UNIVERSITY OF LIEGE

DEPARTMENT OF ELECTRICAL ENGINEERING
AND COMPUTER SCIENCE

DISTRIBUTED CONTROL OF ELECTROMECHANICAL OSCILLATIONS IN VERY LARGE SCALE ELECTRIC POWER SYSTEMS

BY

DA WANG

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF DOCTOR OF PHILOSOPHY IN ENGINEERING SCIENCE

DOCTORAL THESIS

2013-2014

Abstract

Electromechanical oscillations were observed in power systems as soon as synchronous generators were interconnected to provide more power capacity and supply reliability. These oscillations are manifested in the relative motions of generator mechanical axes accompanied by power and voltage oscillations. Some characteristics of modern large-scale electric power systems, such as long transmission distances over weak grids, highly variable generation patterns and heavy loading, tend to increase the probability of appearance of sustained wide-area electromechanical oscillations. The term "wide-area" is used here to emphasize the possible co-existence of local and inter-area oscillation modes of different frequencies that might appear simultaneously in different parts of large-scale systems. Such oscillations threaten the secure operation of power systems and if not controlled efficiently can lead to generator outages, line tripping and even large-scale blackouts.

In general, it is usually considered that electromechanical oscillations are caused by insufficient system damping. Due to the limited reinforcement available for current network structure, most of efforts for electromechanical oscillations damping focus on setting different controllers such as Power System Stabilizers (PSSs), Thyristor Controlled Series Compensators (TCSCs), and so on, in order to improve system damping. These damping controllers mostly use local measurements at their inputs, and their control rules and parameters are determined in off-line studies using time-domain simulations, Prony or eigenanalysis, and usually remain fixed in practice.

However, increasing uncertainties brought by renewable generation, and the growing complexity resulting from new power flow control devices, could make the damping effects of these designs become questionable in practice. Moreover, the controllers scattered into different areas and installed at different moments are to be further coordinated to obtain better global damping performances. So, it yields the need for more efficient, adaptive, and more

widely coordinated damping control schemes.

To this end, this thesis proposes a trajectory-based supplementary damping control. It treats damping control as a multi-step optimization control problem with discrete dynamics and costs, which calculates supplementary input signals for existing controllers using model-based methods or learning-based methods. At each control time, it collects current system states, solves the optimal control problem and superimposes the calculated supplementary inputs on the outputs of existing controllers. With the help of these supplementary inputs, it forces the controlled system response to evolve along the desired trajectory. These supplementary signals are continuously updated, which allows to adaptively adjust and coordinate a subset of the existing damping controllers, and eventually all of them.

Assuming that power system dynamics can be quite accurately modeled, and given the recent progresses in large-scale optimization, a natural idea is to apply Model Predictive Control (MPC) to embody the proposed control scheme. MPC is a proven technique with numerous real-life applications in different engineering fields and it may be designed in different ways: centralized, distributed and hierarchical.

In this thesis, firstly, a fully centralized MPC scheme is designed based on a linearized, discrete, complete state space model. Its performances are evaluated both in ideal conditions and considering realistic state estimation errors, and computation and communication delays. The effects of the number and type of available damping controllers are also studied in order to assess the versatility of this scheme. This scheme is further extended into a distributed scheme with the aim of making it more viable for very large-scale or multi-area systems. Different ways of decoupling and coordinating between subsystems are analyzed. Finally, a robust hierarchical multi-area MPC scheme is proposed, introducing a second layer of MPC based controllers at the level of individual power plants and transmission lines. The performances of these three MPC schemes are illustrated and compared on a 16 generators, 70 buses test system.

While MPC, being a closed-loop control scheme, has some intrinsic level of robustness to modeling errors, it nevertheless relies on the use of a correct dynamic model of the system. Within the context of power system oscillation damping, load-dynamics, and dynamics of renewable and dispersed generation may have a significant impact on the system behaviour; since the composition of the load and dispersed generation may change significantly from one period of time to another (e.g. intra-daily, and seasonal effects

driven by weather conditions) the system dynamics at a particular moment may not be well enough approximated by the model computed from the available data in TSO control centers to yield satisfactory performances of any one of the proposed MPC schemes.

Another supplementary damping control, considered in this thesis, is based on model-free learning methods. Specifically, a tree-based batch mode Reinforcement Learning (RL) algorithm is applied in place of MPC to calculate the supplementary signals for existing damping controllers. Using a set of dynamic and reward four-tuples, it constructs an approximation of the optimal Q-function over a given temporal horizon by iteratively extending its prediction horizon. In each iteration, the Q-function is approximated by an ensemble of extra-trees. Between two iterations, the Q-function is refreshed using the obtained instantaneous rewards at current step and the Q-function of the previous step. The actions greediest with respect to the Q-function are applied as supplementary signals to existing damping controllers. The scheme is firstly tested on a single generator, and then on multiple generators. Finally, the combined control effects of MPC and RL are also investigated.

Acknowledgments

First of all, I would like to express my sincere gratitude to my professor, Louis Wehenkel, for the valuable research opportunity and guidance given by him. I really learned a lot from his thorough insight into power systems stability and control, and his way of carrying out research.

I most deeply thank Mevludin Glavic for his very useful advices and constant help in my research.

I acknowledge the funding of University of Liège, which has supported my research and life during the past four years. This thesis presents research results of the Belgian Network DYSCO, funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State and of the PASCAL Network of Excellence funded by the European Commission. The scientific responsibility rests with the author.

I thank the members of the Jury for devoting themselves to read this manuscript and to all the related duties.

I am also indebted to Samuel Hiard for his help in my study and life, to Raphael Fonteneau for helping out with the fitted Q iteration code, to Diane Zander, and other colleagues and friends of Montefiore Institute.

List of acronyms and symbols

Acronyms

AC	Alternative Current	DAE	Differential Algebraic Equation
DC	Direct Current	DP	Dynamic Programming
EMS	Energy Management System	EW	East West oscillation mode
FACT	Flexible Alternative Current Transmission	HVDC	High Voltage Direct Current transmission
LC	Local Controller	MC	Monte Carlo
MPC	Model Predictive Control	PSS	Power System Stabilizer
PST	Power System Toolbox	RL	Reinforcement Learning
QP	Quadratic Programming	RAM	Random Access Memory
SE	State Estimation	SSSC	Static Synchronous Series Compensator
STATCOM	Static Synchronous Compensator	SVC	Static Var Compensator
TCSC	Thyristor Controlled Series Compensator	TD	Temporal Difference
TG	Turbine Governor	TSO	Transmission System Operator
UCTE	Union for the Coordination of the Transmission of Electricity	ULC	Unstable limit Cycle
UPFC	Unified Power Flow Controller	WAMS	Wide Area Measurement System

Symbols

A	State matrix	$\hat{A}$	Supervised learning algorithm
$\hat{a}$	Attribute	B	Input matrix
C	Output matrix	$\hat{c}$	Lower MPC correction
d	Variable number	$\hat{d}$	Output feedback
$\hat{E}$	Expectation	$\tilde{E}'_d$	d axis component of stator voltage
$\tilde{E}_f$	Field voltage	$\tilde{E}'_q$	q axis component of stator voltage
$\tilde{E}_r$	Electromotive force at the receiving end	$\tilde{E}_s$	Electromotive force at the sending end
f	Input-output function	$\hat{G}$	Transfer function
$\hat{H}$	Hypothesis space of input-output function	$\tilde{I}_s$	Current at the sending end
i	Discrete dynamic step	$\hat{i}$	State-action pair input
j	Iteration number	$\hat{j}$	$\hat{j}$-th test
$\hat{K}$	Attribute number	k	MPC control time step
$\hat{k}$	$\hat{k}$-th attribute	L	Expected loss over a learning sample set
l	l-th row in a matrix or element of a vector	$\hat{l}$	Loss function
M	Tree number	$\hat{M}$	Sample number
m	m-th subsystem	$\hat{m}$	$\hat{m}$-th sample
N	Local controller number	n	n-th local controller
$\hat{n}_{min}$	Leaf node size	$\hat{o}$	Output of state-action pair
P	Power	$\hat{P}$	Probability distribution
p	Differential operator	$\hat{p}$	$\hat{p}$-th evolution path
Q	State-action value function	R	Control return over a time horizon
$\hat{R}$	Line resistance	$\tilde{R}_f$	Exciter state
r	One-step instantaneous reward	S	RL state space
$\hat{S}$	Score for a splitting	$\tilde{S}_s$	Complex power at the sending end

$s[t]$	Set-point trajectory	s	RL state
U	Control action space	$u, \hat{u}$	Control action and estimated value
T	Matrix transposition	$\hat{T}$	Tree
T_h	Prediction horizon	T_c	Control horizon
t	Time	$\hat{t}$	Test threshold value
V	MPC cost function	$\tilde{V}_l$	Load bus voltage
V_r	Regulator voltage	V_{ref}	Voltage reference of exciter
V_t	Generator terminal voltage	V_{pss}	PSS output
W	Weight matrix	X	Power system state space
$\hat{X}$	Line reactance	$x, \hat{x}$	Power system state and estimated value
Y	Measured output space	$y, \hat{y}$	Measured output and estimated value
y_{ref}	Output reference	Z	Controlled output space
$\tilde{Z}_l$	Load impedance	$\tilde{Z}_r$	Line impedance at the receiving end
$\tilde{Z}_{sr}$	Mutual-impedance	$\tilde{Z}_{ss}$	Self-impedance
$\tilde{Z}_s$	Line impedance at the sending end	$z, \hat{z}$	Controlled output and estimated value
α	Searching step of Active Set methods	$\hat{\alpha}$	Time Difference learning variable
γ	Discount factor	Δ	Increment
δ	Rotor angle	$\hat{\delta}$	Time dynamic
ε	A small probability value	θ	Rotor angle difference
π	Control policy	τ	Delay
$\tilde{\psi}''_d$	d axis air gap flux linkage	$\tilde{\psi}''_q$	q axis air gap flux linkage
$\tilde{\psi}_{1d}$	Amortisseur circuit flux linkages of d axis	$\tilde{\psi}_{1q}$	Amortisseur circuit flux linkages of q axis
ω	Generator speed	ω_{ref}	Generator speed reference
$\mathbb{R}$	Real number space	$\mathcal{TS}$	Training set
$\mathcal{F}$	Four-tuple set		

Contents

Part I

Introduction

Chapter 1

Power system electromechanical oscillations

1.1 Oscillation modes

Electromechanical oscillations in power systems are usually characterized by low frequency and poor damping. The stability of these oscillations is of vital concern, and is a prerequisite for secure system operation.

In terms of oscillation ranges and frequencies, electromechanical oscillations can be divided into two categories [1, 3]:

- Local modes: oscillations associated with a single generator or a single power plant. These oscillations have the frequencies in the range 0.7 to 2Hz. In some cases, local oscillations can excite other oscillation modes, namely inter-area modes.

- Inter-area modes: the generators in one sub-area oscillate against the generators in other sub-areas. Inter-area oscillations have their frequencies in the range 0.1 to 0.8Hz. Once excited, inter-area oscillations may propagate over the whole system.

As far as this thesis is concerned, the focus is on inter-area electromechanical oscillations analysis and control. Let us take the European power system as an example to illustrate electromechanical oscillations [4]. The interconnected synchronous power system of continental Europe (former UCTE) extends from Greece and Iberic peninsula in the south, to Denmark and Poland

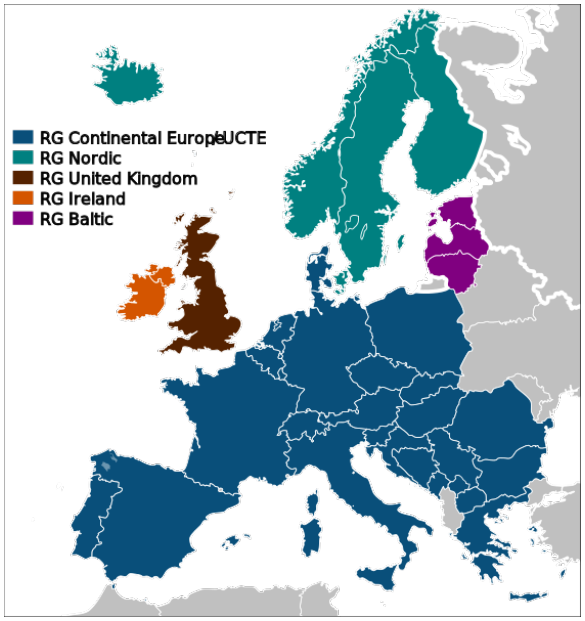

Figure 1.1: European interconnection synchronous areas. Taken from [4]

to the north and up to the border of the Black Sea to the east, as shown in Fig. 1.1. It is the largest interconnected system in the world, and in the year 2012 [5] it supplied power to about 500 million customers in 24 countries with a total annual energy consumption of approximate 3336TWh. In the same year this system had a peak load of about 508GW [5]. The system is operated in a decentralized way, by a group of about 40 TSOs, each one being responsible for a given control zone, assisted by two coordination centers.

In a classical scenario with a power flow of 350MW from Spain to France and of 1300 MW from CENTREL (Poland, Czech Republic, Slovakia and Hungary) to UCTE, four dominant inter-area modes (also called global modes) are identified in the low frequency range 0.2 to 0.5 Hz (adapted from [6]):

- Global mode 1: the generators in Spain and Portugal swing against the generators in the eastern part at a frequency of 0.2Hz.

- Global mode 2: this mode has a frequency of about 0.3 Hz and forms three coherent groups of generators. The generators in Spain and Portugal (group1) are almost in phase with the generators of CENTREL and eastern parts of Germany and Austria (group 2). The third group

(France, Italy, Switzerland and western parts of Germany and Austria) is in phase opposite to the other two groups.

- Global mode 3: it appears at a frequency of about 0.5 Hz and involves mainly the generators in Poland and Hungary. The parts of Austria, Slovakia and Italy are also involved while the rest UCTE system is almost not involved.

- Global mode 4: it is observed close to 0.5 Hz. This mode does not seem to be critical for the East-West transit (between UCTE and CENTREL) but has to be observed carefully in case of bulk power transfers from the North to the South.

Among these four oscillation modes, global mode 1 and 2 are the most interesting as they involve almost the whole interconnected system. Due to the foreseen interconnection of Turkey to the UCTE interconnected system, a new inter-area mode, close to 0.15 Hz, will probably appear [7].

Similar electromechanical oscillations are also detected in other interconnections around the world [8, 9].

Oscillations are acceptable as long as they decay rapidly enough. Sustained and increasing oscillations can cause generator outage, line tripping, network splitting and even blackout. Usually, the restriction or even the curtailment of power exchange among interconnected areas is the most effective operational measure avoiding such undamped oscillations. However, it is not the most effective approach from the economical point of view. More economical damping controls are discussed at the end of this chapter.

1.2 Mathematical formulations

Mathematical insights into electromechanical oscillations can be obtained with the help of a two-generator system shown in Fig. 1.2. The amplitudes of electromotive force $\tilde{E}_s$ and $\tilde{E}_r$ are assumed to be constant. $\tilde{E}_r$ is taken as the reference and the shunt capacitances of transmission line are ignored. $\tilde{Z}_s$ and $\tilde{Z}_r$ are the transmission line impedances at the sending end and at the receiving end. $\tilde{Z}_l$ represents a constant impedance load. Based on the circuit superposition principle, the current from the sending end $\tilde{I}_s$ is:

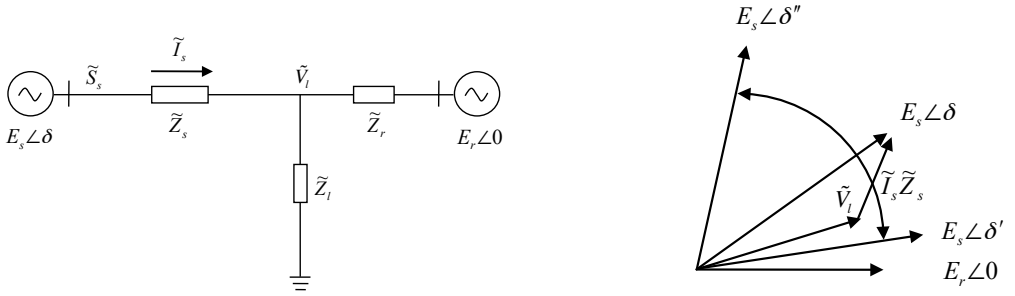

Figure 1.2: Two-generator system and corresponding phasor diagram

$$\tilde{I}_s = \frac{\tilde{E}_s}{\tilde{Z}_{ss}} - \frac{\tilde{E}_r}{\tilde{Z}_{sr}} = \frac{\tilde{Z}_l E_r}{\tilde{Z}_s\tilde{Z}_l + \tilde{Z}_s\tilde{Z}_r + \tilde{Z}_r\tilde{Z}_l}[\frac{E_s}{E_r}(\frac{\tilde{Z}_r}{\tilde{Z}_l} + 1)e^{j\delta} - 1] \tag{1.1}$$

where $\tilde{Z}_{ss}$ is the self-impedance and $\tilde{Z}_{sr}$ is the mutual-impedance.

$$\tilde{Z}_{ss} = \tilde{Z}_s + \frac{\tilde{Z}_r\tilde{Z}_l}{\tilde{Z}_l + \tilde{Z}_r}; \quad \tilde{Z}_{sr} = \tilde{Z}_s + \tilde{Z}_r + \frac{\tilde{Z}_s\tilde{Z}_r}{\tilde{Z}_l}; \tag{1.2}$$

The complex power at the sending end is:

$$\tilde{S}_s = \tilde{E}_s\tilde{I}_s^* = \frac{\tilde{Z}_l E_s E_r}{\tilde{Z}_s\tilde{Z}_l + \tilde{Z}_s\tilde{Z}_r + \tilde{Z}_r\tilde{Z}_l}[\frac{E_s}{E_r}(\frac{\tilde{Z}_r}{\tilde{Z}_l} + 1) - e^{j\delta}] \tag{1.3}$$

And the voltage at the node l is:

$$\tilde{V}_l = \tilde{E}_s - \tilde{I}_s\tilde{Z}_s = \frac{\tilde{Z}_s\tilde{Z}_r E_r}{\tilde{Z}_s\tilde{Z}_l + \tilde{Z}_s\tilde{Z}_r + \tilde{Z}_r\tilde{Z}_l}(\frac{\tilde{Z}_l E_s}{\tilde{Z}_s E_r}e^{j\delta} + 1) \tag{1.4}$$

The equations (1.1), (1.3) and (1.4) show when δ fluctuates in a certain range, the electrical variables like line current, transmission power and node voltage all will fluctuate correspondingly following δ. The right part of Fig.1.2 is an example of voltage fluctuation when δ varies from δ' to δ''.

1.3 Electromechanical oscillation mechanisms

1.3.1 Negative damping

It is usually considered that electromechanical oscillations are caused by insufficient damping or negative damping in power systems. The damping of power systems normally depends on system structures, controller parameters and operation conditions:

- System structures: electromechanical oscillations occur when existing generation/load areas are connected to other similar areas by relatively weak transmission lines. Weak connection means large inter-area line impedances. The frequencies and damping ratios[1] of inter-area oscillations drop as the inter-area line impedances increase. The inter-area DC links, parallel to AC tie-lines, can strengthen the inter-area connection and do not introduce new inter-area oscillation modes. But the DC links can affect the frequencies and damping ratios of existing inter-area modes [3]. In addition, when the high impedance lines are parallel with the low impedance lines, the loss of low impedance lines may cause instable inter-area oscillations [11].

- Controller parameters: an exciter is of major help in providing the synchronizing torque, but it also may destroy the natural damping of power systems [12]. The effect of a single fast exciter on damping depends on its location in the system, and the types of exciters on other generators [3]. Power System Stabilizer (PSS) is the most cost-effective method of enhancing the damping of electromechanical oscillations. But when a local power system is incorporated into a large power system, PSS parameters in this area must be verified carefully and adjusted in order to achieve a positive contribution for damping inter-area oscillations [6].

- Operating conditions: the damping of inter-area oscillations is also influenced by power flows. For example, the next foreseen expansion of European power system is the interconnection of Turkey power system to UCTE. The analysis of paper [7] shows that when Turkey exports

[1]For a pair of complex eigenvalues $\lambda = \sigma \pm j\omega$, the damping ratio ζ is $\zeta = -\frac{\sigma}{\sqrt{\sigma^2+\omega^2}}$. It determines the rate of decay of the amplitude of the oscillation [1, 10].

active power, the damping of inter-area mode EW-1 is strongly decreased.

