized controllers that independently of each other control the actuators in the physical system directly. Higher

*Corresponding author. Tel.: +31 1527 86524; fax: +31 1527 86679.
Email address: `r.r.negenborn@tudelft.nl` (R.R. Negenborn)

[1] Bart De Schutter is also with the Marine and Transport Technology department of the Delft University of Technology.

Preprint submitted to Control Engineering Practice

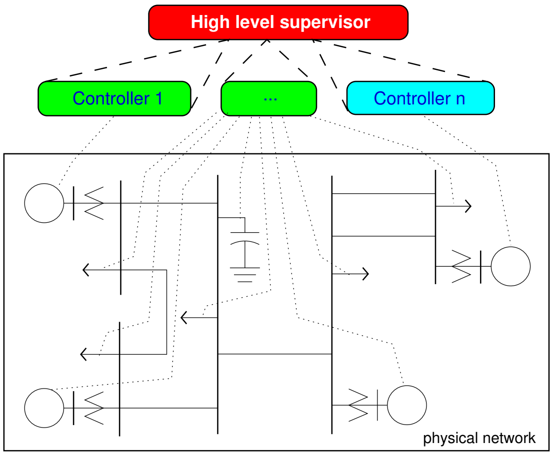

Figure 1: Multi-layer control of a power network: higher layers provide set-points to lower layers and the lowest layer controls the physical system directly. Dashed and dotted lines indicate control signals going from a higher to a lower layer (see section 5.1).

control layers consist of supervisory controllers that determine set-points for lower control layers to obtain coordination. The set-points have to be determined in such a way that some objectives defined for the full system are achieved (Mesarovic et al., 1970). The controllers in higher layers hereby typically take into account nonlinear behavior of the system, behavior that may be neglected by the lower control layers.

The particular task of the higher-layer controller considered in this work is to prevent voltage collapses from appearing in power networks. Briefly stated, voltage instability stems from the attempt of the load dynamics to restore the power consumption beyond the capability of the transmission and generation system (Van Cutsem and Vournas, 1998). Typically, this situation occurs after the outage of one or more components in the network, such that the system cannot satisfy the load demand with the given inputs at a physically sustainable voltage profile.

1.2. The higher layer

In current power network operation higher-layer control is typically done by human operators that determine the set-points for lower layers, using offline studies, experience, system conditions observed via telemetry, heuristics, knowledge bases, and state-estimator outputs. As it becomes more complex for humans to accurately predict the consequences of faults in the network due to deregulation of the energy market, the increase in power demands, and the emergence of embedded generation (Jenkins et al., 2000), intelligent automatic online control systems become increasingly necessary. Such an approach for online control is proposed in this paper.

To adequately steer the lower control layer, a higher-layer supervisor has to monitor the state of the lower control layer and the underlying physical system and foresee when the behavior of the system is going into an undesirable direction. The supervisor proposed in this paper uses model predictive control (MPC) (Maciejowski, 2002).

MPC has traditionally been employed in the process industry, and is now gaining increasing attention in fields like amongst others power networks (Geyer et al., 2003), railway networks (De Schutter et al., 2002), steam networks (Majanne, 2005), greenhouse control (Piñón et al., 2005), and drug delivery (Bleris et al., 2007). In MPC control actions are obtained at each control sample step by solving an optimization problem that minimizes an objective function over a finite horizon subject to a prediction model and operational constraints. The main advantages of MPC are its explicit way of integrating soft as well as hard constraints, its easy way of integrating forecasts, and its straightforward design procedure.

The complexity of power networks makes developing appropriate prediction models a complex task. The number of elements in large-scale power networks is huge. Moreover, the dynamics of power systems are hybrid, in the sense that the dynamics are the result of interactions between continuous dynamics (e.g., due to loads and generators) and discrete events (due to, e.g., saturation in generators or discrete switching of transformer taps, capacitor banks, or loads) (Leirens et al., 2005). Over the last decade modeling languages and simulation environments have been introduced that allow general-purpose physical modeling based on non-causal modeling, with true equations and the use of object-oriented constructs (Mattsson et al., 1998; Piela et al., 1991; Barton and Pantelides, 1994; Dynasim, 2004), therewith easing the development of such complex prediction models.

1.3. Proposed control approach

In (Larsson and Karlsson, 2003; Hill et al., 2003) a tree search has been proposed to solve a nonlinear MPC problem for coordinating controllers in a power network with dynamics in loads and generators. A discrete time prediction model has been used with quantified small variations of the control inputs, leading to a purely combinatorial optimization problem. In (Geyer et al., 2003; Negenborn et al., 2007) approaches have been proposed using mixed-integer linear programming to solve optimization problems derived from the nonlinear MPC problem. The approach proposed in this paper uses an object-oriented model as prediction model and multi-start pattern search as solver of the nonlinear MPC problem.

Several authors have considered approaches for nonlinear MPC, e.g., (Oliveira and Biegler, 1995; Chen and Allgöwer, 1998; Diehl et al., 2002; Martinsen et al., 2004). These approaches each make particular assumptions on the type of prediction model (e.g., systems of differential equations, smooth transition functions) and assume explicit access to the equations of the prediction model. Efficient control strategies result from exploiting the structure of the nonlinear MPC programming problems. In this paper, a multi-start pattern search approach is proposed that does not rely on the explicit knowledge of the model equations. Recently, a pattern-search method has been applied to power systems in (Al-Othman and El-Naggar, 2007) to solve an economic dispatch problem. However, there, no network dynamics and MPC strategy are considered.

So, here, a higher layer MPC controller is proposed with the following aspects:

- the prediction model that the higher layer is object-oriented;
- the prediction model is a full nonlinear, non-smooth model of the lower control layer and physical network;

- the optimization technique used to solve the optimization problem is the direct-optimization method pattern search.

Noted that the proposed approach is not restricted to the voltage collapse problem, and can easily be adapted to other control problems, e.g., minimization of power losses, automatic generation control, etc. Also, the approach may be suitable for higher-layer control in other domains, such as water and road traffic networks.