- Load characteristics: electromechanical oscillations are accompanied by voltage oscillations at load nodes, and hence cause load variations. Conversely, load variations will influence system voltage profiles and influence electromechanical oscillations. So, it forms a feedback loop between loads and electromechanical oscillations. Load characteristics, including constant impedance, constant current, constant power characteristics and the rate of load recovery, all affect load response to voltage variations. It may alleviate or aggravate power system electromechanical oscillations [13, 14].

1.3.2 Other causes

In addition to the above described negative damping mechanisms, there are other causes of electromechanical oscillations:

- Forced oscillations: for power systems containing the sources of forced oscillations, such as various types of cyclic loads, diesel engine driven generators and wind turbine driven generators, if the frequency of any of electromechanical oscillation modes coincides with that of forced oscillations, a resonance phenomenon will occur. When the inherent damping of electromechanical oscillations is lower, the resonance of forced oscillations with an electromechanical oscillation mode could lead to a complete system breakdown [15].

- Modal (strong or weak) resonance: it appears when eigenvalue pairs of two stable oscillatory modes coincide approximately, due to the changes in power system operating parameters: the two pairs of eigenvalues first approach one another, interact, and then one of the eigenvalue pairs may cross the imaginary axis and become unstable [16, 17].

- Bifurcations: when the state matrix A has a pair of purely imaginary eigenvalues (a mode with zero damping), a Hopf bifurcation is said to occur. By carrying out certain Taylor series nonlinear computations, the mode can be classified to be supercritical (nonlinear stable), or subcritical (nonlinear unstable). For the subcritical Hopf mode, there exist unstable limit cycles (ULC) which bound the region of attraction

around small signal stable equilibria. The ULC defines the boundary separating oscillations of decreasing amplitude (positive damping response) and oscillations of growing amplitude (negative damping response). So, the oscillation stability can be judged by checking whether the post-fault initial condition is inside or outside the transient stability boundary anchored by a ULC [18, 19].

1.4 Analysis methods

1.4.1 Linear methods (adapted from [1])

The distinguishing feature of linear methods is to linearize a nonlinear power system at an equilibrium to obtain a linearized state space model. Based on a linearized model, modal analysis is used to calculate the eigenvalues of state matrix A. The eigenvalues may be real or complex. Since the state matrix A is real, the complex eigenvalues occur in complex conjugate pairs. Each complex conjugate pair of eigenvalues corresponds to one oscillation mode. By modal analysis, the dynamic behaviors of power systems under all different oscillation modes can be investigated and all unstable or weakly damped oscillation modes can be detected. The participation factor indicates the influence of one state to a mode, which gives very useful information on the most efficient location to undertake any possible control measure.

Modal analysis describes the small signal behaviors around a specific operating point. The nonlinear behaviors during large perturbations are not taken into account. So, the damping controls designed using modal analysis must be tested by nonlinear simulations under a wide range of operating conditions and faults, in order to assure their control effects in practice [1].

1.4.2 Nonlinear methods

All nonlinear methods involve Differential Algebraic Equations (DAE) that depict system dynamics. Time domain simulation is the most prevalent nonlinear method and it is widely used by power system operators and planners to study power system dynamics. It firstly builds a system-wide model by combining all devices' DAE according to power system topology. Then, taking a certain operating point as the initial value, it calculates state variables and algebraic variables step by step with the help of different mathematical

techniques. If the simulation is continued for a sufficiently long time, it is possible to determine whether electromechanical oscillations decay with time (they are stable), or continue at a constant amplitude or increase in amplitude (they are unstable) by observing rotor angle trajectories, tie-line power trajectories or other state trajectories. The advantage of time domain simulation is to all the modeling of all nonlinear and discrete dynamics. But in practice, the time required for time domain simulation is long and the computation is large. Moreover, it only can tell whether the post-disturbance system is stable or not for a given set of scenarios, but it can not provide a deeper insight about oscillation modes [1].

Except for solving DAE step by step, we also can perform a two-order Taylor series expansion of DAE in the neighborhood of an operating point, and further transfer Taylor series into normal form or modal series to calculate system response including the nonlinearity. Contribution factor, nonlinear participation factor and other nonlinear indices derived from the higher-order items can provide more refined information about nonlinear interaction between the fundamental oscillation modes, which is unavailable in linear modal analysis [20, 21].

1.4.3 Signal analysis methods

Signal analysis methods, like Wavelet transform, Prony analysis and Hillbert-Huang transform [22–24], can directly extract electromechanical oscillation mode information from response curves obtained from measurements gotten from practical operation or from time-domain simulations.

Among these methods, the one most used is Prony analysis.

Defining the output as a linear combination of exponential functions, the basic Prony analysis estimates the damping, phases and magnitudes of electromechanical oscillation modes by solving a least-squares problem using evenly sampled data [25]. But, the estimation precision of Prony analysis may deteriorate due to the existence of measurement or simulation inaccuracies (e.g. noise, or inappropriate sampling frequencies). Multi-signal Prony analysis, input signal pre-filtering and iterative Prony method all can improve the precision of Prony analysis [26–28].

In addition, in the condition assuming the input is a finite summation of delayed signals with the same characteristic eigenvalue, Prony analysis can identify a reduced-order transfer function incorporating local and inter-area electromechanical oscillatory modes [23].

1.5 Controllers and actuators

Strengthening grid structure can improve power system damping and electromechanical oscillation stability. However, it is uneconomic and difficult to be realized due to economic and environmental constraints. On the other hand, diverse controllers and actuators are placed in power systems initially for certain reasons other than to damp oscillations. Once installed, these devices also can be used to increase the damping of certain electromechanical oscillation modes [1, 10].

1.5.1 PSS

PSS is the most cost-effective electromechanical oscillation control. It uses auxiliary stabilizing signals, like generator shaft speed and accelerating power to produce an additional component of electrical torque proportional to speed change for its excitation to increase electromechanical oscillation damping. During the past three decades, extensive researches were paid to PSS placement and parameter setting, which made PSS become a technologically mature and widely used damping control.

1.5.2 FACTs devices

FACTs do not indicate a particular controller but a host of controllers like Static Var Compensator (SVC), Static Synchronous Compensator (STATCOM), Thyristor Controlled Series Compensator (TCSC), Static Synchronous Series Compensator (SSSC), and Unified Power Flow Controller (UPFC). FACTs devices can flexibly and rapidly control node voltages, change line impedances and adjust power flow to enhance the damping of electromechanical oscillations.

1.5.3 HVDC modulation

Using HVDC modulation to damp AC system oscillations is another interesting technique. The damping of electromechanical oscillations in an AC system can indeed be increased by modulating the current at the rectifier or the current and voltage at the inverter. The design of the modulation scheme must however be tailored to a specific system [29].

While all the previous methods provide a set of potentially effective tools to help damping electromechanical oscillations, their controller tuning needed to reach an optimal level of damping may be sensitive to prevailing system operation conditions (such as topology, generation pattern, and loading level).

1.6 Conclusions

In this chapter, electromechanical oscillations have been shortly described, and some examples of oscillations in the UCTE power system were given. Furthermore, the evolution of electrical states in oscillations was illustrated mathematically using a two-generator equivalent system. Several factors, which influence power system damping, have been discussed. Electromechanical oscillations could be studied using linear methods, nonlinear methods and signal analysis methods.

Three types of frequently used oscillation damping controls have been briefly discussed. In practice, diverse damping controllers nonlinearly interact and jointly decide control effects. Introduction of new damping controllers could jeopardize damping effects of existing controllers [6]. Consequently, it is very important to coordinate diverse damping controllers and adjust their parameters following the evolution of operation conditions, in order to obtain satisfactory and robust control effects. It is the objective that this thesis attempts to achieve.

Chapter 2

Research motivations, methods and contributions

2.1 Research motivations

Electromechanical oscillations, especially inter-area electromechanical oscillations, become more common in large-scale interconnected power systems. Fig. 2.1 gives an example of inter-area electromechanical oscillations in the European power system [30]. These oscillations restrict inter-area power exchange, propagate disturbances over the whole system and even lead to blackouts, if cascading outages are caused [31–33]. Different controllers, like PSSs, TCSCs and SVCs, are already designed for damping electromechanical oscillations.

The typical model-based design of damping controllers of electromechanical oscillations normally begins with the recognition of oscillation modes, then proceeds to determine controller parameters producing better damping performances and robustness, and ends with the verification via time-domain simulations [1, 34]. The control rules and parameters are usually calculated based on local information and objectives, and remain "frozen" in practical application. In recent researches on electromechanical oscillations, some remote information reflecting global dynamics is introduced as additional inputs to local damping controllers so as to enhance their performances of damping inter-area oscillations [35–37].

However, increasing uncertainties brought by the renewable generation, and the growing complexity resulting from new power flow control devices,

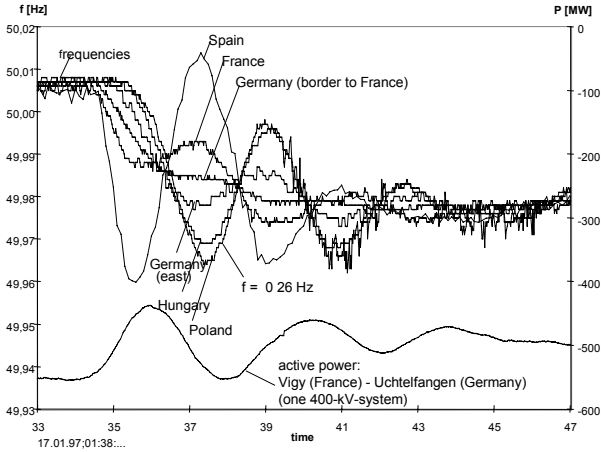

Figure 2.1: Inter-area oscillations after power plant outage in Spain. Taken from [30]

make the robustness of this typical design approach become questionable. Moreover, the controllers scattered into different areas and installed at different moments need to be further coordinated so as to obtain satisfactory global performances. Improper combination of diverse controllers could indeed potentially deteriorate the damping level of electromechanical oscillations [6].

This thesis investigates methods to adjust and coordinate existing damping controllers to improve further their global damping effects. This work is done from the following perspectives:

- Global information: due to the continuous extension of interconnected power systems, damping controllers designed based only on local information cannot always efficiently damp inter-area electromechanical oscillations. In order to obtain better global damping effects, a controller can benefit from a wider view of the system. That is to say, it should know more about the interaction between its dynamics and controls with external dynamics and controls when it makes a control decision.

  Emergence of new technological solutions such as synchronized phasor measurements and improved communication infrastructure enable the development of Wide Area Measurement Systems (WAMS) [38] and

the design of new types of controllers [39]. These controllers may be designed from the perspective of the whole system, focus on a wider spectrum of oscillation modes and offer improvements with respect to current local control strategies.

- Optimization: most damping control schemes only target to meet the minimum stability criterion, for example defined by a damping ratio larger than 0.05. Little work is dedicated to optimize damping control effects [40]. A scheme considering damping control optimization might potentially bring better control effects.

- Adaptivity: modern power systems abound with uncertainties on generation schedules, load patterns, and network topologies [41]. The real damping characteristics of power systems are complicated and volatile. A good damping scheme should be able to automatically adjust its control policies following such changes in the power system [42].

- Coordination: global damping effects are decided by all controllers existing in the system, which makes it important to coordinate their control efforts [43, 44]. A good damping control scheme should be able to make full use of available controllers and avoid the counteraction of damping effects.

The ultimate goal of all efforts to design, coordinate, and adapt damping controllers is to make the controlled system dynamics better meet the requirement of damping electromechanical oscillations.

In this thesis, a trajectory-based supplementary damping control is designed, which is superimposed on existing damping controllers (PSSs, TCSCs, and so on) to achieve the above improvements. It treats damping control as a discrete-time, multi-step optimal control problem. At a sequence of measurement times, it collects current system measurements, and based on these latter, it determines supplementary inputs to be applied at the next control time to existing damping controllers in order to obtain the maximum control return over a given temporal horizon. The objective of damping electromechanical oscillations can be achieved by defining a particular control return. For example, one can define the control return of a sequence of supplementary damping inputs as the negative distance between angular speeds and the rated angular speed over a future temporal horizon. Maximizing the return will force angular speeds to return and remain near the rated speed, and

when all generators run at this speed, oscillations are damped. Thus, optimization of these supplementary inputs brings adaption and/or coordination to the existing damping controllers without the need for changing their own structures and parameters. The above optimization is carried on repeatedly and supplementary inputs are updated adaptively at a series of subsequent times by taking into account changes in the power flows and topology of the power system.

2.2 Trajectory-based supplementary control

In this chapter, the proposed trajectory-based supplementary damping control is introduced in terms of feasibility, overall principle, mathematical formulation of the control objective, solution approaches and implementation strategy.

2.2.1 Feasibility

The feasibility of the proposed method derives from the following advances in power systems' technology, machine learning, and large-scale optimization.

- Dynamic and real-time measurements: WAMS can provide real-time and synchronized information about system dynamics, especially the information closely related to the recognition of oscillation modes and the improvement of global damping performances [45–48];

- Future response prediction: if a system model is available, future response of power systems can be approximately modeled; if not, the return over an appropriate temporal horizon of one damping control can be learned from the observation of power system trajectories in similar conditions, and then used to approximately predict the effect of controls on the future system response;

- Efficient algorithms: model-based methods and learning-based methods [49–51] have evolved a lot in the last twenty years by exploiting ongoing progress in terms of optimization and machine learning algorithms.

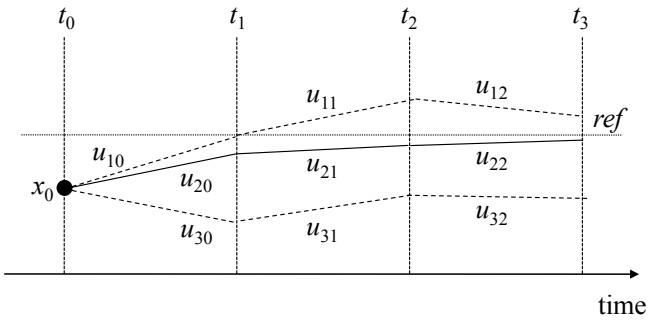

Figure 2.2: Trajectory-based supplementary damping control

2.2.2 Overall principle of the proposed method

The power system is considered in this work as a discrete-time system (power system dynamics are continuous in essence, but in our framework we consider discrete-time dynamics). Its trajectories are considered as time evolution of state variables of the controlled system. If the system model is available in the form:

$$x_{t+1} = f(x_t, u_t), \tag{2.1}$$

then it is possible to compute all future system dynamics by iterating Eqn. (2.1), and based on them optimize control policies. In Eqn. (2.1), x_t is a state vector consisting of elements of the state space X, and u_t is an input vector whose items are elements of action space U (random disturbances can be considered as a subset of these inputs).

If the system model is not available, then its trajectories can still be recorded by using a real-time measurement system.

In both cases constraints on states and actions can in principle also be incorporated by restricting the set of possible states and by restricting the actions that are possible for individual states.

Using certain supplementary inputs, we can force system dynamics to evolve approximately along the desired trajectories in which oscillations are damped, while taking into account random disturbances, prediction inaccuracy, measurement errors, and so on. This is the underlying concept of trajectory-based damping control.

This idea is illustrated in Fig. 2.2, which describes three possible angular speed trajectories of one generator. From an initial state x_0, the angular speed will evolve along a path $\hat{p}$ under a control sequence $(u_{\hat{p}0}, u_{\hat{p}1}, u_{\hat{p}2}, ..., \hat{p} =$

1, 2, 3. The target is to search a particular sequence of discrete supplementary inputs to the damping controller on this generator (exciter or PSS), in order to drive its angular speed to return to the reference, and remain close to the reference as much as possible, like path 2 in Fig. 2.2 (plain line).

So, the key of trajectory-based supplementary damping control is to search a correct and exact sequence of discrete supplementary inputs ($u_{\hat{p}0}, u_{\hat{p}1}$, $u_{\hat{p}2}, ...$). Here, the meaning of "correct" is that calculated supplementary inputs can make system dynamics evolve along the desired trajectory determined by a particular control objective, like the path 2 in Fig. 2.2. The meaning of "exact" is that there are not too large errors between decision-making scenarios and real system dynamics, like large measurement errors and model errors. If there are too large scenario errors, it may not be possible to find a control sequence yielding good damping effects in practice.

2.2.3 Control problem formulation

When system states move from x_t to x_{t+1} after applying an action u_t, a bounded reward of one step $r_t \in \mathbb{R}$ is obtained. The definition of r_t is closely related to the control objective. As far as damping electromechanical oscillations is concerned, the negative distance between the angular speed vector w and the rated angular speed vector w_{ref} is defined as r_t.

$$r_t = -\int_t^{t+1} |w - w_{ref}| \mathrm{d}t. \tag{2.2}$$

Starting from an initial state x_t and applying a sequence of supplementary inputs $(u_{t+0}, u_{t+1}, ..., u_{t+T_h-1})$, the discounted return $R_t^{T_h}$ over a temporal horizon of T_h is defined as:

$$R_t^{T_h} = \sum_{i=0}^{T_h-1} \gamma^i r_{t+i}, \tag{2.3}$$

where $\gamma \in [0, 1]$ is the discount factor, and $i = 0, 1, 2, ..., T_h - 1$. The sequence of actions u_{t+i} is computed by a control policy π mapping states to actions. A T_h-step optimal policy π^* is the one that maximizes $R_t^{T_h}$.

2.2.4 Model-based vs model-free solution methods

Depending if dynamic models are available in analytical form, or if control returns from past trajectories are available in numerical form, the solution to the T_h-step optimization control problem of Eqn. (2.3) is divided into two categories: model-based methods and model-free learning-based methods. Two approaches naturally fit this type of problems: MPC as model-based and RL as model-free.

1) Model Predictive Control (MPC): MPC is based on a linearized, discrete-time state space model given by:

$$\begin{aligned} \hat{x}[k+1|k] &= A\hat{x}[k|k] + B\hat{u}[k|k]; \\ \hat{y}[k|k] &= C\hat{x}[k|k]. \end{aligned} \tag{2.4}$$

The future dynamics over a temporal horizon of T_h is obtained by iterating (2.4):

$$\begin{bmatrix} \hat{y}[k+1|k] \\ \hat{y}[k+2|k] \\ \vdots \\ \hat{y}[k+T_h|k] \end{bmatrix} = P_x\hat{x}[k|k] + P_u \begin{bmatrix} \hat{u}[k|k] \\ \hat{u}[k+1|k] \\ \vdots \\ \hat{u}[k+T_h-1|k] \end{bmatrix}, \tag{2.5}$$

where P_x and P_u are given by,

$$P_x = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{T_h} \end{bmatrix}, \; P_u = \begin{bmatrix} CB & 0 & \dots & 0 \\ CAB & CB & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{T_h-1}B & CA^{T_h-2}B & \dots & CB \end{bmatrix}.$$

The return function of (2.3) can be rewritten as:

$$\begin{aligned} R_t^{T_h} = \sum_{i=0}^{T_h-1} & \left(\hat{y}[k+i+1|k] - y_{ref}[k+i+1|k]\right)^T W_{y_i} \\ & \left(\hat{y}[k+i+1|k] - y_{ref}[k+i+1|k]\right) \end{aligned} \tag{2.6}$$

which is minimized subject to linear inequality constraints:

$$\begin{aligned} u_{min} &\leq \hat{u}[k+i|k] \leq u_{max} \\ z_{min} &\leq \hat{z}[k+i+1|k] \leq z_{max} \end{aligned} \tag{2.7}$$

where y_{ref} is a vector of target values, $\hat{z}$ is a vector of constrained operation variables like currents or voltages, and the W_{y_i} are weighting matrices. $\hat{u}[k+i|k]$ varies over the first T_c steps, and T_c is called the control horizon that usually set equal to or less than prediction horizon T_h. This yields a typical quadratic programming problem, which can be solved by Active Set or Interior Point methods [49].

MPC works as follows: at a control time, based on current measurements, it calculates a sequence of optimal supplementary inputs minimizing the objective function (2.6) over a given temporal horizon. Only the first-stage control of the sequence is applied. The above steps are repeated at subsequent control times and continuously update these supplementary inputs.

In Part II of this work, we will investigate several MPC schemes, in particular a fully centralized one using a complete system model and system wide measurements, as well as area-wise distributed and hierarchical schemes.

2) Reinforcement Learning (RL): If the analytical system dynamics and return functions are unknown, one can still solve the problem by learning the map of system states to control actions using observations collected through a sample of real-time measurements. This problem is naturally set as a Markov Decision Problem (MDP) with the use of RL to learn the control policy. The use of system trajectories as time evolution of all system state variables is problematic, in this context, because of the so-called "curse of dimensionality" problem [50] and/or limitations in the measurement system and/or communications.

The approach adopted in Part III of this work is therefore to design a set of RL controllers (agents) acting on some system elements (e.g. generators) through learned mapping of its states (in the form of a single system state variable or a combination of several system state variables) towards local control actions along the system trajectories. Consequently, an RL agent considers trajectories of its state and overall system behaviour results from collective actions of individual RL agents. The states of these RL agents is to be clearly differentiated from the system state and we therefore denote the states considered by a given RL-based controller by s.

Given a set of trajectories represented in the form of samples of four-tuples (s_t, u_t, r_t, s_{t+1}), a near-optimal control policy is a sequence of control actions

minimizing the discounted return (2.3). This policy can be determined [52,53] by computing the so-called action-value function (also called Q-function) defined by:

$$Q(s_t, u_t) = E\{r_t(s_t, u_t) + \gamma \max_{u_{t+1}} Q(s_{t+1}, u_{t+1})\}, \tag{2.8}$$

and by then defining the optimal control policy as:

$$u_t^*(s_t) = arg \max_{u_t} Q(s_t, u_t). \tag{2.9}$$

In Part III of this thesis, the RL-based supplementary damping control scheme is first implemented on a single generator and then several possibilities are investigated for extending it to multiple generators. Finally, the possible benefits of combining RL-based control with Model Predictive Control (MPC) are assessed.

3) Discussion about solutions: MPC is a proven control technique with numerous real-life applications in different engineering fields, in particular process control [49]. The efforts applying MPC to damp electromechanical oscillations have already been reported in [37]. In this thesis, we study and compare three different schemes using MPC principles.

RL-based control of TCSC for oscillations damping has been proposed in [52]. In [54], RL is applied to adaptively tune the gains of the conventional PSSs. The use of RL to adjust the gains of adaptive decentralized backstepping controllers has been demonstrated in [55]. Wide-area stabilizing control, exploiting real-time measurements provided by WAMS, using RL has been introduced in [56].

Notice that both model-based or learning-based approaches usually find only suboptimal solutions due to the non-convexity of practical problems, modeling errors, randomness and limited quality of measurements.

While Q-learning based approaches have been proposed in previous works about oscillations damping [52, 54–56], in this thesis we propose to use a model-free tree-based batch mode RL algorithm to optimize supplementary inputs to existing damping controllers [57]. This choice is motivated by the following reasons [57–59]:

- This algorithm outperforms other popular RL algorithms on several nontrivial problems [59];

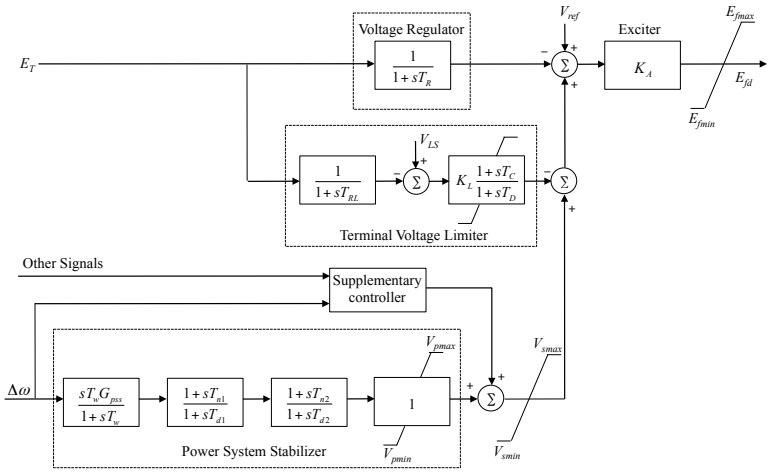

Figure 2.3: Supplementary damping control

- It can infer good policies from relatively small samples of trajectories, even for some high-dimensional problems;
- Using a tree-based batch mode supervised learning technique [58] it solves the generalization problem associated with RL techniques in a generic way [57, 59].

2.2.5 Implementation strategy

The proposed trajectory-based control scheme is not intended to replace existing damping controllers, but rather to optimize some supplementary control signals and superimpose them on the outputs of existing damping controllers so as to improve damping effects. In this way, the adaptation and/or the coordination of these existing controllers is implicitly achieved. The implementation considered in our work is illustrated in Fig. 2.3. For a given generator, the supplementary controller adds its control signal at the output of the PSS of the generator; it uses as inputs the angular speed of the generator and possibly other remote signals such as active power flows over tie-lines, so as to help damping of oscillation modes other than local ones. At a control time, such a controller collects inputs and then it calculates the supplementary control signals so as to maximize the control return over a future temporal horizon.

2.3 Thesis contributions and structure

2.3.1 Thesis contributions

Using the above proposed supplementary damping control approaches, the thesis attempts to design adaptive and coordinated damping control schemes based on the global information provided by WAMS and by leveraging recent progress in optimization and machine learning.

As the first step, a fully centralized MPC scheme is investigated (this work is reported in Chapter 4 and in our publication [60]). In this scheme, the MPC approach is implemented in a hypothetical system-wide control center and is responsible to compute supplementary inputs for all damping controllers so as to optimize their global control effects. It uses a linearized discrete-time state space model, calculates an optimal input sequence over a chosen time horizon by solving a quadratic programming problem. The first-stage control decision is applied to all controllers and the calculation is repeated at the subsequent measurement and control times. This scheme is firstly evaluated in ideal conditions: complete state observability and controllability, neglecting measurement errors and communication and computing delays. Next, the technical feasibility of this approach is assessed by studying also the effects of state-estimation errors, communication and computation delays and non-controllability of some local controllers. Finally, the scheme is tested to check if it can incorporate diverse damping controllers like PSS, TCSC and SVC, and efficiently coordinate them.

To design a damping control scheme more realistic in the context of large-scale interconnected power systems, the centralized MPC scheme is then replaced by a distributed MPC scheme (Chapter 5, and our publication [61]). Considering information-exchange restrictions and organizational barriers in certain power grids, a large-scale power system is decomposed into some smaller control areas (subsystems) according to practical constraints. And then a MPC controller is set in each subsystem. Similar to the centralized MPC, a subsystem-wide MPC controller formulates its quadratic programming problem using a subsystem-wide model and control objective. It solves this problem to provide supplementary inputs for damping controllers under its authority. Each MPC controller tries to solve its own oscillation problem and the overall system stability is hoped to result from these area-wise controls.

Next, a distributed and hierarchical MPC scheme is proposed to further

improve the robustness and reliability of damping control (Chapter 6, and our publication [62]). In addition to the MPC controllers working at the level of subsystems (the upper level), we set the new MPC controllers in the lower level consisting of local damping controllers. An upper MPC controller calculates supplementary inputs for all damping controllers under its authority, based on a subsystem model and subsystem-wide measurements. And it sends these inputs to lower MPC controllers as their control bases. The lower MPC controllers calculate, at a faster rate, corrections to the bases given by their corresponding upper MPC controllers, according to their own models, control objectives and measurements. Because the MPC controllers in the lower level only consider dynamics of one device like a generator or a transmission line, they need less time to measure, compute and apply their decisions. So, they can update more rapidly their control decisions following changes of local states, so as to approach better their control targets. The robustness of hierarchical MPC is tested using a larger refreshing interval at the upper level and incomplete measurements, considering controller and communication failures, and in different operation conditions.

Considering the possibility that system dynamics cannot be well enough approximated by a state space model at particular moments in practice, a model-free RL-based supplementary damping control is proposed in the second part of our research (Chapter 8, and our paper [63] under revision at the time of writing this manuscript). With the help of a tree-based batch mode RL, the control return of a candidate action at current state is approximately calculated over a given temporal horizon, based only on dynamic and reward information collected from observed trajectories of power systems. The action with the largest return is superimposed on existing controller's own output to improve damping effects. The scheme is first implemented on a single generator, and then several possibilities are investigated for extending it to multiple generators. Finally the possible benefits of combining this RL-based control with MPC are assessed.

2.3.2 Thesis structure

The thesis is organized as follows:

- Chapter 1 introduces power system electromechanical oscillations in terms of the definition, classification and consequences, and gives some oscillation examples in the European power system. Next, mathemat-

ical insights into electromechanical oscillations are provided on a two-generator equivalent system. Electromechanical oscillation mechanisms and analysis methods are also discussed in this chapter. Finally, common damping controls are shortly discussed.

- Chapter 2 presents research motivations, methods and contributions. The research objective is to design the optimized, adaptive and coordinated damping control scheme based on global dynamic information. We define damping control as a T_h-step optimal decision problem, which can be solved by MPC or RL in a way of receding horizon.

- In chapter 3, the underlying concept of MPC is described. Based on a linearized, discrete and time-invariant state space model, MPC prediction equations and objective function are derived. We get a quadratic programming problem that can be solved by Active Set methods, Interior Point methods, and so on. MPC stability and robustness must be carefully checked in practical applications. Finally, diverse MPC control schemes are discussed to offer the reference for our MPC designs.

- Chapter 4 focuses on a centralized MPC scheme for damping electromechanical oscillations. It firstly describes the proposed centralized MPC scheme, and gives its mathematical formulations in ideal conditions. Next, it considers the effects of state-estimation errors, communication and computation delays and non-controllability of local damping controllers. The scheme's ability of incorporating diverse damping controllers is tested.

- Chapter 5 introduces a distributed MPC scheme which is considered to be more viable for damping low-frequency oscillations in very large-scale interconnected power systems. It discusses the decomposition of a large-scale power system, and formulates MPC for a subsystem based on a given decomposition. Then, it analyzes the coordination between subsystem-wise MPC controllers. Results of distributed MPC are compared with those of centralized MPC.

- Chapter 6 proposes a hierarchical MPC scheme that introduces the new lower MPC controllers on the basis of distributed MPC. It firstly outlines the hierarchical MPC scheme, and then discusses the coupling between two layers of MPC and the coordination between lower MPC

controllers. Finally, it investigates the robustness and computation efficiency of this hierarchical MPC scheme.

- Chapter 7 firstly introduces the concept of learning and three learning forms: supervised learning, unsupervised learning and reinforcement learning. The rest of this chapter focuses on reinforcement learning, whose definition, elements and elementary solutions are detailedly discussed.

- Chapter 8 designs the supplementary damping control using a tree-based batch mode RL algorithm. It firstly describes this method in terms of extra-trees, fitted Q iteration and greedy decision making. Then it tests the method on a single generator, and multiple generators. Finally the combination between MPC and RL is investigated.

- Chapter 9 offers some conclusions and discusses some further works.

- Detailed information about the simulation tool and test system used for our investigations can be found in the appendices.

2.4 Summary

This chapter introduced our research motivation, namely to improve damping effects of existing controllers by using global information, and by adaptively adjusting and coordinating them. For this purpose, a trajectory-based supplementary damping control approach is proposed, considering the latest advances in power systems, machine learning and large-scale optimization, to compute supplementary inputs to superimpose on existing controllers to adjust and coordinate them. This approach aims at forcing the controlled system dynamics to evolve approximately along a desired trajectory, so as to damp electromechanical oscillations, with the help of certain supplementary control inputs. Mathematically, it is described as a T_h-step optimization control problem, which could be solved either by MPC or by RL. Finally, the contributions and structure of this thesis are listed.

Part II

Model predictive control for damping inter-area oscillations of large-scale power systems

Chapter 3

Model predictive control background

The content of the present chapter is adapted from [49, 64, 65]. Before presenting our MPC-based damping schemes, MPC's theoretical knowledge [49], advantages [64] and available control structures [65] are provided in order to make the thesis self-contained.

3.1 What is MPC ?

The principle of MPC can be shortly summarized as follows. At any time, the MPC algorithm uses the collected measurements, a model of the system and a specification of the control objective to compute an optimal open-loop control sequence over a specified time horizon. The computed optimal first-stage controls are applied to the system. At the next time step, as soon as measurements (or model) updates are available, the entire procedure is repeated by solving a new optimization problem with the time horizon shifted one step forward [49]. The MPC technology was originally developed for power plant and petroleum refinery applications, but can now be found in a wide variety of manufacturing environments including chemicals, food processing, automotive, aerospace, metallurgy, pulp and paper [64].

The basic idea of MPC for a single-input single-output system is shown in Fig 3.1 [49]. We assume a discrete-time setting, and that the current time is labeled as time step k. The set-point trajectory $s[t]$ is the trajectory that the real output $y[t]$ should follow. The solid line $y_{ref}[t|k]$ is termed

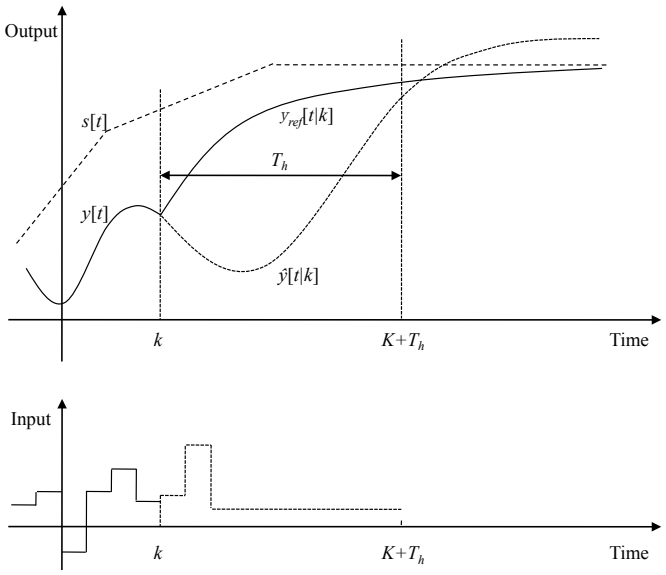

Figure 3.1: The basic idea of MPC. Taken from [49]

the reference trajectory, which starts at the current system output $y[k|k]$, and defines an ideal trajectory along which the system should return to the set-point trajectory.

At time k, a predictive controller collects current state $\hat{x}[k|k]$ and calculates the optimal input trajectory $\hat{u}[k+i|k]$ using a dynamic system model, in order to obtain the best possible future behavior $\hat{y}[k+i+1|k]$ with respect to a reference trajectory $y_{ref}[k+i+1|k]$, over a given prediction horizon T_h, $i = 0, 1, ..., T_h - 1$. A cost function is defined to assess what is the optimal sequence of inputs. In this example, $\hat{u}[k+i|k]$ varies over the first T_c steps, but remains constant thereafter. So the control horizon T_c is 3.

Once a future input trajectory $\hat{u}[k+i|k]$ is chosen, only the first element of the trajectory $\hat{u}[k|k]$ is applied as the real input $u[k|k]$ to the system. Then the whole cycle of measurement, prediction, and input trajectory determination is repeated at subsequent refreshing times: a new $\hat{x}[k+1|k+1]$ is obtained; a new reference trajectory $y_{ref}[k+1+i|k+1]$ is defined; predictions are made over the time horizon T_h; a new input trajectory $\hat{u}[k+1+i|k+1]$ is chosen, and finally the first input $\hat{u}[k+1|k+1]$ is applied to the system.

The success of MPC technology can be attributed to three important factors [64]. The first and foremost is the incorporation of an explicit process

model into the control calculation. This allows MPC, in principle, to deal directly with all significant features of process dynamics. Secondly the MPC algorithm considers system behaviors over a future temporal horizon. This means that the effects of feedforward and feedback disturbances can be anticipated and removed, to drive the controlled system to evolve more closely along a desired future trajectory. Finally the MPC controller considers process input, state and output constraints directly in the control calculation. This means that constraint violations are far less likely, resulting in tighter control at the optimal constrained steady-state.

3.2 Mathematical formulation of MPC

In the present section MPC's prediction equations, objective functions and constraints are formalized based on a linearized, discrete and time-invariant state space model in the form:

$$\begin{aligned} x[k+1|k] &= Ax[k|k] + Bu[k|k] \\ y[k|k] &= C_y x[k|k] \\ z[k|k] &= C_z x[k|k], \end{aligned} \tag{3.1}$$

where x is a d_x-dimensional state vector, u is a d_u-dimensional input vector, y is a d_y-dimensional vector of measured outputs, and z is a d_z-dimensional vector of outputs which are to be controlled, either to particular set-points, or to satisfy some constraints, or both. The index k counts 'time steps'. The variables in y and z usually overlap to a large extent, and frequently they are the same. That is to say, all the controlled outputs are frequently measured. So, we assume that $y \equiv z$, and we use C to denote both C_y and C_z. The Eqn. (3.1) can thus be rewritten as:

$$\begin{aligned} x[k+1|k] &= Ax[k|k] + Bu[k|k] \\ y[k|k] &= Cx[k|k]. \end{aligned} \tag{3.2}$$

In practice, we shall not assume that all state variables can be measured directly and exactly, so we use an estimated $\hat{x}[k|k]$ instead of the real state $x[k|k]$. Correspondingly, $\hat{x}[k+1|k]$ and $\hat{y}[k+1|k]$ denote the predictions of variables x and y at time $k+1$, on the assumption that one input $\hat{u}[k|k]$ is

applied at time k. So, Eqn (3.2) can be changed further into:

$$\hat{x}[k+1|k] = A\hat{x}[k|k] + B\hat{u}[k|k] \tag{3.3}$$
$$\hat{y}[k|k] = C\hat{x}[k|k]. \tag{3.4}$$

Now, we can predict the future system states over a given prediction horizon T_h by iterating Eqn. (3.3), and we get:

$$\begin{aligned}
\hat{x}[k+1|k] &= A\hat{x}[k|k] + B\hat{u}[k|k] \\
\hat{x}[k+2|k] &= A\hat{x}[k+1|k] + B\hat{u}[k+1|k] \\
&= A^2\hat{x}[k|k] + AB\hat{u}[k|k] + B\hat{u}[k+1|k] \\
&\vdots \\
\hat{x}[k+T_h|k] &= A\hat{x}[k+T_h-1|k] + B\hat{u}[k+T_h-1|k] \\
&= A^{T_h}\hat{x}[k|k] + A^{T_h-1}B\hat{u}[k|k] + ... + B\hat{u}[k+T_h-1|k].
\end{aligned} \tag{3.5}$$

We can re-express this equation in the matrix-vector form:

$$\begin{bmatrix} \hat{x}[k+1|k] \\ \hat{x}[k+2|k] \\ \vdots \\ \hat{x}[k+T_h|k] \end{bmatrix} = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{T_h} \end{bmatrix} \hat{x}[k|k] + \begin{bmatrix} B & & & \\ AB & B & & \\ \vdots & \vdots & \ddots & \\ A^{T_h-1}B & A^{T_h-2}B & \dots & B \end{bmatrix} \times \begin{bmatrix} \hat{u}[k|k] \\ \hat{u}[k+1|k] \\ \vdots \\ \hat{u}[k+T_h-1|k] \end{bmatrix}. \tag{3.6}$$

The input vector $\hat{u}[k+i|k]$ changes only at the first T_c steps, namely at the times $k, k+1, ..., k+T_c-1$, and will remain constant thereafter. That is to say, $\hat{u}[k+i|k] = \hat{u}[k+T_c-1]$ for $T_c \leq i \leq T_h - 1$. So, we combine the items in the input matrix corresponding to the same $\hat{u}[k+i|k]$ in Eqn. (3.6),

and obtain:

$$\begin{bmatrix} \hat{x}[k+1|k] \\ \hat{x}[k+2|k] \\ \vdots \\ \hat{x}[k+T_c|k] \\ \vdots \\ \hat{x}[k+T_h|k] \end{bmatrix} = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{T_c} \\ \vdots \\ A^{T_h} \end{bmatrix} \hat{x}[k|k] + \begin{bmatrix} B & & & \\ AB & B & & \\ \vdots & \vdots & \ddots & \\ A^{T_c-1}B & A^{T_c-2} & \dots & B \\ \vdots & \vdots & \vdots & \vdots \\ A^{T_h-1}B & A^{T_h-2}B & \dots & \sum_{i=0}^{T_h-T_c} A^i B \end{bmatrix} \begin{bmatrix} \hat{u}[k|k] \\ \hat{u}[k+1|k] \\ \vdots \\ \hat{u}[k+T_c-1|k] \end{bmatrix}. \tag{3.7}$$