1.4. Outline of this paper

In Section 2 object-oriented power network modeling and the assumptions regarding the prediction model are discussed. In Section 3 the supervisory control problem is formulated in a nonlinear MPC fashion. In Section 4 direct-search versus gradient-based optimization methods are discussed, and a multi-start implementation of the direct-search method pattern search is presented. In Section 5 the direct-search approach is compared with a gradient-based approach and performance is assessed by means of simulation studies on a voltage control problem.

2. Object-oriented prediction models

2.1. Object-oriented modeling

To face the difficulty of developing complex power network models, object-oriented approaches for analysis and simulation have received increasing attention (Manzoni et al., 1999). In object-oriented modeling, models are mapped as closely as possible to the corresponding physical subsystems that make up the overall system. Models are described in a declarative way, i.e., only local equations of the objects and the connections between the objects are defined. Objected-oriented modeling concepts, such as *inheritance* and *composition* enable proper structuring of models and generally lead to more flexible, modular, and reusable models. Inheritance is a way to form new classes of models using classes that have already been defined. The new classes take over or inherit attributes and behavior, e.g., dynamics, of the already existing classes. Extended models can then be constructed by inheriting dynamics and properties of more basic or more general models. E.g., advanced generator models are designed in this way by extending a basic generator model containing the basic dynamics of a synchronous machine only. Composition is a way to combine simple models into more complex ones. E.g., a voltage regulator and a turbine governor do not have functionality by themselves. However, composing them with a basic generator model results in a regulated generator with complex dynamics.

As stated above the dynamics of power networks involve continuous and discrete dynamics and are therefore hybrid. Each of the objects of a power network can therefore be modeled with a mixture of differential equations, algebraic equations, and discrete-event logic, e.g., in the form of if-then-else rules. The model for the overall system then consists of the models for the objects and in addition algebraic equations interconnecting the individual objects.

2.2. *Modeling tools*

Several object-oriented approaches have been developed over the years, e.g., (Piela et al., 1991; Barton and Pantelides, 1994; The Modelica Association, 2005; Mattsson et al., 1998; Dynasim, 2004). These approaches typically support both high-level modeling by composition and detailed component modeling using equations. Models of system components are typically organized in model libraries. A component model may be a composite model to support hierarchical modeling and specify the system topology in terms of components and connections between them. Using a graphical model editor, e.g., Dymola (Dynasim, 2004), a model can be defined by drawing a composition diagram by positioning icons that represent the models of the components, drawing connections, and giving parameter values in dialog boxes.

It should be noted that some of the object-oriented simulation software packages, e.g., Simulink, assume that a system can be decomposed into sub-models with fixed causal interactions. This means that the models can be expressed as the interconnection of sub-models with an explicit state-space form. Often a significant effort in terms of analysis and analytical transformations is required to obtain a model in this form (Dynasim, 2004). In general, causality is not assigned in power networks. Setting the causality of an element of the power network, e.g., a transmission line, involves representing the model equations in an explicit input-output form. In a voltage-current formulation this means that currents are expressed as function of voltages, or vice versa. Non-causal modeling permits to relax the causality constraint and allows to focus on the elements and the way these elements are connected to each other, i.e., the system's topology. An environment that allows non-causal modeling, and that is used in this work, is Dymola (Dynasim, 2004), which implements the object-oriented modeling language Modelica. See (Navarro et al., 2000) for an example of non-causal and object-oriented power system modeling.

2.3. *Prediction model assumptions*

For the object-oriented model to be useful as a prediction model, a method has to be available that can evaluate the model over a certain time horizon from time t_0 until t_f. This means that an initial value problem has to be solved that given the initial continuous state $\mathbf{x}_\mathrm{c}(t_0) \in \mathbb{R}^{n_{x_\mathrm{c}}}$, the initial discrete state $\mathbf{x}_\mathrm{d}(t_0) \in \mathbb{Z}^{n_{x_\mathrm{d}}}$, the initial derivative of the continuous state $\dot{\mathbf{x}}_\mathrm{c}(t_0)$, the initial algebraic variables $\mathbf{z}(t_0) \in \mathbb{R}^{n_z}$, and inputs $\mathbf{u}(t)$ specified over the full prediction interval, computes the output $\mathbf{y}(t) \in \mathbb{R}^{n_y}$, for $t \in [t_0, t_\mathrm{f}]$. For power networks, $\mathbf{x}_\mathrm{c}$ typically are state variables of generators, loads, etc., $\mathbf{x}_\mathrm{d}$ are transformer tap settings or status information like whether or not a saturation point has been reached, $\mathbf{z}$ represent voltage magnitudes, voltage angles, and auxiliary algebraic variables used in modeling physical phenomena in generators, $\mathbf{y}$ are the outputs, e.g., voltage magnitudes, power losses, etc., and $\mathbf{u}$ are actuator inputs for generators, turbine governors, loads, etc.

Note that the set-points cannot be continuously provided, but only at discrete control sample steps $k_\mathrm{c} + i$, for $i = \{0, 1, \ldots\}$, where control sample step $k_\mathrm{c} + i$ corresponds to continuous time $t_0 + (k_\mathrm{c} + i)T_\mathrm{c}$, with T_c the distance between two control sample steps in continuous time units. The controller uses a zero-order hold approach to make the transformation between the discrete-time input signal and the continuous-time input signal. So $\mathbf{u}(t) = \mathbf{u}_{k_\mathrm{c}}$, for $t \in [t_0 + k_\mathrm{c}T_\mathrm{c}, t_0 + (k_\mathrm{c} + 1)T_\mathrm{c})$. Therefore, instead of specifying a continuous input signal, the prediction model is given a sequence of N_c inputs, collected in $\tilde{\mathbf{u}}(k_\mathrm{c}) = [\mathbf{u}(k_\mathrm{c})^\mathrm{T}, \ldots, \mathbf{u}(k_\mathrm{c}+N_\mathrm{c}-1)^\mathrm{T}]^\mathrm{T}$, where $N_\mathrm{c} = \left\lfloor \frac{t_\mathrm{f}-t_0}{T_\mathrm{c}} \right\rfloor + 1$ is the number of set-point updates over the prediction horizon[2].