In the predictive control, we often compute the input change $\Delta\hat{u}[k+i|k]$ rather than $\hat{u}[k+i|k]$. Defining that $\Delta\hat{u}[k+i|k] = \hat{u}[k+i|k] - \hat{u}[k+i-1|k]$, we have:

$$\begin{bmatrix} \hat{u}[k|k] \\ \hat{u}[k+1|k] \\ \vdots \\ \hat{u}[k+T_c-1|k] \end{bmatrix} = \begin{bmatrix} I \\ I \\ \vdots \\ I \end{bmatrix} u[k-1] + \begin{bmatrix} I & & & \\ I & I & & \\ \vdots & \vdots & \ddots & \\ I & I & \dots & I \end{bmatrix} \times \begin{bmatrix} \Delta\hat{u}[k|k] \\ \Delta\hat{u}[k+1|k] \\ \vdots \\ \Delta\hat{u}[k+T_c-1|k] \end{bmatrix}. \tag{3.8}$$

Here, $u[k-1]$ is known. Substituting Eqn. (3.8) into Eqn. (3.7) we obtain:

$$\begin{bmatrix} \hat{x}[k+1|k] \\ \hat{x}[k+2|k] \\ \vdots \\ \hat{x}[k+T_c|k] \\ \vdots \\ \hat{x}[k+T_h|k] \end{bmatrix} = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{T_c} \\ \vdots \\ A^{T_h} \end{bmatrix} \hat{x}[k|k] + \begin{bmatrix} B \\ AB+B \\ \vdots \\ \sum_{i=0}^{T_c-1} A^i B \\ \vdots \\ \sum_{i=0}^{T_h-1} A^i B \end{bmatrix} u[k-1] + \\ \begin{bmatrix} B & & & \\ AB+B & B & & \\ \vdots & \vdots & \ddots & \\ \sum_{i=0}^{T_c-1} A^i B & \sum_{i=0}^{T_c-2} A^i B & \dots & B \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{i=0}^{T_h-1} A^i B & \sum_{i=0}^{T_h-2} A^i B & \dots & \sum_{i=0}^{T_h-T_c} A^i B \end{bmatrix} \begin{bmatrix} \Delta\hat{u}[k|k] \\ \Delta\hat{u}[k+1|k] \\ \vdots \\ \Delta\hat{u}[k+T_c-1|k] \end{bmatrix}. \tag{3.9}$$

Finally, the predictions of y over the whole prediction horizon are given by:

$$\begin{bmatrix} \hat{y}[k+1|k] \\ \hat{y}[k+2|k] \\ \vdots \\ \hat{y}[k+T_h|k] \end{bmatrix} = \begin{bmatrix} C & & & \\ 0 & C & & \\ \vdots & \vdots & \ddots & \\ 0 & 0 & \dots & C \end{bmatrix} \begin{bmatrix} \hat{x}[k+1|k] \\ \hat{x}[k+2|k] \\ \vdots \\ \hat{x}[k+T_h|k] \end{bmatrix}. \tag{3.10}$$

We can rewrite Eqn. (3.9) and (3.10) as

$$X[k] = P_x \hat{x}[k|k] + P_u u[k-1] + P_{\Delta u} \Delta U[k] \tag{3.11}$$

$$Y[k] = P_y X[k] = P_y P_x \hat{x}[k|k] + P_y P_u u[k-1] + P_y P_{\Delta u} \Delta U[k] \tag{3.12}$$

where:

$$P_x = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{T_h} \end{bmatrix} \quad P_u = \begin{bmatrix} B \\ AB+B \\ \vdots \\ \sum_{i=0}^{T_h-1} A^i B \end{bmatrix} \quad P_y = \begin{bmatrix} C & & & \\ 0 & C & & \\ \vdots & \vdots & \ddots & \\ 0 & 0 & \dots & C \end{bmatrix}$$

$$P_{\Delta u} = \begin{bmatrix} B & & & \\ AB+B & B & & \\ \vdots & \vdots & \ddots & \\ \sum_{i=0}^{T_c-1} A^i B & \sum_{i=0}^{T_c-2} A^i B & \dots & B \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{i=0}^{T_h-1} A^i B & \sum_{i=0}^{T_h-2} A^i B & \dots & \sum_{i=0}^{T_h-T_c} A^i B \end{bmatrix}.$$

Based on the above prediction equations, the MPC seeks for an optimal sequence of $\hat{u}[k+i|k]$ which minimizes a given cost function. A typical cost function is given in Eqn. (3.13). It penalizes the deviation of the predicted controlled output $\hat{y}[k+i+1|k]$ from a reference trajectory $y_{ref}[k+i+1|k]$ and the input change $\Delta\hat{u}[k+i|k]$. The reference trajectory $y_{ref}[k+i+1|k]$ may be some predetermined trajectories:

$$V[k] = \sum_{i=0}^{T_h-1} ||\hat{y}[k+i+1|k] - y_{ref}[k+i+1|k]||^2_{W_{y_i}} + \sum_{i=0}^{T_c-1} ||\Delta\hat{u}(k+i|k)||^2_{W_{\Delta u_i}} \tag{3.13}$$

Actually we do not necessarily start penalizing the deviation of $\hat{y}[k+i+1|k]$ from $y_{ref}[k+i+1|k]$ immediately because there may be some delay between applying an input and seeing any effect. So, the i of the first item in the right part of Eqn. (3.13) could begin from a value larger than 0. W_{y_i} and W_{u_i} are weight matrices. The solution minimizing Eqn. (3.13) should be subject to the following inequality constraints:

$$\begin{aligned} E_{\Delta\hat{u}}\ vec(\Delta\hat{u}[k|k], ..., \Delta\hat{u}[k+T_c-1|k], 1) &\le vec(0) \\ E_{\hat{u}}\ vec(\hat{u}[k|k], ..., \hat{u}[k+T_c-1|k], 1) &\le vec(0) \\ E_{\hat{y}}\ vec(\hat{y}[k+H_w|k], ..., \hat{y}[k+T_h|k], 1) &\le vec(0). \end{aligned} \tag{3.14}$$

Here, $vec(0)$ denotes a column vector, each element of which is 0; $E_{\Delta\hat{u}}$, $E_{\hat{u}}$ and $E_{\hat{y}}$ are matrices of suitable dimensions. We can use the constraints of Eqn. (3.14) to represent possible actuator slew rates, actuator ranges and constraints on the controlled variables. These constraints hold over the control horizon and prediction horizon. Actually, when we solve the predictive

control optimization problem, all above inequalities must be translated into the inequalities concerning $\Delta\hat{u}[k+i|k]$, namely

$$E(\Delta\hat{u}[k|k], \Delta\hat{u}[k+1|k], ..., \Delta\hat{u}[k+T_c-1|k]) = E\Delta U[k] \leq e. \tag{3.15}$$

Here, E is a coefficient matrix of proper dimensions.

3.3 Solutions to an MPC problem

3.3.1 Mathematical conversion of MPC optimization

Firstly, we rewrite Eqn. (3.13) as:

$$V[k] = ||Y[k] - Y_{ref}[k]||^2_{W_y} + ||\Delta U[k]||^2_{W_{\Delta u}} \tag{3.16}$$

where

$$Y[k] = \begin{bmatrix} \hat{y}[k+1|k] \\ \vdots \\ \hat{y}[k+T_h|k] \end{bmatrix},\ Y_{ref}[k] = \begin{bmatrix} y_{ref}[k+1|k] \\ \vdots \\ y_{ref}[k+T_h|k] \end{bmatrix},\ \Delta U[k] = \begin{bmatrix} \Delta\hat{u}[k|k] \\ \vdots \\ \Delta\hat{u}[k+T_c-1|k] \end{bmatrix}$$

and the weight matrices W_y and $W_{\Delta u}$ are given by:

$$W_y = \begin{bmatrix} W_{y_1} & & \\ & \ddots & \\ & & W_{y_{T_h}} \end{bmatrix},\ W_{\Delta u} = \begin{bmatrix} W_{\Delta u_0} & & \\ & \ddots & \\ & & W_{\Delta u_{T_c-1}} \end{bmatrix}.$$

Let us substitute Eqn. 3.12 into Eqn. 3.16. And before doing it, we firstly rewrite Eqn. 3.12 in order to obtain a simpler form.

$$Y[k] = P_{y.x}\hat{x}[k|k] + P_{y.u}u[k-1] + P_{y.\Delta u}\Delta U[k] \tag{3.17}$$

where $P_{y.x} = P_y P_x, P_{y.u} = P_y P_u, P_{y.\Delta u} = P_y P_{\Delta u}$.

Now, we have:

$$\begin{aligned} V[k] &= ||P_{y.x}\hat{x}[k|k] + P_{y.u}u[k-1] + P_{y.\Delta u}\Delta U[k] - Y_{ref}[k]||^2_{W_y} + ||\Delta U[k]||^2_{W_{\Delta u}} \\ &= (\hat{x}^T[k|k]P^T_{y.x} + u^T[k-1]P^T_{y.u} - Y^T_{ref}[k])W_y(P_{y.x}\hat{x}[k|k] + P_{y.u}u[k-1] \\ &- Y_{ref}[k]) + 2(\hat{x}^T[k|k])P^T_{y.x} + u^T[k-1]P^T_{y.u} - Y^T_{ref}[k])W_y P_{y.\Delta u}\Delta U[k] \\ &+ \Delta U[k]^T(P^T_{y.\Delta u}W_y P_{y.\Delta u} + W_{\Delta u})\Delta U[k] \\ &= const + 2(F^T\Delta U[k] + 1/2\Delta U^T[k]G\Delta U[k]) \end{aligned} \tag{3.18}$$

where

$$\begin{aligned} const &= (\hat{x}[k|k]^T P_{y.x}^T + u[k-1]^T P_{y.u}^T - Y_{ref}^T[k]) W_y (P_{y.x}\hat{x}[k|k] \\ &\quad + P_{y.u}u[k-1] - Y_{ref}[k]) \\ F &= P_{y.x}\hat{x}[k|k] + P_{y.u}u[k-1] - Y_{ref}[k] \\ G &= P_{y.\Delta u}^T W_y P_{y.\Delta u} + W_{\Delta u}. \end{aligned} \tag{3.19}$$

The objective of MPC is to minimize

$$V'[k] = F^T \Delta U[k] + 1/2 \Delta U[k]^T G \Delta U[k] \tag{3.20}$$

subject to

$$\begin{aligned} E\Delta U[k] &\leq e \\ H\Delta U[k] &= h. \end{aligned} \tag{3.21}$$

Except the inequality constraints, there are also some equality constraints as shown in Eqn. (3.21). This is a standard optimization problem known as Quadratic Programming (QP), which can be solved by methods of Interior Point, Active Set, Augmented Lagrangian, Conjugate Gradient, and so on [49, 66, 67].

3.3.2 Solution approaches

The Active Set methods assume that a feasible solution is available at the beginning. For such a solution, the subset of the inequality constraints of Eqn. (3.21) that are active is called the active set, denoted by e_a. That is to say, $E_a \Delta U[k] = e_a$. It is possible for the active set to be empty, if none of the inequality constraints is active at the current feasible solution.

Suppose that at the jth iteration we have a feasible solution $\Delta U_j[k]$. The Active Set method then finds an improved solution $\Delta U_{j+1}[k] = \Delta U_j[k] + \Delta_{\Delta U_j[k]}$ which minimizes the cost function of Eqn. (3.20) while satisfying $H\Delta U_{j+1}[k] = h$ and $E_a \Delta U_{j+1}[k] = e_a$, without considering the inactive inequality constraints. If this new solution is feasible, that is if $E\Delta U_{j+1}[k] \leq e$, then it is accepted as the feasible solution of next iteration. If it is not feasible, then a line-search is made in the direction $\Delta_{\Delta U_j[k]}$ to locate the point at which the feasibility is lost-namely the point at which one of the inactive inequality constraints becomes active. The solution at this point is accepted as the feasible solution of next iteration, namely $\Delta U_{j+1}[k] =$

$\Delta U_j[k] + \alpha_j \Delta_{\Delta U_j[k]}$, where $0 < \alpha_j < 1$, and the newly active constraint is added to the active set. We can judge whether this new solution is already the global optimum of the QP problem, or whether further improvement can be obtained by checking whether $\Delta U_{j+1}[k]$ satisfies the *Karush-Kuhn-Tucker* (KKT) conditions.

Another kind of prevalent approach for solving the above QP problem is the Interior Point method. Different with the Active Set methods that search the optimal solution among the points on the boundary of a feasible region, the Interior Point methods search the optimal solution among the points in the interior of the feasible region. They first transfer the optimization problem with inequality constraints into the unconstrained optimization problem by introducing a logarithmic penalty factor, and then iteratively solve the unconstrained optimization to search a minimum. The logarithmic penalty factor prevents the search of minimum solution from leaving the feasible region, since it becomes infinite on the boundary of the region.

We must notice that we cannot find a solution satisfying the constraints using the above methods in some cases. It might be caused by large disturbances, internal model errors and other unanticipated factors. One systematic strategy for dealing with the infeasibility is to 'soften' the constraints. That is to say, rather than regard the constraints as 'hard' boundaries which can never be crossed, allow them to be crossed occasionally, but only if really necessary. Usually, we only soften the output constraints because the input constraints are really hard constraints and there is no way in which they can be softened like actuator slew rates, except for fitting more powerful actuators. We can add 'slack variables' to soften output constraints which are defined non-zero only if the constraints are violated. Then their non-zero values are very heavily penalized in the cost function, so that the optimizer has a strong incentive to keep them at zero if possible.

3.4 MPC stability and robustness

3.4.1 MPC stability

MPC, using the receding horizon idea, is a feedback control policy solved over a finite time horizon as an approximation of an infinite time horizon control problem. There is therefore a risk that the resulting closed loop might be unstable. Even though the performance of the plant is being optimized

over the prediction horizon, and even though the optimization keeps being repeated, each optimization does not care about what happens beyond the prediction horizon, and so could be putting the plant into such a state that it will eventually be impossible to stabilize. This is particularly likely to occur when there are constraints on the possible input signals.

It turns out that the stability can theoretically be ensured by choosing a convenient combination of optimization criterion and time-step, and/or by making the prediction horizon long enough, or even infinite. But considering the computational feasibility, MPC only could use a finite prediction horizon, or reformulate an infinite horizon control problem into one with a finite and small enough number of optimization variables, so as to ensure small enough computing times, given the duration of each time step. In our work, we have determined a proper time-step and prediction horizon in an empirical way by trial and error under different simulation settings.

3.4.2 MPC robustness

The robustness means that the MPC stability is maintained and that the performance specifications are met for a specified range of uncertainties [68]. The uncertainties may come from measurement noises, disturbances and model variations. In our work, we will check empirically the robustness of the proposed MPC schemes in terms of measurement errors, delays, different operation conditions, communication failures, controller failures and so on.

The interested readers are referred to [49] for more information about MPC stability and robustness.

3.5 MPC control structures

For large-scale power systems, it is very difficult to design a fully centralized control structure due to computational complexity, robustness and reliability considerations, and communication limitations. So, we next focus on discussing several distributed and hierarchical MPC structures proposed in the literature [65].

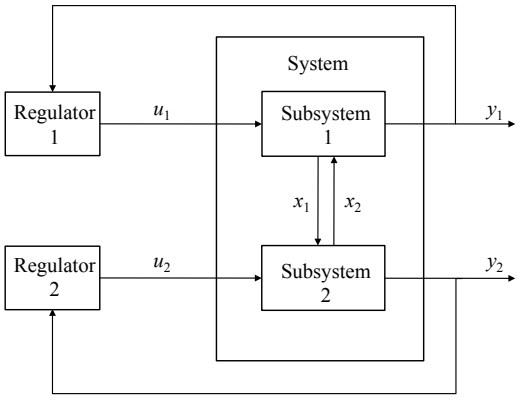

Figure 3.2: Decentralized control. Taken from [65]

3.5.1 Decentralized control

As shown in Fig. 3.2, a large system under control is assumed to be composed by two subsystems 1 and 2. The input variables and the controlled output variables are grouped into the disjoint sets (u_1, y_1) and (u_2, y_2). Two subsystems interact due to the mutual effects of states x_1 and x_2 [69].

Two local regulators 1 and 2 are designed to operate in a completely independent fashion. They can be single-input single-output or multivariable (locally centralized) depending on the cardinality of the selected input and output groups. When the interaction among subsystems is weak, it is not difficult to compute local control laws using the standard MPC algorithms by neglecting the mutual interaction. But on the contrary, it is well known that the strong interaction can prevent local regulators from achieving the stability and/or the given performances [70].

3.5.2 Distributed control

In contrast with the decentralized control, some information is transmitted among the local regulators in the distributed control so that each one of them has some knowledge on the behavior of the other, please see Fig. 3.3. When the local regulators are designed with MPC, the information transmitted typically consists of the predicted future control or state variables, so that any local regulator can predict the interaction effects over the considered prediction horizon. Depending on the topology of communication network, the distributed control can be divided into:

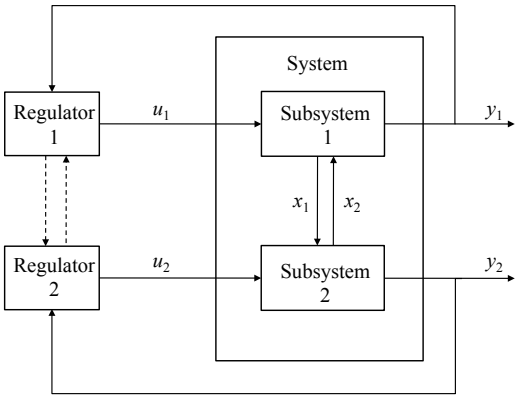

Figure 3.3: Distributed control. Taken from [65]

- Full connection: the information is transmitted from any local regulator to all the others.
- Partial connection: the information is transmitted from any local regulator to a given subset of the others.

The exchange of information among local regulators can be made in the following ways:

- Noniterative exchange: the information is transmitted by local regulators only once at each sampling time.
- Iterative exchange: the information can be transmitted by local regulators many times within a sampling interval.

For local regulators, they can use two kinds of different control objectives.

- Local objective: each local regulator minimizes a local performance index.
- Global objective: each local regulator minimizes a global cost function.

3.5.3 Hierarchical control for coordination

An example of two-level hierarchical control structure is shown in Fig. 3.4. The system is decomposed into two subsystems linked through some interconnecting variables x_1 and x_2. Each subsystem solves its control problem

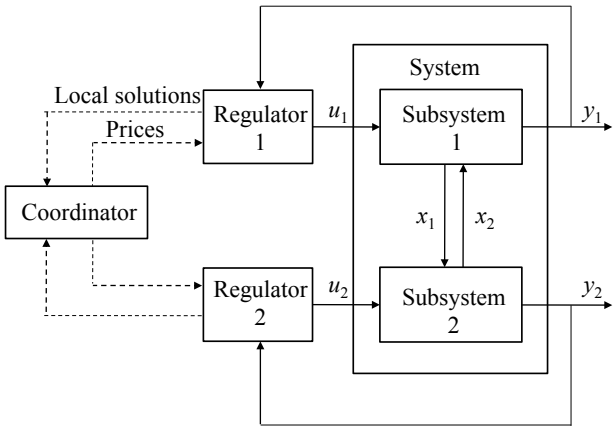

Figure 3.4: Hierarchical control for coordination. Taken from [65]

by minimizing a suitable local cost function subject to local state, input and output constraints. The coordinator at the higher level is responsible to coordinate local regulators: the coordinator sets the prices, which coincide with the Lagrange multipliers of the coherence constraints in the global optimization problem. In turn, these optimal prices are sent to the low-level local optimizers which take them as the reference and recompute the optimal trajectories of the state, input and output variables over the considered prediction horizon. The iterations are stopped when the interconnecting variables satisfy the required coherence conditions. This conceptual iterative procedure must be specialized to guarantee its convergence as well as some properties of the resulting final solution.