In general, there is no analytic expression for the solution of the initial value problem. Instead, the trajectories of all variables of interest have to be approximated by numerical means to obtain values for the variables at discrete points in time. It is assumed that computing a sample of the output $\mathbf{y}$ once every T_p time units is sufficient to adequately represent the underlying continuous signals. Therefore the prediction horizon is defined as $N_\mathrm{p} = \left\lfloor \frac{t_\mathrm{f}-t_0}{T_\mathrm{p}} \right\rfloor + 1$, and the outputs as $\tilde{\mathbf{y}}(k_\mathrm{p}) = [\mathbf{y}(k_\mathrm{p})^\mathrm{T}, \ldots, \mathbf{y}(k_\mathrm{p}+N_\mathrm{p}-1)^\mathrm{T}]^\mathrm{T}$, where discrete time step $k_\mathrm{p}+i$ corresponds to continuous time $t_0 + (k_\mathrm{p}+i)\,T_\mathrm{p}$.

In the following it is assumed without loss of generality that the object-oriented model of the power network is given by the mapping

$$\tilde{\mathbf{y}}(k_\mathrm{p}) = \mathscr{P}(\dot{\mathbf{x}}_\mathrm{c}(t_0), \mathbf{x}_\mathrm{c}(t_0), \mathbf{x}_\mathrm{d}(t_0), \mathbf{z}(t_0), \tilde{\mathbf{u}}(k_\mathrm{c})), \tag{1}$$

where $\mathscr{P}$ maps the derivative of the continuous state $\dot{\mathbf{x}}_\mathrm{c}(t_0)$, the continuous state $\mathbf{x}_\mathrm{c}(t_0)$, the discrete state $\mathbf{x}_\mathrm{d}(t_0)$, the algebraic variables $\mathbf{z}(t_0)$ at time t_0, and the N_c inputs collected in $\tilde{\mathbf{u}}(k_\mathrm{c})$ to the N_p outputs collected in $\tilde{\mathbf{y}}(k_\mathrm{p})$. The prediction model thus includes the procedure to perform the time-domain simulation of the object-oriented model. In this paper, the prediction model is emulated by the differential-algebraic equations solver DASSL (Petzold, 1983; Brenan et al., 1996) as implemented in Dymola (Dynasim, 2004), which solves the initial-value problem of the system of differential-algebraic equations obtained after transforming the object-oriented model. It should be noted that the prediction model $\mathscr{P}$ is nonlinear and non-smooth, and involves the numerical solution of systems of differential-algebraic equations in combination with discrete logic, and that therefore computing the predictions is a computationally intensive process.

3. Supervisory voltage control problem

The supervisory controller uses MPC to determine which set-points to provide to the lower control layer. The MPC problem of the supervisory controller is formulated as follows. Every T_c time units the supervisory controller has to provide a set of set-points for the next T_c time units. These inputs have to be chosen in such a way that costs over a prediction horizon of N_c control sample steps, i.e., over a time span of $N_\mathrm{c}T_\mathrm{c}$ time units, are minimized.

The control objectives of the supervisory controller consist of providing set-points to the decentralized controllers in the lower layer such that: all voltage magnitudes are maintained as much as possible within certain lower and upper bounds; changes in set-points provided earlier are minimized; and acceptable steady-state voltage magnitudes are obtained (i.e., no oscillations are present). These control objectives are quantified

[2] $\lfloor v \rfloor$ indicates the largest integer value smaller than or equal to v.

through the following objective function:

$$J(\tilde{\mathbf{u}}(k_\mathrm{c}),\tilde{\mathbf{y}}(k_\mathrm{p})) = \sum_{i=0}^{N_\mathrm{p}-1} \|\mathbf{Q}_y \mathbf{y}_\mathrm{err}(\mathbf{y}(k_\mathrm{p}+i))\|_\infty + \|\mathbf{Q}_u(\mathbf{u}(k_\mathrm{c})-\bar{\mathbf{u}})\|_1 + \sum_{i=0}^{N_\mathrm{c}-1} \|\mathbf{Q}_u(\mathbf{u}(k_\mathrm{c}+i)-\mathbf{u}(k_\mathrm{c}+i-1))\|_1, \qquad (2)$$

where $\bar{\mathbf{u}}$ are the set-points provided at the last control sample step, $\mathbf{Q}_y$ and $\mathbf{Q}_u$ are penalty matrices, $\|\cdot\|_\infty$ and $\|\cdot\|_1$ denote the infinity norm and one norm, respectively, where $\mathbf{y}$ are the voltage magnitudes and $\mathbf{u}$ are the set-points for the lower control layer, and where $\mathbf{y}_\mathrm{err}$ is the violation of the desired voltage bounds, of which the entries are computed as

$$\mathbf{y}_{\mathrm{err},q}(\mathbf{y}(k_\mathrm{p}+i)) = \begin{cases} \mathbf{y}_{q,\mathrm{lower}} - \mathbf{y}_q(k_\mathrm{p}+i) & \text{for } \mathbf{y}_q(k_\mathrm{p}+i) \leq \mathbf{y}_{q,\mathrm{lower}} \\ 0 & \text{for } \mathbf{y}_{q,\mathrm{lower}} < \mathbf{y}_q(k_\mathrm{p}+i) < \mathbf{y}_{q,\mathrm{upper}}, \\ \mathbf{y}_q(k_\mathrm{p}+i) - \mathbf{y}_{q,\mathrm{upper}} & \text{for } \mathbf{y}_q(k_\mathrm{p}+i) \geq \mathbf{y}_{q,\mathrm{upper}} \end{cases}$$

where $\mathbf{v}_q$ indicates entry q of vector $\mathbf{v}$, and $\mathbf{y}_{q,\mathrm{lower}}$ and $\mathbf{y}_{q,\mathrm{upper}}$ are the desired upper and lower bounds for the corresponding voltage magnitudes. The infinity norm is taken for minimization of the variables $\mathbf{y}_\mathrm{err}$, such that the worst-case error is minimized (Geyer et al., 2003; Beccuti and Morari, 2006). Using the infinity norm is sufficient for representing the control objective of maintaining all voltage magnitudes between the given lower and upper bounds[3]. The one norm is used for the changes in the inputs $\mathbf{u}(k_\mathrm{c}+i)-\mathbf{u}(k_\mathrm{c}+i-1)$, such that the changes in each of the inputs are minimized.