3.5.4 Hierarchical control of multilayered systems

In the hierarchical multilayered systems, the control is performed by a number of controllers working at different time scales, i.e. slow and fast dynamics, as shown in Fig. 3.5. A regulator acting at lower frequencies computes both the control actions (u_{slow}) of manipulated variables which have long-term effects on the system, i.e. the slow control variables, and the reference values of fast control variables, states and outputs ($u^{ref}_{fast}, x^{ref}_{fast}, y^{ref}_{fast}$). A second regulator takes these reference values as inputs and computes the fast control variables u_{fast} solving a tracking problem at a higher rate.

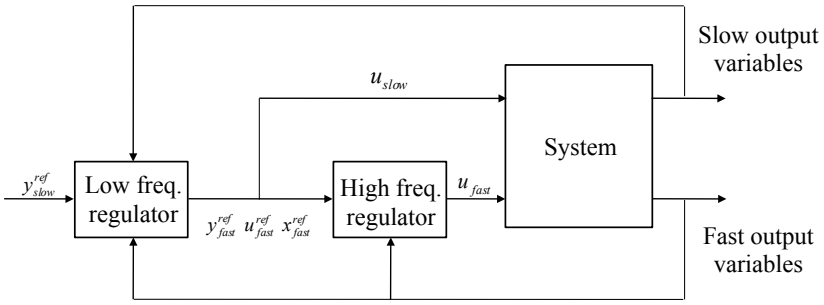

Figure 3.5: Hierarchical control of multilayer systems. Taken from [65]

3.6 MPC applications to power systems

Security constrained optimal power flow [71–74] and coordinated secondary voltage control [75] are the two applications that pioneered MPC ideas in the context of bulk power systems control.

The publication [71] defines the security-constrained economic dispatch problem as a cost minimization problem, subject to operating constraints and security constraints for a list of given disturbances. It first searches an operating point without considering the post-disturbance security. And then, different corrective controls, like generation rescheduling and network switching, are applied to eliminate the violations of post-disturbance security by minimizing a penalty function to zero. More recent developments and several new variants of security constrained optimal power flow algorithms targeting very large-scale power systems are discussed in [72–74], as well as in some of the references cited in these papers.

The publication [75] decomposes a secondary voltage control problem into two sequential parts: 1) static level: calculate the new static objectives that are consistent with the current network states and constraints; 2) dynamic level: this above calculation is updated as soon as a change in the network states is detected. The proposed predictive control scheme divides transmission delays into synchronous delays and asynchronous delays, and treats them respectively in order to improve its dynamic robustness against uncertain transmission delays.

MPC was also more recently proposed for automatic generation control [76, 77], emergency alleviation of thermal overloads [78, 79], voltage control [80–85], transient stability control [86, 87] and electromechanical oscillations damping [37, 88–91]:

- Thermal overload control: based on a DC load flow model, paper [78] computes a sequence of discrete controls which minimize the control cost and are subject to line power flow constraints and control constraints. It only applies the first-stage controls of the computed sequence, and continuously updates the control signals at subsequent control times to bring the currents in overloaded transmission lines below their admissible values, before these lines are taken out of service by the protection system. Paper [79] extends the MPC approach of paper [78] by introducing an accurate thermal model of bare overhead conductors and dynamic stochastic weather models to more exactly predict the temperatures of transmission lines.

- Voltage control: paper [84] introduces the voltage stability margin as security constraint to MPC mathematical formulations, to prevent voltage instability and maintain a desired amount of post-disturbance transient voltage stability margin. Paper [85] uses explicit formulations to model the evolution of load power with time, and then solves a control cost minimization problem at multiple steps and in a receding horizon way so as to correct nonviable or unstable transmission voltages.

- Transient stability control: at each control time, paper [86] repeatedly computes a sequence of FACTs controls minimizing the terminal cost and applies the first one of this sequence to yield the first swing stability after a large disturbance.

- Electromechanical oscillations damping: one of the earliest MPC applications to electromechanical oscillations damping is presented in [88] where generalized predictive control is used to switch capacitors for damping power system oscillations. The control strategy is computed by minimizing a quadratic cost function combining local system outputs and rates-of-change of control over the prediction horizon. An MPC for step-wise series reactance modulation of a TCSC to stabilize electromechanical oscillations is presented in [89], where a reduced two-machine model of the power system is used and updated using local measurements. Defining deviations of the predicted outputs from references and control input increments as an objective function, reference [90] proposed a model predictive adaptive controller based on an equivalent model to damp inter-area oscillations in a four-generator

system. References [37, 91] present a wide-area MPC to control low-frequency oscillations. A bank of linearized system models is used with the assumption that the actual system response can be represented by a suitable combination of a finite number of linearized models. For each model in the bank an observer-based state feedback controller is designed a priori and MPC is formulated to optimize the weights for individual controllers in the bank.

3.7 Conclusions

In this chapter, the underlying concept of MPC is introduced and the reasons of its success are analyzed. Detailed prediction equations and the cost function are listed based on a linearized, time-invariant, discrete state space model. This leads to a Quadratic Programming problem that could be solved by Active Set methods, Interior Point methods, and so on. Several available distributed and hierarchical control structures are discussed to provide the reference for our MPC design. The above contents are adapted from [49], [64] and [65]. Finally a review of the most important applications of MPC to power systems is made.

Chapter 4

Centralized MPC scheme

4.1 Introduction

In this chapter, a new centralized MPC scheme, based on a linearized, discrete-time state space model combined with a quadratic objective function, is proposed to coordinate local control devices (PSSs, TCSCs and SVCs) to damp wide-area electromechanical oscillations.

The content of this chapter is mainly based on our publication

[58] *D. Wang, M. Glavic, and L. Wehenkel. A new MPC scheme for damping wide-area electromechanical oscillations in power systems. In Proceedings of the 2011 IEEE PES PowerTech, Trondheim, Norway, June 2011.*

The results presented in this chapter aim at evaluating the potential of MPC for electromechanical oscillations damping, in ideal conditions where an accurate and complete system model is available, and where some authority would be able to exploit system wide measurements in order to simultaneously control all available devices throughout the system. While not necessarily practical, the performances of this centralized scheme will be useful to assess and compare the more practical schemes developed in the subsequent chapters of this thesis. To avoid overoptimistic results, we also asses the effect of measurement errors and communication delays.

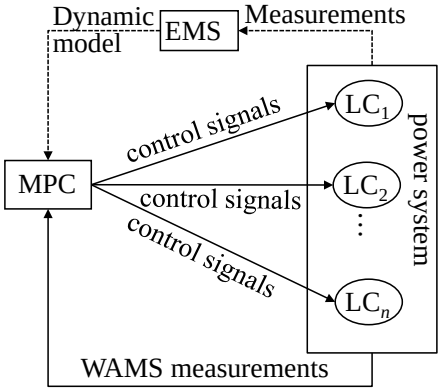

Figure 4.1: Schematic diagram of the centralized MPC scheme

4.2 Outline of the proposed centralized MPC scheme

Fig. 4.1 illustrates our centralized MPC scheme. In this scheme, the MPC controller obtains a complete system model from a system-wide Energy Management Systems (EMS), which can be refreshed from time to time following changes of the load level, of the generation schedule and/or of the grid topology. It collects system states from a state estimator[1] fed by a WAMS at discrete measurement times $k\Delta t$. It then computes an open loop sequence of the control variables u over a certain time horizon. It applies the controls determined for the first period of Δt seconds to local controllers (LCs), and then waits for the next set of measurements to be received in order to start this computation again.

Next, we give the detailed MPC formulation in terms of the model (prediction equations) and optimal control problem (objective function and constraints). Then data acquisition errors and time delays due to computation and communication resources are taken into account, as well as the possibility of non-response of some of the local controllers.

[1]To get the actual system states from WAMS measurements, one normally would need a state estimator, but the detailed description of the WAMS and corresponding state-estimator technologies is out of the scope of the thesis.

4.3 Mathematical formulation

4.3.1 Discrete time linearized dynamic system model

The proposed MPC scheme is based on a state-space dynamic model of a multi-machine power system in the form of the following linearized continuous time model:

$$\begin{cases} \dot{x} = A_c x + B_c u \\ y = C_c x \end{cases} \tag{4.1}$$

where x is a vector of state variables modeling also the already existing controllers, u is a vector of supplementary MPC controls, and y is a vector of performance measurements used by MPC (outputs).

Next, from (4.1) the transition at time t for a small step of $\hat{\delta}$ seconds is inferred:

$$\begin{cases} x[t+\hat{\delta}] = (\hat{\delta}A_c + I)x[t] + \hat{\delta}B_c u[t] \\ y[t] = C_c x[t] \end{cases} \tag{4.2}$$

yielding a discrete-time dynamics (for time step $t + i\hat{\delta}$) given by:

$$\begin{aligned} &x[i+1] = Ax[i] + Bu[i]; \\ &\quad y[i] = Cx[i]. \\ &\text{with: } A = (\hat{\delta}A_c + I); B = \hat{\delta}B_c; C = C_c. \end{aligned} \tag{4.3}$$

4.3.2 MPC formulation as a quadratic programming problem

At time $t = k\Delta t$ (system states are collected every Δt seconds), based on the estimated state $\hat{x}[k|k]$ of the current system states and on the system model, the predicted output $\hat{y}[k+i+1|k]$ over the next horizon of $T_h\hat{\delta}$ is obtained by iterating (4.3) i times, $i = 0, 1, 2, \ldots T_h - 1$.

$$\begin{bmatrix} \hat{y}[k+1|k] \\ \hat{y}[k+2|k] \\ \vdots \\ \hat{y}[k+T_h|k] \end{bmatrix} = P_x\hat{x}[k|k] + P_u \begin{bmatrix} \hat{u}[k|k] \\ \hat{u}[k+1|k] \\ \vdots \\ \hat{u}[k+T_h-1|k] \end{bmatrix} \tag{4.4}$$

where P_x and P_u are given by

$$P_x = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{T_h} \end{bmatrix}, \; P_u = \begin{bmatrix} CB & 0 & \dots & 0 \\ CAB & CB & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{T_h-1}B & CA^{T_h-2}B & \dots & CB \end{bmatrix}$$

Using these equations, the following quadratic optimization problem is solved at every time step [49]:

$$\begin{aligned} \min_{u[\cdot]} \sum_{i=0}^{T_h-1} & (\hat{y}[k+i+1|k] - y_{ref}[k+i+1|k])^T W_{y_i} \\ & (\hat{y}[k+i+1|k] - y_{ref}[k+i+1|k]) \end{aligned} \tag{4.5}$$

subject to linear inequality constraints:

$$\begin{aligned} & u_{min} \leq \hat{u}[k+i|k] \leq u_{max}, i = 0, 1, 2, ...T_h - 1. \\ & z_{min} \leq \hat{z}[k+i+1|k] \leq z_{max}, i = 0, 1, 2, ...T_h - 1. \end{aligned} \tag{4.6}$$

where W_{y_i} is a symmetric positive definite matrix of weights, y_{ref} is the vector of performance targets, and $\hat{z}$ is a vector of constrained operation variables. Here, we use two kinds of time steps: $\hat{\delta}$ is the time step of representing MPC dynamics; Δt is the time step of measuring, computing and controlling. For simplicity, Δt is the integral multiple of $\hat{\delta}$. During one Δt, it is assumed that $\hat{u}[k+i|k]$ remains constant.

The *Quadprog*, a MATLAB function, is used to compute the solution of this quadratic programming problem. The first part of this solution is injected into the local controllers (LCs, in Fig. 4.1), and the calculation is repeated at the next measurement step $(k+1)\Delta t$.

4.4 Simulation results in ideal conditions

In this section, we present the results in ideal conditions where the controlled system has complete state observability and controllability, and state-estimation (SE) errors and communication and computing delays are also ignored.

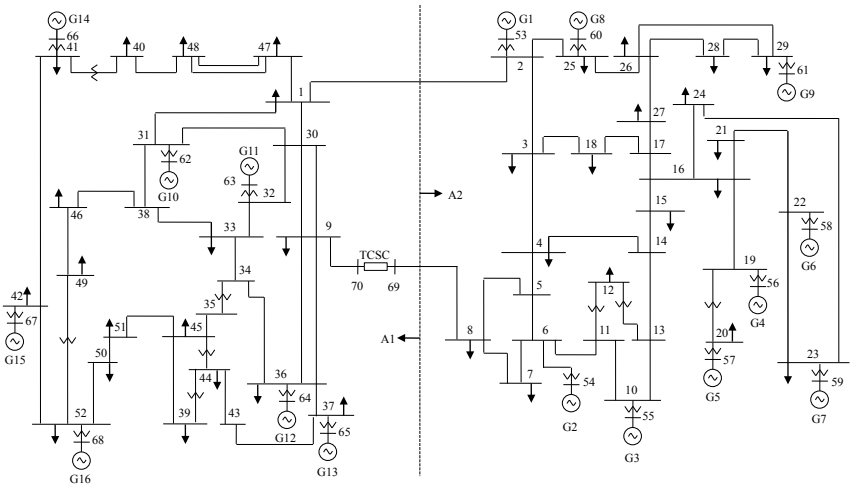

Figure 4.2: 16 generators / 2 areas / 70 buses test system

4.4.1 Test system and simulation parameters

The centralized MPC scheme is tested on a 16 generators, 70 buses system, as shown in Fig. 4.2. Power System Toolbox (PST) [1,92] is used to simulate system response and to derive a linearized state space model. More details about test system and PST are given in appendices. A TCSC is installed between buses 69 and 70, and there is a PSS on each generator. The system is composed of two areas: A1 and A2, which are connected to each other by the tie-lines 1-2 and 8-9, the latter being equipped with a TCSC. To assess the dynamics of the power system, we simulate and observe its response over a period of 20s. In the presented tests, a temporary three-phase short-circuit to ground at bus 1 (cleared by opening the tie-line 1-2 followed by its reconnection after a short delay) causes oscillations between area A1 and area A2. As shown in Fig. 4.3, the temporal evolution over a period of 20s of the power flow through line 1-2 (P_{1-2}) and the angular speeds of all generators corresponds to sustained, poorly damped oscillations.

Further details concerning this test system are provided in Appendix A. The PST is described in appendix B, while the dynamic models used in all our simulations are described in Appendix C.

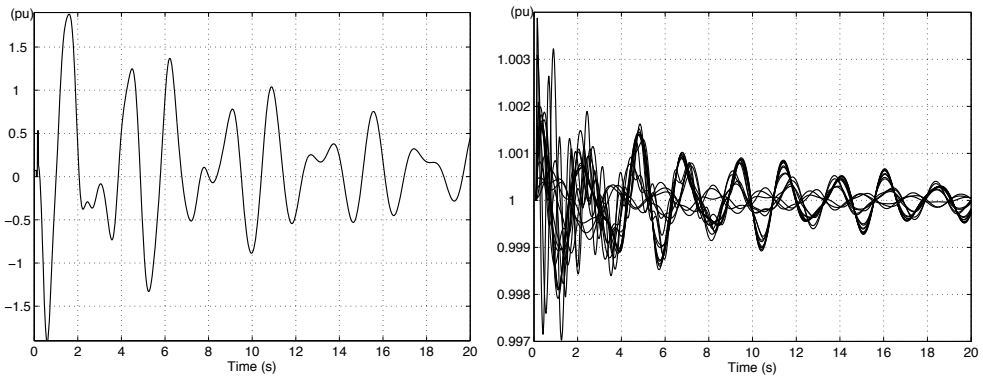

Figure 4.3: Response without MPC: P_{1-2} (left), and angular speeds (right)

4.4.2 MPC control effects

We next introduce a centralized MPC controller to improve damping control effects of existing PSSs and TCSC. Fig. 4.4 illustrates two types of local controllers considered and the way supplementary controls computed by MPC are brought to local controllers. The MPC state vector x contains 6×16 generator states, 3×16 exciter states, 3×16 PSS states, 3×16 turbine governor states and 1 TCSC state. The output y is the vector of angular speeds, which reference y_{ref} is a unit vector of dimension 16. The input vector u consists of 17 supplementary signals: 16 of them for PSSs and one for TCSC, which is constrained to $-0.1 \leq u \leq 0.1$.

Every $\Delta t = 0.1$s, the MPC controller collects system states, calculates control sequence and applies the first-stage control. The size of prediction horizon has an appreciable influence on MPC performances. After trying different prediction horizons, we use in our simulations a prediction horizon of 1.5 seconds while we use a time step $\hat{\delta} = 0.005$ seconds for formulating the MPC dynamics and optimization problem. In the objective function (4.5), all deviations of the predicted outputs from references are weighted uniformly and independently, i.e. W_{y_i} is a 16×16 identity matrix. In order to reduce computing times for solving the MPC optimization problem, we further impose that u over the horizon of 1.5 seconds may take only three different values, namely $\hat{u}[k+i|k] = u_0$ for the first 20 steps (0.1 seconds), $\hat{u}[k+i|k] = u_1$ for the next 20 steps (0.1 seconds), and $\hat{u}[k+i|k] = u_2$ for the remaining 260 steps (1.3 seconds). That is to say, the control horizon T_c is 3.

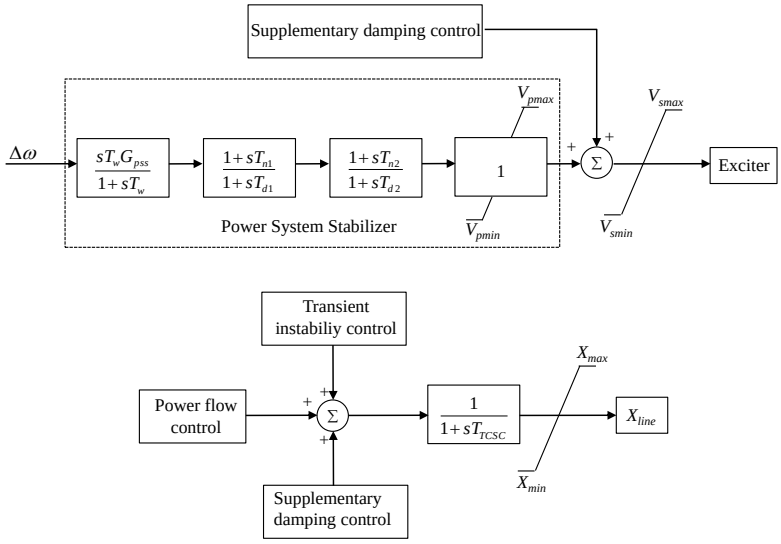

Figure 4.4: Supplementary control inputs: PSS (top), TCSC (bottom)

This reduces significantly the computational burden[2] without jeopardizing the performances.

Controlled system response over a time period of 20s in terms of the power flow through tie-line 1-2 and the angular speed of generator 1 is given in Fig. 4.5. In the subsequent parts, only the effects of different control schemes and assumptions on the tie-line 1-2 power flow (P_{1-2}) and on the angular speed of generator 1 (w_{g1}) are shown. These are indeed representative of the overall dynamics of the system.

Compared with the system response without MPC (dashed line), we clearly observe that the proposed MPC scheme damps oscillations of the tie-line 1-2 transmission power and the angular speed of generator 1, more efficiently (settling time is decreased to approximately 10 seconds). Fig. 4.6 shows the supplementary control signals computed by the MPC scheme for a representative PSS and the TCSC (first $5s$).