Hence, the supervisory MPC control problem is formulated at control step k_c, corresponding to time $t=t_0$, as

$$\begin{aligned} \min_{\tilde{\mathbf{u}}(k_\mathrm{c}),\tilde{\mathbf{y}}(k_\mathrm{p})} \quad & J(\tilde{\mathbf{u}}(k_\mathrm{c}),\tilde{\mathbf{y}}(k_\mathrm{p})) \\ \text{subject to} \quad & \tilde{\mathbf{y}}(k_\mathrm{p}) = \mathscr{P}(\dot{\mathbf{x}}_\mathrm{c}(t_0),\mathbf{x}_\mathrm{c}(t_0),\mathbf{x}_\mathrm{d}(t_0),\mathbf{z}(t_0),\tilde{\mathbf{u}}(k_\mathrm{c})) \\ & \tilde{\mathbf{u}}_\mathrm{lower} \leq \tilde{\mathbf{u}}(k_\mathrm{c}) \leq \tilde{\mathbf{u}}_\mathrm{upper}, \end{aligned}$$

where $\mathbf{u}_\mathrm{lower}$ and $\mathbf{u}_\mathrm{upper}$ are given bounds on $\mathbf{u}$, and $\tilde{\mathbf{u}}_\mathrm{lower} = [\mathbf{u}_\mathrm{lower}^T,\ldots,\mathbf{u}_\mathrm{lower}^\mathrm{T}]^\mathrm{T}$ and $\tilde{\mathbf{u}}_\mathrm{upper} = [\mathbf{u}_\mathrm{upper}^T,\ldots,\mathbf{u}_\mathrm{upper}^\mathrm{T}]^\mathrm{T}$.

Note that instead of keeping the relation for the prediction model as an explicit equality relation, this relation can be eliminated by substituting it into the objective function, which has computational advantages. The resulting control problem reduces to minimization of the objective function subject to simple bound constraints,

$$\begin{aligned} \min_{\tilde{\mathbf{u}}(k_\mathrm{c})} \quad & J(\tilde{\mathbf{u}}(k_\mathrm{c}),\mathscr{P}(\dot{\mathbf{x}}_\mathrm{c}(t_0),\mathbf{x}_\mathrm{c}(t_0),\mathbf{x}_\mathrm{d}(t_0),\mathbf{z}(t_0),\tilde{\mathbf{u}}(k_\mathrm{c}))) \qquad (3) \\ \text{subject to} \quad & \tilde{\mathbf{u}}_\mathrm{lower} \leq \tilde{\mathbf{u}}(k_\mathrm{c}) \leq \tilde{\mathbf{u}}_\mathrm{upper}. \end{aligned}$$

[3]In the case that all voltage magnitude errors should be penalized the infinity norm can simply be replaced by the one or two norm.

Since the objective function of this problem includes the procedure to compute the prediction model and due to the definition of $\mathbf{y}_{\mathrm{err}}$, this is a nonlinear, non-smooth optimization problem subject to simple bound constraints. In addition it should be mentioned that evaluating the objective function is expensive, since it involves time-domain simulation of the prediction model. In the following two approaches are considered to solve problem (3). In particular the direct-search method pattern search is used as an appropriate solver, for its more effective way of dealing with the problem at hand when compared to solvers for nonlinear optimization that require gradient or Hessian information.

4. Multi-start pattern search for nonlinear optimization

In optimization problem (3), evaluating the objective function is expensive due to the evaluation of the prediction model. In practice, within the limited available computation time, a solution that is as good as possible has to be determined. Many nonlinear optimization methods rely on gradient and Hessian information (Nocedal and Wright, 1999; Bertsekas, 2003). However, the saturations and the use of the infinity norm in the objective function make that the objective function has many flat areas in which the gradient and Hessian are both equal to zero and thus not informative. Solvers that use this first-order or second-order information will therefore perform unnecessary numerical estimation, involving numerous objective function evaluations. In addition, due to the non-smoothness of the problem there are many local minima in which this type of solvers typically can get stuck quickly.

Instead of using gradient-based or Hessian-based solvers, so-called direct-search optimization methods can be used, which do not explicitly require gradient and Hessian information (Conn et al., 1997; Wright, 1996). The only property that these methods require is that the values of the objective function can be ranked (Lewis et al., 2000). Moreover the feature that direct-search methods are suitable for non-smooth problems (Conn et al., 1997), make that these methods are suitable for solving the control problem at hand. The direct-search method which is proposed to be used in particular is pattern search (Lewis et al., 2000), for its straightforward implementation and its ability to yield good solutions, even for objective functions with many local minima, in combination with a multi-start method, to improve the probability of obtaining a solution close to a globally optimal solution. Several theoretical issues of pattern search are discussed in (Torczon, 1997; Lewis and Torczon, 2000; Audet and Dennis Jr., 2007). Appendix Appendix A contains more details on the way in which pattern search works.

The combination of pattern search with multi-start for solving the control problem at control step k_{c} consists of solving the control problem from n_{init} different initial solutions. The proposed multi-start implementation involves starting from different initial solutions as long as computation time is available. The first initial solution is based on the shifted solution of control sample step $k_{\mathrm{c}}-1$, since the solution of control sample step $k_{\mathrm{c}}-1$ typically gives a good guess of the solution at control sample step k_{c}. The solution with the minimal objective function value after optimization with pattern search when the maximum optimization time has elapsed is used as the solution at control sample step k_{c}. Although multi-start methods generally increase the time

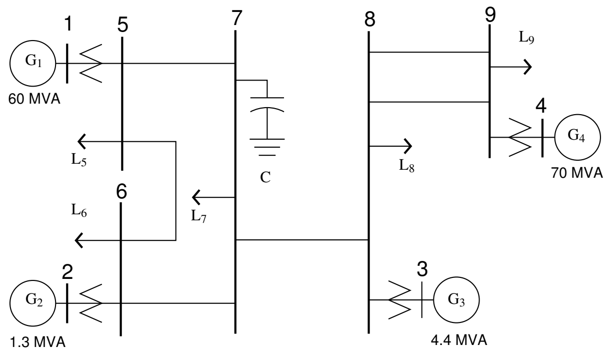

Figure 2: Network topology of the 9-bus dynamic network.

required to solve an optimization problem significantly, they can typically be executed in a highly parallel fashion.