[2]by reducing the number of optimization variables from $17\times300 = 51,000$ to $17\times3 = 51$

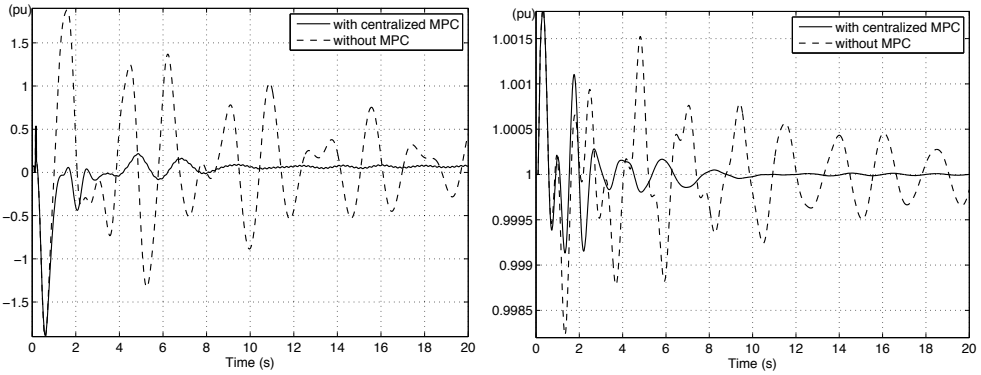

Figure 4.5: Response of centralized MPC in ideal conditions: P_{1-2} (left), and w_{g1} (right)

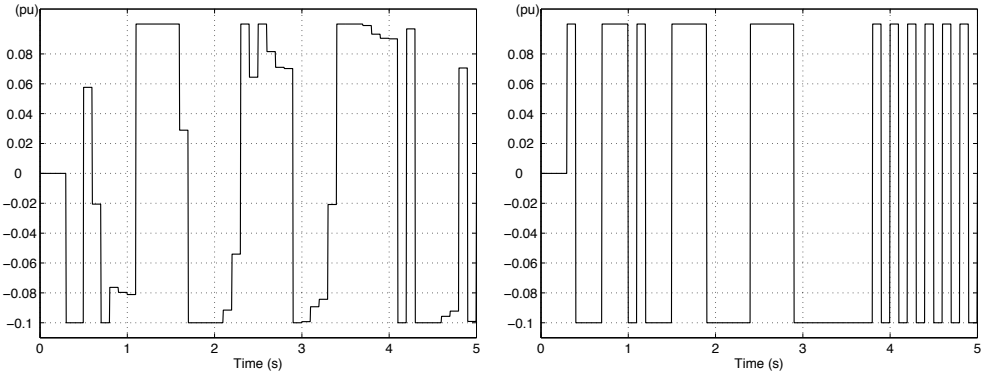

Figure 4.6: Centralized MPC signals: PSS of generator 1 (left), and TCSC (right)

4.5 Modeling of state-estimation errors

4.5.1 MPC formulation involving SE errors

At any refreshment time $k\Delta t$, the MPC controller uses the results $\hat{x}[k|k]$ from a State Estimator (SE) to compute the optimized supplementary signals for existing PSSs and TCSC. Consequently, the SE imprecision may have a detrimental effect on the MPC controller's decisions.

In order to compensate for this imprecision, at time $k\Delta t$, the difference $\hat{d}[k|k]$ between the actual output and its predicted value is calculated to correct Eqn. (4.4) as:

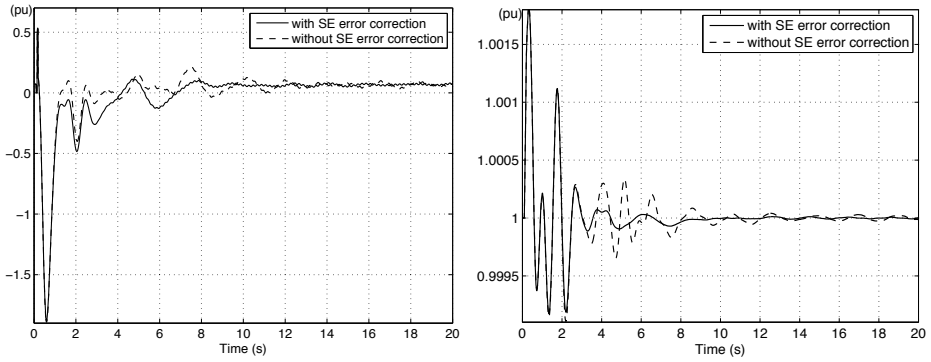

Figure 4.7: Response of centralized MPC with SE errors: P_{1-2} (left), and w_{g1} (right)

$$\begin{bmatrix} \hat{y}[k+1|k] \\ \hat{y}[k+2|k] \\ \vdots \\ \hat{y}[k+T_h|k] \end{bmatrix} = P_x \hat{x}[k|k] + P_u \begin{bmatrix} \hat{u}[k|k] \\ \hat{u}[k+1|k] \\ \vdots \\ \hat{u}[k+T_h-1|k] \end{bmatrix} + \begin{bmatrix} I \\ I \\ \vdots \\ I \end{bmatrix} \hat{d}[k|k] \tag{4.7}$$

where

$$\hat{d}[k|k] = y[k|k] - C\hat{x}[k|k-1] \tag{4.8}$$

It is thus assumed that $\hat{d}[k|k]$ is refreshed each time a set of new SE results is collected, and then remains unchanged over the entire prediction horizon used to compute the controls.

4.5.2 Simulation results

SE errors are simulated as an additive random noise. Specifically, uniformly distributed pseudorandom errors in the range from -10% to $+10\%$ are superimposed on the exact states. The idea is to test performances of the control in presence of relatively high errors (maximum error in a state variale is $\pm 3 * \sigma$ with $\sigma = 3.3\%$, and these measurement standard deviations of measurements are revealed to be realistic even in case of PMUs since their accuracy is impacted by existing measurement equipments such as measurement transformers [93]). The results are shown in Fig. 4.7 without (dashed curves) and with (solid curves) SE errors correction.

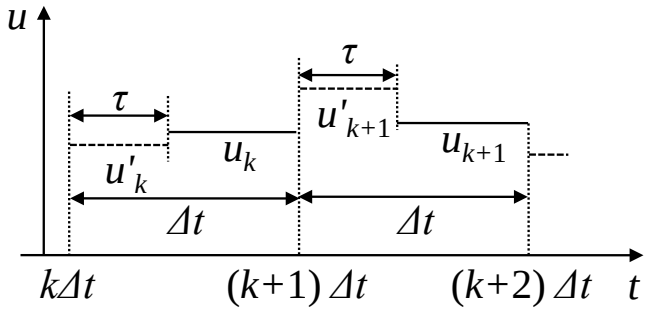

Figure 4.8: inputs u and delay τ

As expected, inaccuracies introduced by SE errors affect the performances of the MPC scheme in terms of magnitude of oscillations and settling time. On the other hand, it is observed that the correction of SE errors considerably improves the control performances (solid curves in Fig. 4.7).

4.6 Consideration of time delays

In all previous simulations, we have supposed that measurements, state-estimation results, and control computation and actuation could be achieved without any delay. In practical situations, communication systems and/or available computing resources will yield however non-negligible delays between the actual system time and the moment where controls can actually be applied. In the present section we therefore assess the effect of such delays in the context of our problem.

4.6.1 A strategy for handling of delays

Various delays, in measurements acquisition, computation and sending supplementary inputs to local controllers, could be involved in the implementation of the proposed MPC scheme [49]. It is therefore important to asses the impact of such delays on the performance of the proposed control scheme. There are two possibilities [49]: to apply control signals as soon as they are available or after a specific time interval. Here, we study the impact of delays in computation and sending inputs to local controllers by assuming that all measurements are taken synchronously, and then exploited with a common

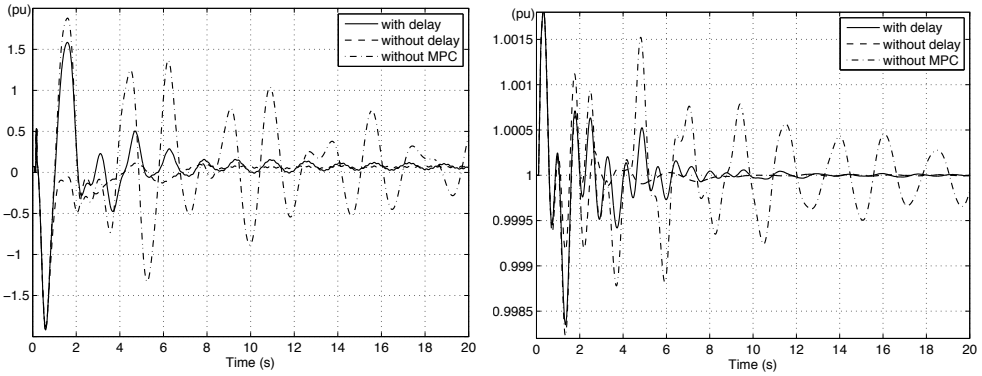

Figure 4.9: Response of centralized MPC with 0.05s delay: P_{1-2} (left), and w_{g1} (right)

delay. This is illustrated in Fig. 4.8 assuming a common delay of τ. We also assume that this delay is smaller than the refreshment period Δt of the MPC scheme.

To take the delays into account in a proper way, we proceed as follows. At every control refreshment instant $k\Delta t$, the MPC controller computes inputs for the next period until $(k+1)\Delta t$ and also for the subsequent period until $(k+2)\Delta t$. Because of the delay τ, inputs calculated at time $k\Delta t$ are only applied from $k\Delta t + \tau$. Thus, between $k\Delta t$ and $k\Delta t + \tau$, the input ${u'}_k$ calculated at the previous refreshment time is applied, while those computed from the measurements at time $k\Delta t$ are applied during the period $k\Delta t + \tau$ until $(k+1)\Delta t + \tau$.

4.6.2 Simulation results

A common delay of $\tau = 0.05s$ in measuring, computing and sending control signals to all available LCs is assumed. The choice of this delay is based on observations presented in [94]. System response under this assumption is shown in Fig. 4.9 (SE errors are modeled in the simulations and corrected in the used MPC schemes). It is clear that the performances taking into account delays are worse than those with only SE errors, but still quite superior to those without MPC. Fig. 4.10 gives MPC signals for PSS of generator 1, in the conditions without delay and with delay.

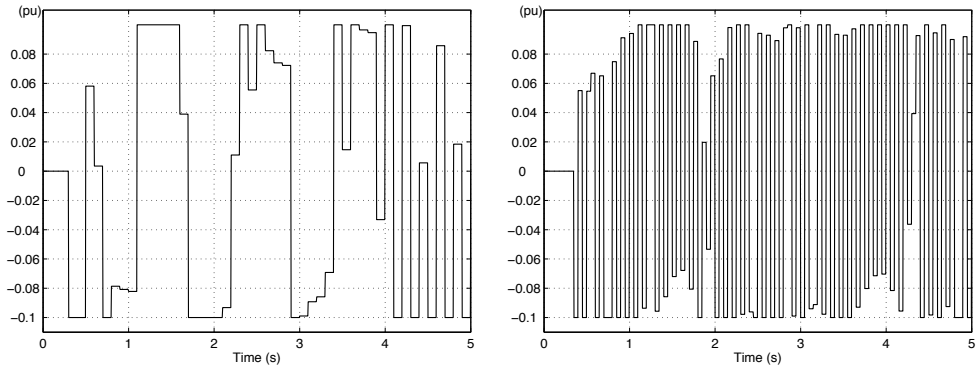

Figure 4.10: MPC signals for PSS of generator 1: without delay (left) and with delay (right)

4.7 Effect of unavailabilities of some local controllers

So far, we assume that all 16 PSSs and the TCSC are available to the MPC controller and are able to receive supplementary control inputs. In this subsection we consider the case where several PSSs and the TCSC are not available to the MPC controller.

In our simulations, we assume that the MPC controller is aware of the remaining available LCs and hence adjusts its strategy by computing supplementary signals only for these latter LCs (non-available LCs are hence removed from the optimal control problem formulation, but the effects of the local control loops are still modeled in the system dynamics).

In other words, we do not assess the robustness of MPC with respect to unnoticed controller failures but rather study the effect of decreased system controllability. We believe that this investigation well illustrates fault-tolerance capability of the proposed MPC scheme in the practical case of identified and isolated LC failure (generally, fault detection and isolation for this type of failure is not considered to be a difficult problem [49]).

The solid curves of Fig. 4.11 show the system response when six PSSs (PSS1-3, PSS10-12) and TCSC are not included in the MPC control scheme. Performances become slightly worse. When eight PSSs (PSS1-4, PSS10-13) and TCSC are not included (this represents 53% of all available damping controllers) performances further deteriorate, but they remain still acceptable compared to the uncontrolled case, as shown by the dashed curves of

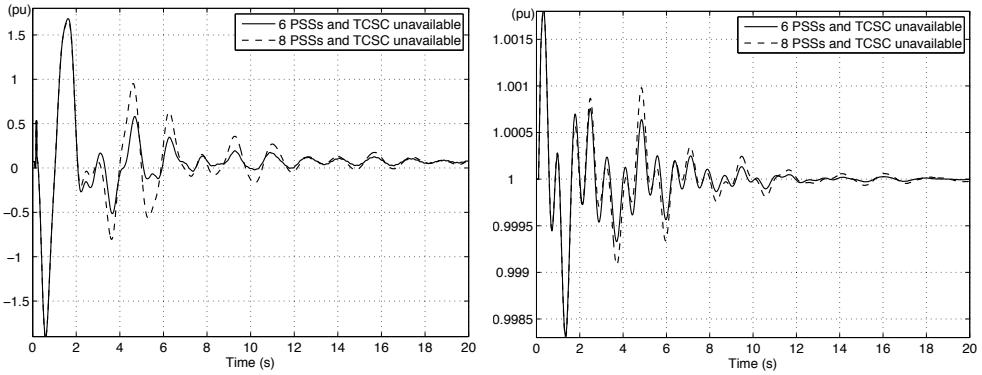

Figure 4.11: Reduced number of available local controllers: P_{1-2} (left), and w_{g1} (right)

Fig. 4.11.

In the literature, it has been noticed that MPC schemes have certain inherent fault-tolerance property [49]. We plan to investigate this property in the next work so as to further assess the robustness of the proposed MPC scheme for inter-area oscillation damping.

4.8 Incorporating additional control devices

One of the key advantages in using MPC to damp electromechanical oscillations is its flexibility to incorporate with any kind of control device. To illustrate this in the context of our test system, we consider in this subsection the possibility to use another type of local controller, namely an SVC. Thus we introduce a new type of LC (an SVC) in addition to the already considered ones. This SVC is installed in bus1.

Fig. 4.12 displays the results for this case (we assume availability of all PSSs and the TCSC), while Fig. 4.13 and 4.14 show supplementary signals for PSS1, TCSC and SVC over the first 5s. These results clearly show that the proposed MPC scheme is indeed able to effectively incorporate a new control device and successfully coordinate its control efforts with the other available controls regardless of their types.

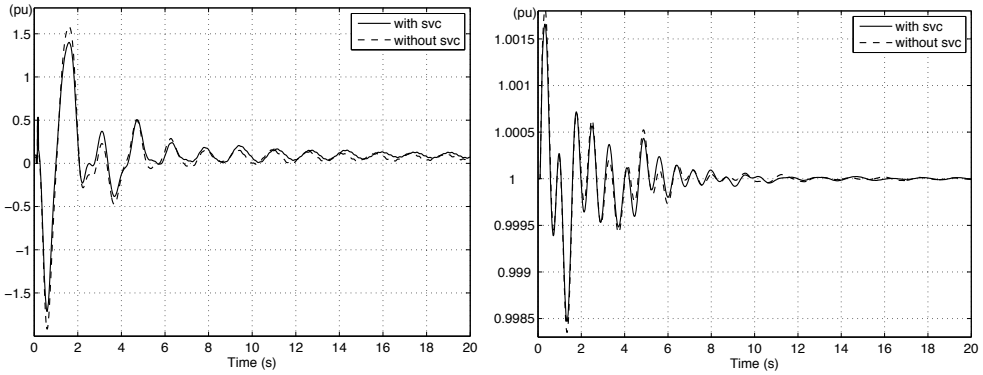

Figure 4.12: Response using PSS, TCSC and SVC: P_{1-2} (left), and w_{g1} (right)

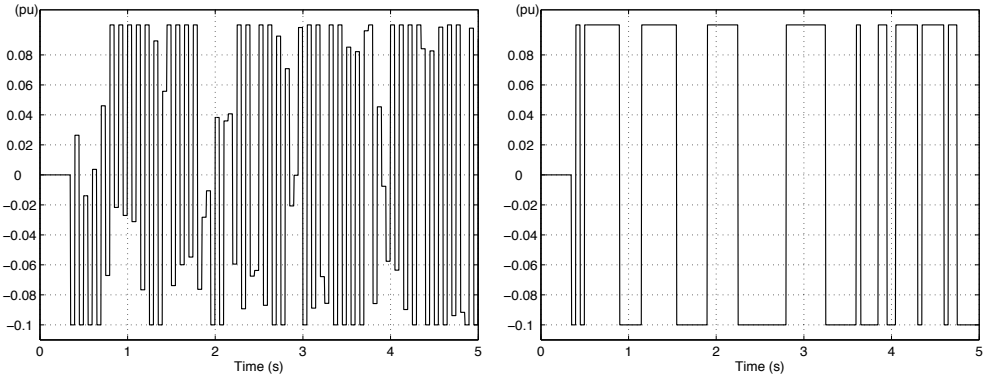

Figure 4.13: MPC control signals: PSS of generator 1 (left), and TCSC (right)

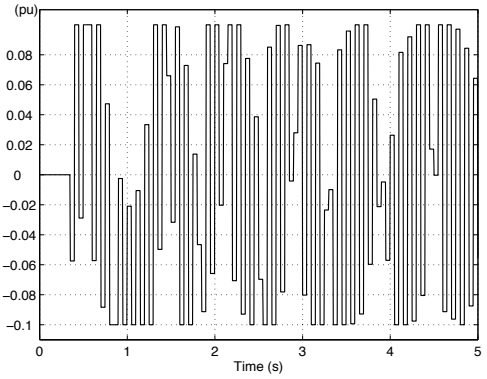

Figure 4.14: MPC control signals of SVC

4.9 Conclusions

In order to help in the design of wide-area control schemes for damping electromechanical inter-area oscillations in large-scale power systems, a new centralized MPC scheme is proposed and investigated in this chapter. This approach is justified by the fact that in modern power systems good dynamic models, accurate and fast measurement systems, and state-of-the-art optimal control formulations and scalable algorithms could be dealt with at the same time and in real-time with available computing platforms.

The proposed MPC scheme calculates all supplementary control signals in a centralized way. Encouraging results obtained using a medium sized power system show that this scheme can indeed effectively and quickly damp inter-area oscillations. Additional experiments are also provided, showing that such MPC schemes can tolerate realistic SE errors and communication and computation delays.

Furthermore, the versatility of the proposed MPC control approach is illustrated by showing how it can adapt effectively to handle a reduced number of control devices and how it can take advantage of additional control devices.

Overall, the simulation results provided in this chapter show that a proper MPC scheme may be quite effective in coordinating a broad diversity of control devices, such as PSSs, TCSCs, and SVCs, for the damping of inter-area electromechanical oscillations.

In the subsequent chapters, this centralized framework will be extended in order to comply with practical constraints of very large-scale interconnected power systems. Specifically, a multi-layer and distributed MPC scheme that is considered to be more appropriate for large-scale multi-area systems will be designed. In such a scheme fast updates of supplementary controls would be carried out in the lower layers based on detailed models of smaller sub-areas, while slower updates of supplementary controls would be used to coordinate them at the higher layers based on the aggregation of the models of the lower layers.

Chapter 5

Distributed MPC scheme

5.1 Introduction

In this chapter, a distributed MPC scheme is proposed. It consists of a set of area-wise MPC controllers that are implicitly coordinated via tie-line constraints. The scheme is implemented on our test system and compared with the fully centralized scheme of chapter 4.

The content of the present chapter is mainly based on our publication

[59] *D. Wang, M. Glavic, and L. Wehenkel. Distributed MPC of wide-area electromechanical oscillations of large-scale power systems. In Proceedings of the 16th Intelligent System Applications to Power Systems (ISAP), Crete, Greece, September 2011.*

5.2 Centralized vs distributed control

For large-scale systems, like the interconnected power systems with long transmission lines, water supply systems widely distributed in space and traffic systems with large-space flexible structures, it is often not feasible to solve their control problems in a centralized way due to the problems of dimensionality, uncertainty and information structure constraints. An alternative is to break down a complex control problem into manageable subproblems which are only weakly related to each other and can be solved independently. It has the following advantages. Firstly, it is computationally efficient to formulate

control laws that use only locally available subsystem states or outputs. Secondly, such an approach is also economical, since it is easy to implement, and can significantly reduce communication cost. Robustness is another attractive feature of decentralized/distributed control laws, since they can make the stability of the closed-loop system tolerant to a broad range of uncertainties, both within the subsystems and in their interconnections. Finally, when the exchange of state information among the subsystems is prohibited, decentralized/distributed structure becomes an essential design constraint [95].