5. Case study: A 9-bus power network

In this section case studies are performed on a voltage control problem in a 9-bus power network. The system and the control setup are described and uncontrolled behavior is illustrated. Then pattern search is compared with a gradient-based approach, and the controlled behavior is illustrated.

5.1. System

Figure 2 shows the topology of the system under study. This system is the 9-bus Anderson-Farmer network (Farmer and Anderson, 1996), as taken from the Dynamical Systems Benchmark Library[4], to which the reader is referred for an exhaustive description. The network consists of 4 generators (G_1–G_4), 5 loads (L_5–L_9), and a capacitor bank. Generators G_1 and G_4 are aggregate generators representing the combination of multiple smaller generators. These generators are modeled with second-order dynamics (Kundur, 1994). Generators G_2 and G_3 are smaller generators and modeled in more detail with sixth-order dynamic models (Kundur, 1994). Transmission lines are represented by the classical π model. The loads are modeled in the original system with static voltage-dependent relations. These static loads have been replaced with dynamic loads by modeling the loads in more detail with a second-order ZIP model (Hill, 1993). Hence, the power network contains both fast dynamics (due to the generators) and slow dynamics (due to the loads).

5.2. Control setup

Control of the power network is done through two-layered control. The lower layer consists of an automatic voltage regulator and turbine governor for each generator, a

[4]URL: `http://psdyn.ece.wisc.edu/IEEEbenchmarks/`

power system stabilizer for generators G_2 and G_3, load shedding controllers for each load, and a capacitor bank switcher. Summarizing, the higher-layer controller can provide the following set-points to the lower control layer:

- the voltage magnitude set-points for the automatic voltage regulators;
- the mechanical power set-point for the turbine governors;
- the reference frequency for the turbine governors and power system stabilizers;
- the amount of load to shed;
- the number of capacitor banks to connect to the grid.

Depending on the particular control problem a higher-layer controller will adjust the values of these controls. For the problem at hand the amount of load shed and the set-points of the automatic voltage regulators are taken as the available controls.

Since typically uncontrolled slow voltage collapses emerge over time spans of several tens of seconds up to several minutes (Van Cutsem and Vournas, 1998), a control sample time of 20 s is acceptable. It should be noted that the speed at which a voltage collapse unfolds depends on the magnitude of the fault occurring. So, depending on the range of faults that should be adequately dealt with, the control sample time will have to be decreased or may be increased.

5.3. Control problem

The control problem of the supervisory controller is formulated as specified in Section 3, using a prediction model that matches the dynamics of the power network and the lower control layer, and a prediction horizon of 40 s. The voltage magnitudes should stay between 0.9 and 1.1 per unit (p.u.).

Figure 3 shows a typical scenario considered where a fault consisting in a 600% impedance increase in the transformer between bus 1 and 5 occurs. Due to the changed transmission capacity of the network and the dynamics of the loads, the voltage magnitudes start to oscillate with a decreasing trend. If set-points to the decentralized controllers of the lower layer are not updated, the network collapses.

5.4. Pattern search versus gradient-based optimization

Pattern search, as part of Matlab's Direct Search and Genetic Algorithms toolbox and in a multi-start implementation, is compared with the derivative-based solver SNOPT, as implemented in Tomlab. SNOPT uses a sparse sequential quadratic programming method, using limited-memory quasi-Newton approximations to the Hessian of the Lagrangian. In principle it requires gradient information, but this information can be approximated numerically if it is not available.

To compare the performance of the solvers, 50 experiments are performed in which faults occur at varying locations (i.e., at the 4 transformers and all lines), with varying magnitudes (i.e., an impedance increase of 100% up to 800%), and at varying times (i.e., the fault time varies between second 20 and 28). The control problems of the first

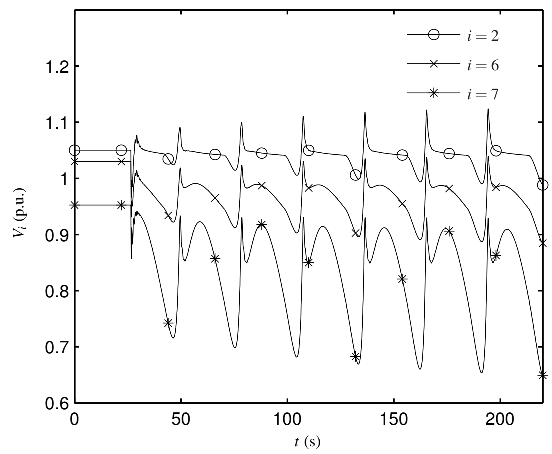

Figure 3: Voltage magnitude profiles V_i of 3 representative buses i, ($i = 2$, $i = 6$, and $i = 7$), for a typical scenario in which no higher control layer is present. After a fault consisting in a 600% impedance increase occurs in the transformer between buses 1 and 5 at $t_{\text{fault}} = 26.5$ s, the voltage magnitudes V_i start oscillating, ultimately resulting in a network collapse.

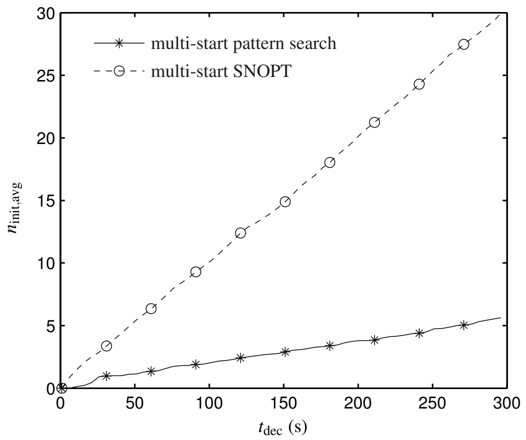

Figure 4: The variable $n_{\text{init,avg}}$ as a function of decision making time t_{dec} is defined as the average over all experiments of the total number of initial solutions considered so far, either for the pattern search approach or for the SNOPT approach.