As far as wide-area power system oscillations are concerned, they tend to appear in very large-scale systems, ranging over thousands of kilometers and involving many different subsystems managed by different transmission system operators (TSOs). So it is often practically not feasible to handle these problems with a fully centralized approach. On the other hand, reliability/vulnerability considerations may suggest that even in a system where a fully centralized control scheme would be feasible, it is not necessarily desirable to do so. Consequently, it is of interest to study distributed MPC schemes that could be more viable and easier to implement. In this setting, local MPC systems could determine optimal supplementary inputs for a subset of existing controllers under their authority, based on a model of their subsystem and a local control objective.

5.3 Related works

Various distributed MPC schemes have been proposed to replace the centralized MPC in large-scale power systems [76, 77, 96–99]. In paper [96], control agents are placed at each generator and load to control power injections to eliminate operating-constraint violations before the protection system acts. They use detailed models of their surrounding areas and simplified models of remote areas to predict system states. The agents cooperate with each other by sharing their objectives and exchanging solutions and measurements. By cloning the boundary nodes of neighboring areas, paper [98] breaks the whole power grid into relatively independent subsystems that only interact through consistency constraints on shared variables; each local MPC controller calculates optimized supplementary inputs for automatic voltage regulators and static var compensators in its area, and coordinates with neighboring MPC controllers by exchanging Lagrange multipliers.

Normally, the centralized MPC can achieve the best attainable perfor-

mance as the effects of interactions among subsystems are accounted for exactly and the conflicting control objectives are resolved optimally. For the distributed MPC, the proper cooperating algorithm must be designed to assure its overall performance. In the paper [76], the local MPC controllers use the latest available external state information from the previous iteration except their local subsystem variables, and then iteratively and cooperatively solve a system-wide control objective in order to achieve similar performance to the centralized MPC. Paper [77] proposes a distributed MPC scheme based on Nash equilibrium where local MPC controllers exchange their state trajectories, and iteratively solve their local optimization problems until no local controller can improve further its solution.

In this chapter, we extend the centralized MPC of last chapter towards a distributed MPC implementation, which would be a more viable solution for damping low-frequency oscillations in very large-scale interconnected power systems. As with any other distributed control scheme, two problems are to be solved, namely decomposition and coordination.

5.4 Control problem decomposition

A large-scale control problem can be decomposed into subproblems by the following two main approaches:

- Problem-driven: construction of a global system model followed by an optimal decomposition into subsystems according to structural properties of the system and the control problem under consideration [100],

- Context-driven: the decomposition of the whole system is imposed by contingent constraints, and hence the construction of the local control schemes has to follow the already given decomposition.

Considering information-exchange restrictions in certain power grids, and organizational barriers, it is quite difficult to construct an exact system-wide model, and even if this were possible, it would be difficult to impose significant changes within the existing control structures. Therefore it is preferred to consider in this chapter a context-driven decomposition of control areas [77].

Let us consider a two-area system, shown in Fig. 5.1, to illustrate our approach. Notice that this system is decomposed a priori into two control

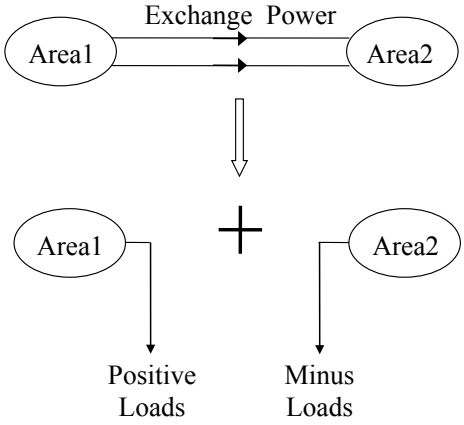

Figure 5.1: Decomposition of an interconnected grid into two subsystems

areas, linked by tie-lines allowing to exchange power. A control system only able to act on one of the two subsystems could view the rest of the system by an equivalent dynamic load in order to compute its control inputs.

Accordingly, the control objective of Eqn. (4.5) is rewritten as the simultaneous and parallel resolution of the following area-wise problems (subscript m refers to area m):

$$\min_{u_m[\cdot]} \sum_{i=0}^{T_h-1} (\hat{y}_m[k+i+1|k] - y_{ref.m}[k+i+1|k])^T W_{y_i.m} \\ (\hat{y}_m[k+i+1|k] - y_{ref.m}[k+i+1|k]) \tag{5.1}$$

subject to linear inequality constraints:

$$\begin{array}{l} u_{min.m} \leq \hat{u}_m[k+i|k] \leq u_{max.m}, i = 0, 1, 2, ..., T_h - 1. \\ z_{min.m} \leq \hat{z}_m[k+i+1|k] \leq z_{max.m}, i = 0, 1, 2, ..., T_h - 1. \end{array} . \tag{5.2}$$

In this distributed scheme, each local MPC controller solves its optimization problem using a detailed model of its own area and a possibly very rough model of the remaining areas (typically a black box model). It then sends the first inputs in the computed optimal control sequence to the controllers under its responsibility, and observes the resulting effects to proceed.

5.5 Coordination of controls of subproblems

Since distributed MPC controllers act in the same system, coordination is needed in order to achieve satisfactory performances in damping low-frequency oscillations throughout the system [76, 83, 96]. They could negotiate/exchange useful information in order to improve global performance. For example, local agents of [96] are designed to inform their neighbors about what they intend to do and pass along measurements that other agents may not be able to sense directly. In [83], local agents exchange Lagrange multipliers and values of shared variables after each computation and computations are carried on until absolute changes in Lagrange multipliers are smaller than a pre-defined small positive constant, before actual controls are applied to the system.

However, in very large-scale systems, there may be many remote areas, thousands of kilometers away from each other and separated by many intermediate areas. Communicating and negotiating between all these areas would require supporting communications infrastructure. In today's large interconnections the lack of communications infrastructure is the main obstacle for implementing advanced control schemes. Upgrade of this infrastructure is costly and will remain an issue at least in the near future. Consequently, instead of explicit communication/negotiation, an implicit coordination scheme is used in this context. Specifically, each subsystem tries to solve its own oscillation problem and the overall system stability emerges from these area-wise controls [76]. As far as the considered scheme is concerned, each area-wise MPC controller adjusts supplementary signals for damping controllers, under its authority, so as to minimize deviations of individual generators' angular speeds from their nominal values, while ensuring that all constraints are satisfied.

5.6 Simulation results

5.6.1 Test system and simulation parameters

In order to illustrate the proposed distributed MPC scheme, we use the same 16 generators / 2 areas / 70 buses system as in chapter 4, shown in Fig. 4.2. The TCSC is assigned to area A1 as its control resource. A same temporary three-phase short-circuit to ground at bus 1 is used to excite

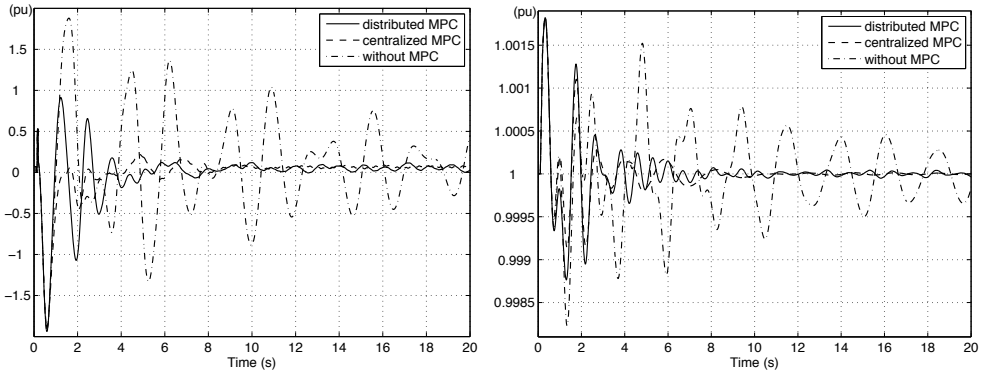

Figure 5.2: Response in ideal conditions: P_{1-2} (left), and w_{g1} (right)

electromechanical oscillations between area A1 and area A2.

One area-wise MPC controller is set in each one of the two areas. MPC1 is responsible for PSS10-16 and TCSC; MPC2 calculates supplementary signals for PSS1-9. The state vectors of the two MPC controllers contain generator, exciter, PSS and turbine governor states of the elements in their control areas (MPC1's state vector also contains 1 TCSC state). Both output vectors y_m compile the generator speeds for each area, whose references $y_{ref.m}$ are the unit vectors of dimension 7 and 9 respectively. Input vector u_1 of MPC1 consists of supplementary inputs for PSS10-16 and TCSC, while MPC2 inputs (u_2) are supplementary inputs for PSS1-9. All inputs are subject to $-0.1 \leq u_m \leq 0.1$.

We still use in our simulations a prediction horizon of 1.5s and a control horizon $T_c = 3$ (i.e. the MPC input is changed over the next 3 steps). To assess controlled system responses, we simulate and observe them over a period of 20s. In the objective function (5.1), all deviations of the predicted outputs from references are weighted uniformly and independently, i.e. $W_{y_i.m}$ is an identity matrix.

5.6.2 Results in ideal conditions

Controlled system responses over a time period of 20s of the power flow through tie-line 1-2 and the angular speed of a representative generator are shown by solid curves in Fig. 5.2. Fig. 5.3 gives centralized MPC signals and distributed MPC signals of PSS1. Compared with the system responses without MPC, namely dot-dashed curves in Fig. 5.2, we clearly observe that this

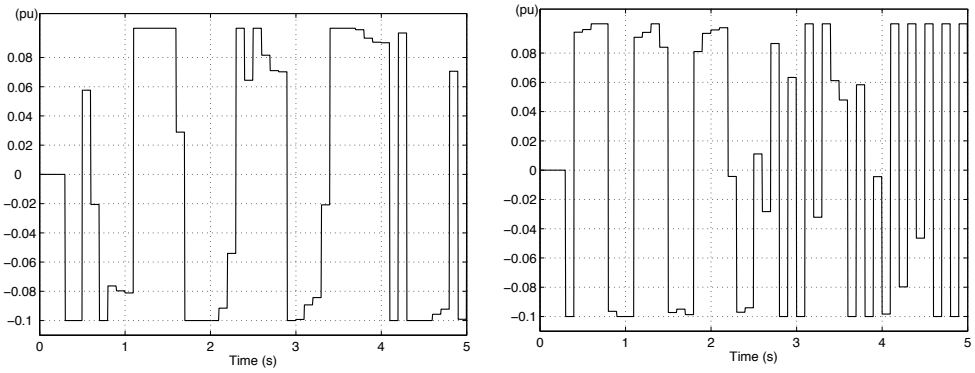

Figure 5.3: MPC signals for PSS of generator 1: centralized MPC (left), and distributed MPC (right)

(trivially) distributed MPC scheme effectively damps the power oscillations of tie-line 1-2 and the angular speed of generator 1.

Compared with the fully centralized MPC scheme (dashed curves of Fig. 5.2), control effects of this distributed MPC scheme are still quite inferior. This is not so astonishing, because the formulation of the optimal solutions of local MPC schemes does not provide any guarantee in terms of global optimal solution optimality (Paper [101] discusses conditions under which optimal solutions of subproblems compose an optimal solution of the overall problem. But they unfortunately do not apply to the present scheme.).

We also note that although this trivial distributed MPC scheme quickly damps large oscillations within the first 5s, small oscillations remain and last from 5s to 20s. These small oscillations are possibly caused by the inaccurate and incomplete models used by sub-area MPC controllers.

5.6.3 Results with SE errors

We still use uniformly distributed pseudorandom errors of $\pm 10\%$ superimposed on the exact states to represent SE errors in two control areas. We introduce the SE errors correction and obtain the results shown in Fig. 5.4. Compared with Fig. 5.2, we can see that the SE errors correction brings slightly improved results, because the output feedback not only corrects SE errors but also corrects to a certain extent model errors. Moreover, Fig. 5.4 also shows that the centralized MPC is still superior to the distributed one

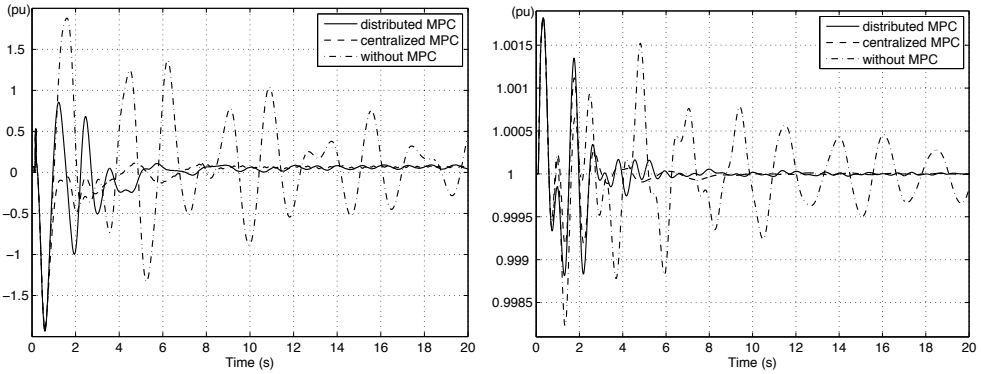

Figure 5.4: Response with SE errors: P_{1-2} (left), and w_{g1} (right)

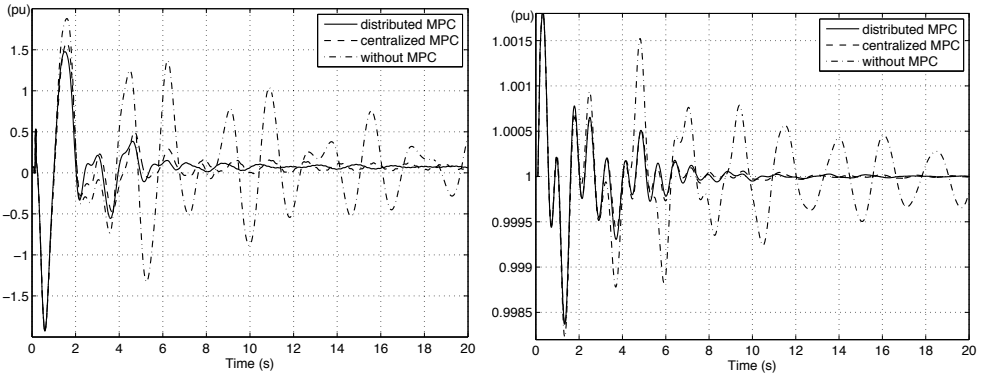

Figure 5.5: Response with 0.05s delay: P_{1-2} (left), and w_{g1} (right)

in the condition only considering SE errors.

5.6.4 Results considering delays

Finally, we explore system responses with a delay of 0.05s. We apply the same treatment to both sub-area MPC controllers, as described in Chapter 4, and obtain the results of Fig. 5.5. It is actually observed that the distributed scheme yields a slightly better control effect. This is due to the chosen decomposition method. The system is decoupled by substituting exchange powers for two equivalent loads which absolute values are equal to exchange powers at steady state. So, linearized models calculated in this condition contain the information about these powers. In addition to the explicit control objective of Eqn. 5.1, the area-wise MPCs thus have the im-

plicit objective to restore exchange powers to the steady state values, and this actually helps to drive the system to steady state.

5.7 Conclusions

In this chapter, a distributed MPC scheme is considered to damp wide-area electromechanical oscillations in large-scale electric power systems. Results obtained on a 16-generator, 70-bus, 2-area test system show that the distributed MPC scheme, even with no explicit coordination can effectively and quickly damp large inter-area oscillations.

Next, the distributed MPC scheme is compared with a fully centralized MPC based control proposed in last chapter in ideal conditions, considering SE errors and delays. In the ideal conditions and only considering SE errors, the centralized MPC is superior to the distributed one. But after introducing a delay of 0.05s, the distributed MPC scheme obtains a little better results due to its used decomposition method.

Further work should address ways to cope with overlapping models, measurements and control areas, and incorporate the practical constraints to further distributed control at the lower (intra-area) levels. Also, the investigation of stability and robustness of the proposed distributed MPC scheme is an important subject of further research.

While we do not believe at this stage that computational considerations are constraining in this problem, we believe that correct model construction is one aspect to keep in mind given the rapidly changing conditions and the need to better take into account the behavior of local subsystems (specially as concerns the distribution levels). Also, limitations in information exchanges due to lack of agreements or of technical communication channels may hinder progress. In these respects, automatic learning techniques could be useful to complement the model predictive control approach.

Eventually, one will be able to confront various distributed control approaches with various local objective functions, by leveraging modern information processing techniques.

Chapter 6

Hierarchical MPC scheme

6.1 Introduction

As the mixture of centralized control and distributed control, hierarchical control has already been used in the designs for voltage instability control [102], load frequency control [103, 104], oscillation damping control [39, 105, 106], transient stability control [107] and blackout restoration control [108].

An advantage of a hierarchical control structure is that it can coordinate local controllers to improve global control effects. In [39], a wide-area central controller is responsible to decouple subsystems dynamics and calculate the interactions among them for lower-level local controllers. Reference [105] proposes a two-loop hierarchical structure: a local loop based on a machine speed signal and a global loop based on a differential frequency between two remote areas. The total PSS signal applied to the machine voltage reference is the sum of the control components of these two loops.

In addition, a hierarchical structure also facilitates the implementation of particular algorithms. For example, in a conventional fuzzy logic system, the number of rules grows exponentially with the number of variables. So, paper [106] decomposes the fuzzy system into a set of simpler fuzzy subsystems connected in a hierarchical manner. In the new hierarchical structure, the total number of involved rules increases only linearly.

In this chapter, we propose a hierarchical MPC scheme based on the previous distributed one, and aiming at improved control robustness, e.g. in the case of failure of one of the area-wise controllers. Its superior performances are showed through comparison with the two MPC based schemes and by

considering the centralized MPC scheme of chapter 4 as the benchmark.

The content of the present chapter is mainly based on our paper

[60] *D. Wang, M. Glavic, and L. Wehenkel. Considerations of model predictive control for electromechanical oscillations damping in large-scale power systems. International Journal of Electric Power and Energy Systems, Accepted for publication, 2014.*

6.2 Outline of hierarchical MPC

This hierarchical MPC control is illustrated in Fig. 6.1. In addition to MPC controllers operating at the level of whole areas (subsystems), a lower level layer of independent MPC controllers attached to N_n basic control devices is added. Compared with MPC controllers in the upper level, MPC controllers in the lower level only concern dynamic behaviors of one generator or tie-line. These controllers need less time to measure, compute and apply their decisions, so that they can update their control decisions more frequently following changes of system states, and thus approach their control targets in a possibly better way. In addition, when upper MPC controllers cannot work normally, lower MPC controllers are designed to work independently with the aid of internal control objectives.

6.3 MPC in the lower level

We install lower MPC controllers respectively on generators and on a TCSC in tie-line 8-9, in the test system used throughout this thesis, whose formulations are given as follows.