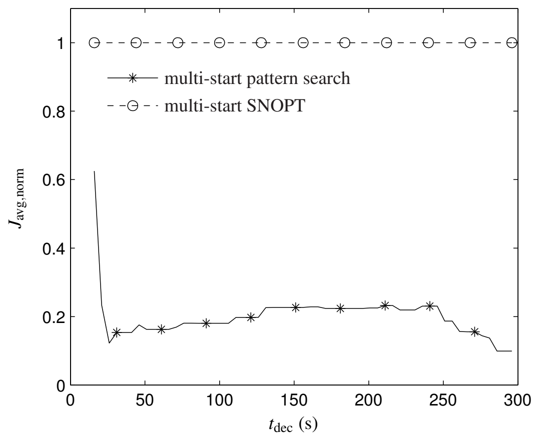

Figure 5: Average normalized performance $J_{\text{avg,norm}}$ of pattern search and SNOPT over all experiments as a function of the decision making time t_{dec}.

control sample step after a fault has been applied are solved by both pattern search and SNOPT, allowing a total decision making time of 300 s[5].

Figure 4 shows that SNOPT considers far more initial solutions within the allowed decision making time span. The time that SNOPT requires to obtain a locally optimal solution is much smaller than the time required by pattern search. This is because SNOPT uses much fewer prediction model evaluations per optimization, since it does not explore the search space as much as pattern search. Figure 5 shows, as decision time t_{dec} progresses, the so-called average normalized performance of multi-start pattern search and multi-start SNOPT. For a particular decision making time t_{dec}, the average normalized performance $J_{\text{avg,norm}}$ of pattern search is computed as follows: the best objective value of pattern search so far is divided by the best objective value of SNOPT so far, and for this quantity the average is taken over all experiments. For obtaining the average normalized performance of SNOPT, the average value of the best objective value of SNOPT so far is divided by itself, yielding values of 1 throughout the decision making step. The figure considers only points for which the fraction can be computed, i.e., both pattern search and SNOPT have finished at least one optimization problem. Pattern search on average, has a best objective value so far that is about a factor 5 smaller than the best objective value so far of SNOPT, and that, hence, pattern search yields significantly lower costs. The comparison shows that pattern search, although it does not require gradient or Hessian information and is straightforward to implement, generally provides solutions that outperform the solutions provided by SNOPT.

[5]This relatively long decision making time is taken to illustrate how the performance of both solvers varies over time. In practice, multiple processors can be employed to parallelize the multi-start approach and in this way obtain acceptable solutions in a more realistic time frame. In addition all code can be optimized for speed and implemented in object code (currently only the SNOPT code is in object code).

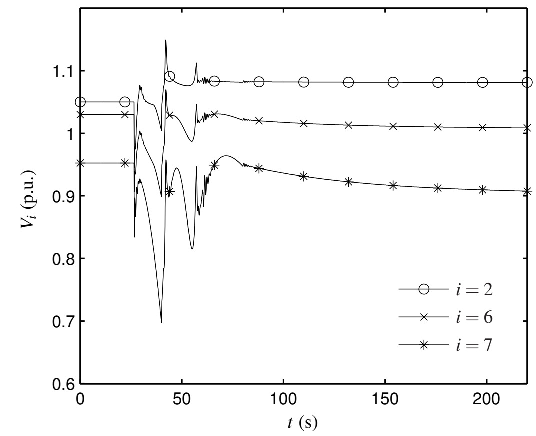

Figure 6: Voltage magnitude profiles for simulation including a higher-layer MPC controller.

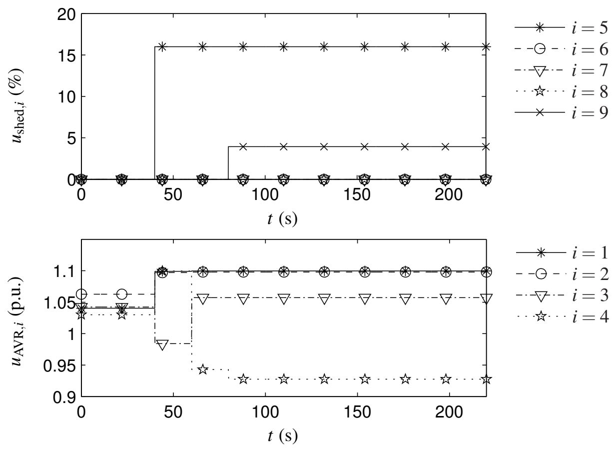

Figure 7: Evolution of the set-points provided by the supervisory controller to the automatic voltage regulators ($u_{\mathrm{AVR},i}$, in p.u.) and the amount of load to shed ($u_{\mathrm{shed},i}$, in % of the total load) for simulation including this controller.

5.5. *Controlled scenario*

To illustrate the performance of the proposed approach, consider again the fault consisting in a 600% impedance increase at $t_{\mathrm{fault}} = 26.5$ s in the transformer in the line from bus 1 to 5. The supervisory controller operates at $T_{\mathrm{c}} = 20$ s using the MPC strategy based on multi-start pattern search as discussed before. The supervisory controller uses a prediction of 40 s, and samples the voltage magnitudes from its prediction model every $T_{\mathrm{p}} = 0.5$ s.

Figures 6 and 7 show the resulting voltage profiles and set-points, respectively. After the fault has appeared, the supervisory controller is able to stabilize the voltage magnitudes between 0.9 and 1.1 p.u. and thus achieves its objectives.

6. Conclusions

A control strategy has been proposed for the controller in a higher control layer of a power network, that provides set-points to a lower layer with decentralized controllers, with the aim to prevent voltage collapses from occurring. The control strategy uses model predictive control (MPC) in which a prediction model describes both the dynamics of the system and the functioning of the lower control layer. The prediction model is formulated in an object-oriented modeling environment, allowing relatively easy construction of models of complex systems. Due to the nature of power networks, the prediction model involves differential, algebraic, and logic relations and is nonlinear, non-smooth, and computationally intensive to evaluate.

Pattern search has been proposed to solve the nonlinear MPC problem of the supervisory controller. Pattern search is a direct-optimization method that does not require gradients and/or Hessians, which are not available in the situation considered. Moreover, due to the discrete elements, such as saturations, the MPC problem is non-smooth, making gradient or Hessian-based approaches less suitable.

Simulation studies on a 9-bus dynamic power system have shown the potential of the proposed approach. It has been shown that the proposed controller is capable of preventing voltage collapses from occurring and that the pattern search method has superior performance when compared to a gradient-based method.

Future research will address the stability of the final closed-loop, employing a quasi-infinite horizon approach, in which a penalty term together with an additional terminal region are determined such that nominal stability of the closed-loop is guaranteed (Chen and Allgöwer, 1998). Future research will also consider using a reduced-order instead of a full-order prediction model, integrating the idea that a supervisory controller should use a less detailed representation of the system behavior to mainly steer the long-term behavior of the lower control layer.

Appendix A. Pattern search

The control problem (3) has an objective function that is computationally expensive to evaluate. Pattern search (Lewis et al., 2000) is proposed to solve the control problem. It is a so-called direct search method and does therefore not explicitly require gradient and Hessian information (Conn et al., 1997; Wright, 1996), and is suitable for solving non-smooth problems (Conn et al., 1997),

Pattern search works in an iterative way in which the solution $\mathbf{x}_s$ at iteration s is replaced by a new solution $\mathbf{x}^+$ only if $f(\mathbf{x}^+) < f(\mathbf{x}_s)$. The new solution $\mathbf{x}^+$ is selected from a finite set of candidate solutions $\mathscr{M}_s$ that is updated at each iteration. An iteration of pattern search is summarized as follows (Lewis et al., 2000):

1. A mesh $\mathscr{M}_s$ around current solution $\mathbf{x}_s$ is constructed, consisting of a discrete set of candidate solutions in $\mathbb{R}^n$ in which the algorithm searches for a new solution. The set of candidate solutions is defined using a set of directions, called the pattern, and a mesh size parameter. The pattern consists of a set of positive spanning

directions $\mathscr{D}$ of which non-negative linear combinations span the solution space. The mesh size parameter $\Delta_s \in \mathbb{R}^+$ determines the coarseness of the mesh. The mesh $\mathscr{M}_s$ then consists of the linear combinations of the current solution $\mathbf{x}_s$ with an individual direction $\mathbf{d}_i$ multiplied m times by the mesh size $\Delta_s \in \mathbb{R}^+$, for each $\mathbf{d}_i \in \mathscr{D}$ and for $m = \{1, 2, \ldots\}$.

2. The mesh $\mathscr{M}_s$ is explored in the following way:
   - In the so-called *search phase* any strategy (e.g., a random or heuristic search) can be used to find a solution $\mathbf{x}^+ \in \mathscr{M}_s$ for which $f(\mathbf{x}^+) < f(\mathbf{x}_s)$, as long as a finite number of points is considered. If a solution $\mathbf{x}^+$ is found, the search was successful and the algorithm continues with step 3.
   - If no solution $\mathbf{x}^+$ is found in the search phase, then the exploration of the mesh $\mathscr{M}_s$ continuous with the so-called *polling phase*. In the polling phase a new solution $\mathbf{x}^+$ for which $f(\mathbf{x}^+) < f(\mathbf{x}_s)$ is searched for in a subset of solutions in $\mathscr{M}_s$, consisting of those solutions that are in the direct neighborhood of the current solution $\mathbf{x}_s$. The direct neighborhood consists of these points of the mesh obtained using multiplication factor $m = 1$. If a solution $\mathbf{x}^+$ is found in this neighborhood then the polling phase was successful.
3. If either of the phases was successful, then $\mathbf{x}_{s+1} = \mathbf{x}^+$, the coarseness of the mesh is set to $\Delta_{s+1} = \varepsilon\Delta_s$, with expansion factor $\varepsilon > 1$, and the next iteration starts. If $\mathbf{x}^+$ was not found, then $\mathbf{x}_{s+1} = \mathbf{x}_s$, the coarseness of the mesh is set to $\Delta_{s+1} = \gamma\Delta_s$, with contraction factor $\gamma \in (0, 1)$, and the next iteration starts.

The iterations continue until a stopping condition is satisfied, e.g., the mesh size is less than a given tolerance, the total number of objective function evaluations reaches a given maximum, or the distance between the point found at one successful poll and the point at the next successful poll is less than a given tolerance.

Analyzing the convergence of pattern search is complicated, since no explicit representation of the directional derivative of the objective function is available. Analysis of pattern search uses a simple decrease criterion, requiring ordinal information. This criterion specifies that a new solution is accepted only if the value of that new solution is strictly less than the value of the previous solution. Pattern search employs at least $n_x + 1$ points (where n_x is the dimension of the solution vector $\mathbf{x}$). This ensures sufficient information about the entire n_x-dimensional domain in the neighborhood of the current iterate. Under very mild assumptions on the objective function f, these simple heuristics provide enough structure to guarantee global convergence to a stationary, usually local-minimum, solution. See (Torczon, 1997; Lewis and Torczon, 2000; Audet and Dennis Jr., 2007) for more details.

Acknowledgments

Research supported by the BSIK project "Next Generation Infrastructures (NGI)", the Delft Research Center Next Generation Infrastructures, the project "Multi-agent control of large-scale hybrid systems" (DWV.6188) of the Dutch Technology Foundation STW, the European 6th Framework Network of Excellence "HYCON", an NWO

Van Gogh grant (VGP79-99), and the European STREP project "Hierarchical and distributed model predictive control (HD-MPC)".

References

Al-Othman, A. K., El-Naggar, K. M., Jan. 2007. Application of pattern search method to power system security constrained economic dispatch. Transactions on Engineering, Computing and Technology 19 (1), 28–33.

Audet, C., Dennis Jr., J. E., 2007. Analysis of generalized pattern searches. SIAM Journal on Optimization 13 (3), 889–903.

Barton, P., Pantelides, C., 1994. Modeling of combined discrete/continuous processes. AIChE Journal 40 (6), 966–979.

Beccuti, A. G., Morari, M., Jun. 2006. A distributed solution approach to centralized emergency voltage control. In: Proceedings of the 2006 IEEE American Control Conference. Minneapolis, Minnesota, pp. 3445–3450.

Bertsekas, D. P., 2003. Nonlinear Programming. Athena Scientific, Beltmore, Massachusetts.

Bleris, L. G., Vouzis, P. D., Garcia, J. G., Arnold, M. G., Kothare, M. V., Oct. 2007. Pathways for optimization-based drug delivery. Control Engineering Practice 15 (10), 1280–1291.

Brenan, K. E., Campbell, S. L., Petzold, L. R., 1996. Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations. SIAM, Philadelphia, Pennsylvania.

Chen, H., Allgöwer, F., Oct. 1998. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. Automatica 34 (10), 1205–1217.

Conn, A. R., Scheinberg, K., Toint, P. L., 1997. Recent progress in unconstrained nonlinear optimization without derivatives. Mathematical Programming 79 (1–3), 397–414.

De Schutter, B., van den Boom, T., Hegyi, A., 2002. A model predictive control approach for recovery from delays in railway systems. Transportation Research Record (1793), 15–20.

Diehl, M., Bock, H. G., Schlöder, J. P., Findeisen, R., Nagy, Z., Allgöwer, F., Jun. 2002. Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. Journal of Process Control 12 (4), 577–585.

Dynasim, 2004. Dymola – User's Manual. Tech. rep., Dynasim AB, Lund, Sweden.

Fardanesh, B., Jul. 2002. Future trends in power system control. IEEE Computer Applications in Power 15 (3), 24–31.

Farmer, R. G., Anderson, P. M., 1996. Series Compensation of Power Systems. PBLSH, Encinitas, California.

Geyer, T., Larsson, M., Morari, M., Sep. 2003. Hybrid emergency voltage control in power systems. In: Proceedings of the European Control Conference 2003. Cambridge, UK, paper 322.

Hill, D. J., Feb. 1993. Nonlinear dynamic load models with recovery for voltage stability studies. IEEE Transactions on Power Systems 8 (1), 166–176.

Hill, D. J., Guo, Y., Larsson, M., Wang, Y., 2003. Global control of complex power systems. In: Chen, G., Hill, D. J., Yu, X. (Eds.), Bifurcation Control: Theory and Applications. Lecture Notes in Control and Information Sciences. Springer, Berlin, Germany, pp. 155–187.

Jenkins, N., Allan, R., Crossley, P., Kirschen, D., Strbac, G., 2000. Embedded Generation. TJ International, Padstow, UK.

Kundur, P., 1994. Power System Stability and Control. McGraw-Hill, New York, New York.

Larsson, M., Karlsson, D., Aug. 2003. Coordinated system protection scheme against voltage collapse using heuristic search and predictive control. IEEE Transactions on Power Systems 18 (3), 1001–1006.

Leirens, S., Buisson, J., Bastard, P., Coullon, J.-L., Aug. 2005. A hybrid approach for voltage stability of power systems. In: Proceedings of the 15th Power Systems Computation Conference. Liège, Belgium, paper 291.

Lewis, R. M., Torczon, V., 2000. Pattern search methods for linearly constrained minimization. SIAM Journal on Optimization 10 (3), 917–941.

Lewis, R. M., Torczon, V., Trosset, M. W., Dec. 2000. Direct search methods: then and now. Journal of Computational and Applied Mathematics 124 (1–2), 191–207.

Maciejowski, J. M., 2002. Predictive Control with Constraints. Prentice-Hall, Harlow, UK.

Majanne, Y., Dec. 2005. Model predictive pressure control of steam networks. Control Engineering Practice 13 (12), 1499–1505.

Manzoni, A., de Silva, A. S., Decker, I. C., 1999. Power systems, dynamics simulation using object-oriented programming. IEEE Transactions on Power Systems 14 (1), 249–255.

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

Mattsson, S. E., Elmqvist, H., Otter, M., Apr. 1998. Physical system modeling with Modelica. Control Engineering Practice 6 (4), 501–510.

Mesarovic, M. D., Macko, D., Takahara, Y., 1970. Theory of Hierarchical Multilevel Systems. Academic Press, New York, New York.

Navarro, I. R., Larsson, M., Olsson, G., Jan. 2000. Object-oriented modeling and simulation of power systems using Modelica. In: Proceedings of the IEEE Power Engineering Society Winter Meeting. Singapore, pp. 790–795.

Negenborn, R. R., Beccuti, A. G., Demiray, T., Leirens, S., Damm, G., De Schutter, B., Morari, M., Jul. 2007. Supervisory hybrid model predictive control for voltage stability of power networks. In: Proceedings of the American Control Conference 2007. New York, New York, pp. 5444–5449.

Nocedal, J., Wright, S., 1999. Numerical Optimization. Springer Series in Operations Research. Springer-Verlag, New York.

Oliveira, N. M. C., Biegler, L. T., Feb. 1995. An extension of Newton-type algorithms for nonlinear process control. Automatica 31 (2), 281–286.

Petzold, L. R., Aug. 1983. A description of DASSL - A differential/algebraic system solver. In: Proceedings of the 10th World Congress on System Simulation and Scientific Computation. Montreal, Canada, pp. 430–432.

Piela, P. C., Epperly, T. G., Westerberg, K. M., Westerberg, A. W., Jan. 1991. ASCEND: an object-oriented computer environment for modeling and analysis: The modeling language. Computers and Chemical Engineering 15 (1), 53–72.

Piñón, S., Camacho, E. F., Kuchen, B., Peña, M., Dec. 2005. Constrained predictive control of a greenhouse. Computers and Electronics in Agriculture 49 (3), 317–329.

The Modelica Association, 2005. Modelica - A Unified Object-Oriented Language for Physical Systems Modeling – Language specification. URL: `http://www.modelica.org/documents/ModelicaSpec22.pdf`.

Torczon, V., 1997. On the convergence of pattern search algorithms. SIAM Journal on Optimization 7 (1), 1–25.

Van Cutsem, T., Vournas, C., 1998. Voltage Stability of Electric Power Systems. Kluwer Academic Publishers, Dordrecht, The Netherlands.

Wright, M. H., 1996. Direct search methods: Once scorned, now respectable. In: Griffiths, D. F., Watson, G. A. (Eds.), Numerical Analysis 1995. Addison Wesley, Harlow, UK, pp. 191–208.