6.3.1 MPC on a generator

An exciter, a PSS and a turbine governor are assumed on each generator, as shown in Fig. 6.2. A generator model used is the one including the effects of sub-transient circuits. These devices are modeled by the following equations:

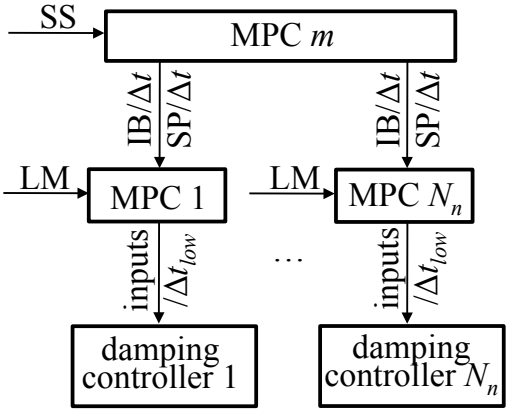

Figure 6.1: Hierarchical MPC scheme of subsystem m

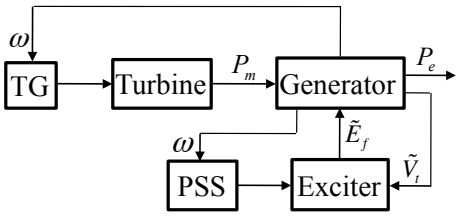

Figure 6.2: Controllers on one generator

$$
\begin{aligned}
&p\delta = \tilde{K}_1 \Delta\omega \\
&p\omega = \tilde{K}_2 T_m - \tilde{K}_3 T_e - \tilde{K}_4 \Delta\omega \\
&p\tilde{E}'_q = \tilde{K}_5 E_f - \tilde{K}_6 \tilde{E}'_q - \tilde{K}_7 \tilde{I}_d + \tilde{K}_8 \tilde{\psi}_{1d} \\
&\tilde{\psi}''_d = \tilde{K}_9 \tilde{E}'_q + \tilde{K}_{10} \tilde{\psi}_{1d} \\
&p\tilde{E}'_d = -\tilde{K}_{11} \tilde{E}'_d - \tilde{K}_{12} \tilde{I}_q + \tilde{K}_{13} \tilde{\psi}_{1q} \\
&\tilde{\psi}''_q = \tilde{K}_{14} \tilde{E}'_d + \tilde{K}_{15} \tilde{\psi}_{1q} \\
&pE_f = -\tilde{K}_{16} E_f + \tilde{K}_{17} V_r \\
&pV_r = -\tilde{K}_{18} V_r - \tilde{K}_{19} pR_f + \tilde{K}_{20}(V_{ref} - V_t - V_{pss}) \\
&pR_f = -\tilde{K}_{21} R_f + \tilde{K}_{22} E_f \\
&V_{pss} = \hat{G}(pss)\omega; P_m = \hat{G}(tg)\omega,
\end{aligned}
\tag{6.1}
$$

where p is the differential operator; $\tilde{E}'_q$ and $\tilde{E}'_d$ are q axis and d axis component of transient stator voltage; $\tilde{\psi}''_q$ and $\tilde{\psi}''_d$ are q axis and d axis air gap flux linkage; $\tilde{\psi}_{1q}$ and $\tilde{\psi}_{1d}$ are amortisseur circuit flux linkages of q axis and d axis; E_f is field voltage; V_r and R_f are exciter states; V_{ref} is voltage reference of exciter; V_t is terminal voltage of generator; V_{pss} is PSS output; P_m is mechanical power; $\hat{G}(pss)$ and $\hat{G}(tg)$ are transfer functions of PSS and turbine governor; $\tilde{K}_1, \tilde{K}_2, \ldots, \tilde{K}_{22}$ are coefficients. Detailed models are given in Appendix C. Using these models, a lower MPC controller calculates a supplementary signal for its PSS to reach the objective of making the corresponding generator run at base frequency, which is defined as (subscript n refers to generator n):

$$
\min_{u_n[\cdot]} \sum_{i=0}^{T_h-1} (\hat{y}_n[k+i+1|k] - y_{ref.n}[k+i+1|k])^2 \tag{6.2}
$$

subject to:

$$
\begin{aligned}
&u_{min.n} \le \hat{u}_n[k+i|k] \le u_{max.n}, i = 0, 1, 2, ..., T_h - 1. \\
&z_{min.n} \le \hat{z}_n[k+i+1|k] \le z_{max.n}, i = 0, 1, 2, ..., T_h - 1.
\end{aligned}
\tag{6.3}
$$

6.3.2 MPC on a TCSC

An MPC controller installed on a TCSC calculates, at every refreshing time, the active power increment of its tie-line and assumes that there will be a

negative increment with same size at the next refreshing time. So, it can calculate line reactances from:

$$P = \frac{|\tilde{V}_1|^2\hat{R} - |\tilde{V}_1||\tilde{V}_2|\hat{R}cos\theta + |\tilde{V}_1||\tilde{V}_2|\hat{X}sin\theta}{\hat{R}^2 + \hat{X}^2} \tag{6.4}$$

at two refreshing times, and then gets the corresponding MPC inputs for the TCSC. In (6.4) P is active power; $|\tilde{V}_1|$ and $|\tilde{V}_2|$ are voltage magnitudes at both ends; θ is angle difference of voltages at both ends; $\hat{R}$ and $\hat{X}$ is line resistance and reactance.

6.4 Coupling between the two layers of MPC

Two ways of coupling are investigated (see Fig. 6.1).

- **Input base coupling**: every Δt seconds, the area-wise MPC controller collects subsystem states and calculates optimal inputs for controllers under its authority. It sends the inputs to device-level MPC controllers as their decision bases. Every Δt_{low} seconds, each device-level MPC controller computes an input correction according to its local measurements and control objective, and combines the correction with input base as the supplementary control input of its damping device.
- **Set-points coupling**: the upper MPC controller solves an optimization problem and sends predicted system dynamics, angular speeds and line powers over a future time horizon, to lower MPC controllers as their set-points. The lower MPC controllers calculate the optimal supplementary inputs for damping controllers in order to drive the controlled outputs to reach set-points given by the upper MPC controller.

6.5 Coordination between lower MPC controllers

The controlled output of a lower MPC controller depends not only on its states and input, but also on states and controls in the rest of a system. For a lower MPC controller, system variables and controls can be divided into two categories: internal and external. During a refreshing interval of upper MPC, it considers external variables and controls as constant, and uses them as

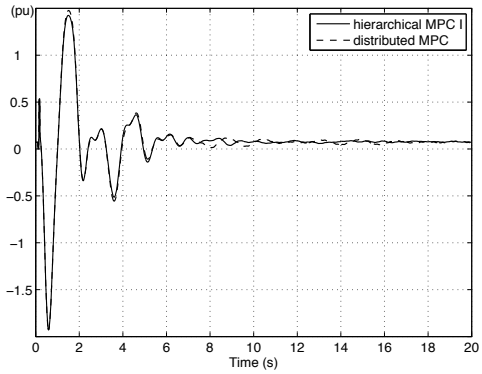

Figure 6.3: P_{1-2} with hierarchical and distributed MPC

simulation scenarios reflecting the anticipated influence of external variables and controls. The model of a lower-level MPC controller n of subsystem m is thus represented as follows:

$$\begin{aligned}\hat{x}_n[k+1|k] &= A_n\hat{x}[k|k] + B_n\hat{u}[k|k] \\ &= [A_{n,n}, A_{n,\text{ext}}]\begin{bmatrix}\hat{x}_n[k|k] \\ \hat{x}_{\text{ext}_n}[k|k]\end{bmatrix} + [B_{n,n}, B_{n,\text{ext}}]\begin{bmatrix}\hat{u}_n[k|k] \\ \hat{u}_{\text{ext}_n}[k|k]\end{bmatrix} \\ &= A_{n,n}\hat{x}_n[k|k] + B_{n,n}\hat{u}_n[k|k] \\ &\quad + A_{m,\text{ext}}x_{\text{ext}_m}[k|k] + B_{m,\text{ext}}u_{\text{ext}_m}[k|k]\end{aligned} \tag{6.5}$$

where x_n is the vector of internal state variables; u_n is the vector of internal controls; A_n and B_n are the parts of A and B that are relative to x_n; x_{ext_n} is the vector of state variables external to controller n, and u_{ext_n} is the external control vector. The external variables and controls for the local MPC are considered constant during one refreshment period of the upper MPC, and they are hence fixed at the values computed at the previous step by the upper MPC.

6.6 Simulation results

We test the hierarchical MPC scheme still using the same 16-generator, 70-bus system of Fig. 4.2 and the same temporary three-phase short-circuit to ground at bus 1.

MPC controllers are installed on each PSS and the TCSC. The upper MPC controllers use the same simulation parameters as distributed MPC.

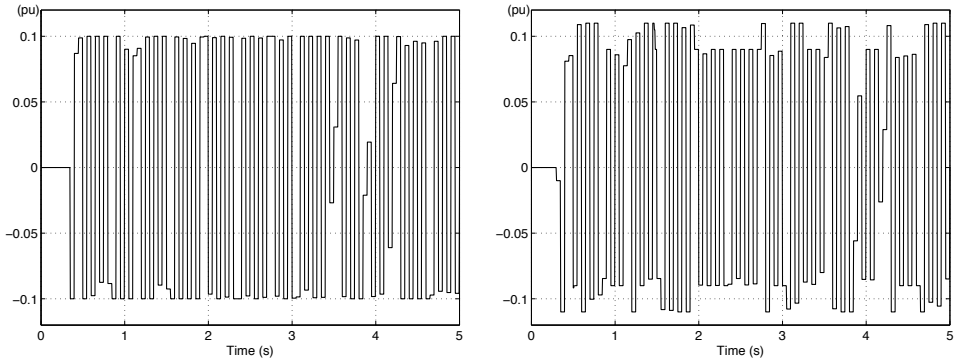

Figure 6.4: MPC signals of PSS1: distributed structure (left), and hierarchical structure (right)

The lower MPC controllers refresh decisions at discrete step of $\Delta t_{low} = 0.01$s and use a control horizon $T_{c.low} = 5$ such steps. The prediction horizons $T_{h.low}$ are set to 0.6 seconds (A1) and 0.4 seconds (A2). Fig. 6.3 shows controlled system response with the same SE errors and delays as before. Compared with the solely distributed MPC scheme, the hierarchical one further slightly improves control effects. The difference of MPC signals between distributed structure and hierarchical structure is shown in Fig. 6.4.

6.7 Advantages of hierarchical MPC

Next, the hierarchical MPC is compared to the distributed scheme in terms of robustness, with respect to the following five aspects:

1. Increasing the refreshing step Δt of area-wise controllers to 0.2s.

2. Simulating the complete failure of the MPC of A1 during the first 5s.

3. Incomplete measurements by assuming that only states of generators 10-13 in area A1 and generators 1-5 in area A2 are available for upper MPC controllers.

4. Communication failure between upper MPC controllers and local damping devices or MPC controllers. It is assumed that PSSs or lower MPC controllers on generators 1-4 and 10-13 can not receive the input signals from the upper MPC controllers.

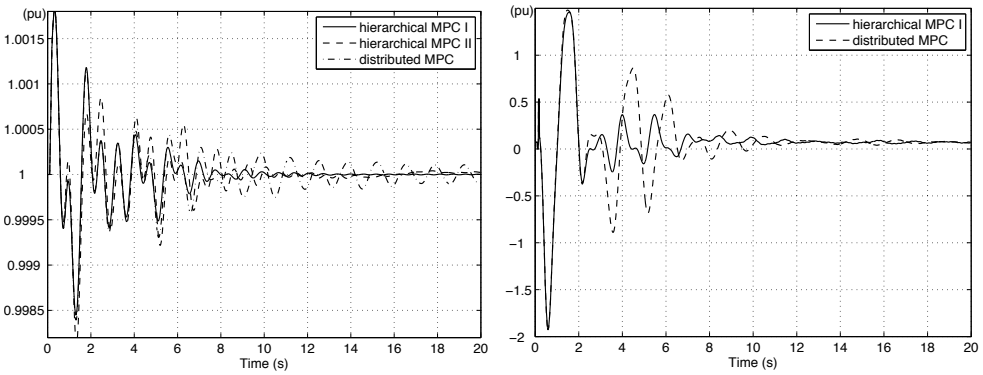

Figure 6.5: w_{g1} using a refreshing interval of 0.2s (left), and P_{1-2} with an upper MPC1 failure (right).

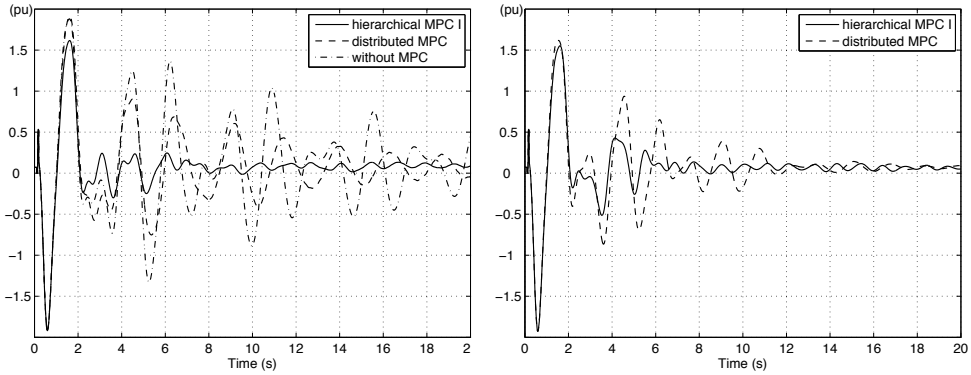

Figure 6.6: P_{1-2} with incomplete measurement (left), and communication failure (right).

5. Different operation conditions through comparison of distributed MPC and hierarchical MPC with topology change (tripping tie-line 1-2) and flow change (change in flows over two tie-lines from 0.07 pu to 0.4 pu in the line 1-2 and from 0.38 pu to 0.14 pu in the line 8-9).

The results are shown in Figs. 6.5, 6.6, and 6.7. They show that hierarchical MPC offers excellent robustness since it accommodates larger refreshing intervals, tolerates controller and communication failures, and works with incomplete measurement and in different operating conditions.

Fig. 6.5 compares two proposed couplings of the hierarchical MPC. Hierarchical MPC I means input base coupling while hierarchical MPC II means

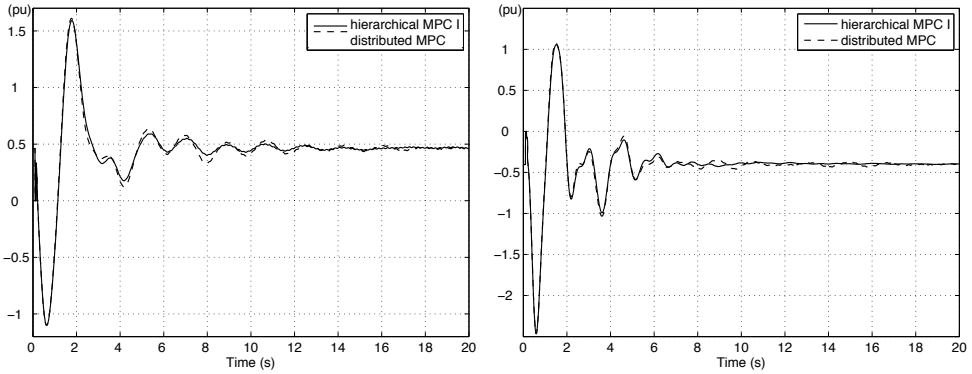

Figure 6.7: P_{1-2} with topology (left) and flow changes (right).

set-points coupling. In the hierarchical MPC II, large refreshing interval makes lower MPC controllers only use predicted values with larger errors as their set-points because prediction precision becomes worse in time. In the hierarchical MPC I, lower MPC controllers use rated angular speeds as their control objectives and continuously adjust their input corrections following system dynamics, thus yielding a slightly better performance.

6.8 Computational considerations

The computational efficiency of the hierarchical MPC is checked on a i7-3610 processor and 8G RAM. The MPC optimizations of (4.5), (5.1) and (6.2) are solved by an Active Set method built in the *Quadprog* function of MATLAB. It takes from 10 to 20 ms for the upper MPC controller, and from 3 to 5 ms for the lower MPC controllers, to complete one optimization.

6.9 Conclusions

In this chapter, a two-level hierarchical multi-area MPC scheme is introduced with the aim to enhance control robustness. The upper MPC controllers optimize inputs of damping controllers in their own areas and send them to the lower MPC controllers as their input bases. Using local models and measurements, the lower MPC controllers correct these input bases.

The performances of the proposed control scheme are tested using a 70-bus test system. Our simulation results show that the hierarchical MPC

further improves control effects of the distributed one, and offers at the same time much better robustness.

While MPC, being a closed-loop control scheme, has some intrinsic level of robustness to modeling errors, it nevertheless relies on the use of a correct dynamic model of the system. Within the context of power system oscillation damping, load-dynamics, and dynamics of renewable and dispersed generation may have a significant impact on the system behavior; since the composition of the load and dispersed generation may change significantly from one period of time to another (e.g. intra-daily, and seasonal effects driven by weather conditions) the system dynamics at a particular moment may not be well enough approximated by the model computed from the available data in TSO control centers to yield satisfactory performances of any one of the proposed MPC schemes. Further work therefore should be devoted to real-time model identification and to methods able to carry out model-free learning of supplementary controls.

Part III

Reinforcement Learning Based Supplementary Damping Control

Chapter 7

Reinforcement learning background

In order to make the thesis more complete, the involved background knowledge about RL is given in this chapter, which is adapted from [2, 50, 109].

7.1 Learning definition

In order to understand the concept of learning, the notion of intelligent agent [109] is introduced first. An intelligent agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators in order to reach a certain goal. This simple idea is illustrated in Fig. 7.1. The human is a good example of intelligent agent. A human agent has eyes, ears and other organs for sensors and hands, legs, mouth and other body parts for actuators.

The agent's perceptual inputs at any given instant are termed percepts. An agent's percept sequence is the complete history of everything the agent has ever perceived. In general, the agent's choice of action at any given instant can depend on the entire percept sequence observed to date. The agent's function is to map from any given percept sequence to an action. Learning is to build better mappings that can select right actions using existing percepts, and can perfect gradually prior percepts during the course of learning, in order to improve the agent's ability to act in the future.

A right action is the one that will cause the agent to be most successful. An agent generates a sequence of actions according to the percepts it receives,

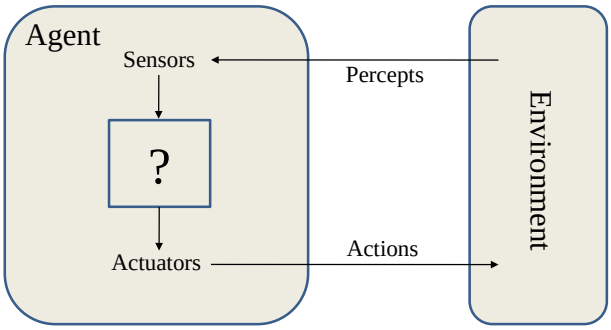

Figure 7.1: An intelligent agent. Taken from [109]

which causes the environment to go through a sequence of states. If the sequence of states is desirable, the agent is considered to be successful. The success is evaluated by a particular performance index, which is specified by the designer who constructs the agent.

7.2 Different learning problems

Learning problems can be classified into three categories: supervised learning, unsupervised learning and reinforcement learning.

7.2.1 Supervised learning (adapted from [2])

The problem of supervised learning involves learning an input-output function from collected samples of inputs and outputs.

Let us consider two random variables, $x \in X$ and $y \in Y$ drawn from some probability distribution $\hat{P}_{X,Y}$ defined over $X \times Y$, a loss function $\hat{\ell}$ defined over $Y \times Y$ and a hypothesis space $\hat{H} \subset Y^X$ of input-output functions. The average loss of a model $f \in Y^X$ is defined by

$$L(f) = \int_{X \times Y} \hat{\ell}(y, f(x)) d\hat{P}_{X,Y}. \tag{7.1}$$

Given a sample $\mathcal{TS} \in (X \times Y)^{\hat{M}}$ of input-output pairs, a supervised learning algorithm $\hat{A}$ aims at building a model $\hat{A}(\mathcal{TS}) \in \hat{H}$ to compute approximations of outputs as a function of inputs. The expected average loss

of the algorithm $\hat{A}$, for samples $\mathcal{TS}$ of size $\hat{M}$ drawn from some sampling distribution $\hat{P}_{\mathcal{TS}}$ defined over $(X \times Y)^{\hat{M}}$ is calculated by:

$$\overline{L}(\hat{A}) = \int_{(X\times Y)^{\hat{M}}} L(A(\mathcal{TS}))d\hat{P}_{\mathcal{TS}}. \tag{7.2}$$

The lowest reachable average loss in $\hat{H}$ is

$$L^*_{\hat{H}} = \inf_{f\in\hat{H}} \int_{X\times Y} \hat{\ell}(y, f(x))d\hat{P}_{X,Y}, \tag{7.3}$$

and the lowest possible average loss is calculated by

$$L^* = \inf_{\hat{H}\subset Y^X} L^*_{\hat{H}}. \tag{7.4}$$

Besides defining general conditions (on $X, Y, \hat{P}_{X,Y}, \hat{P}_{\mathcal{TS}}, \hat{\ell}, \hat{H}, \hat{A}$ etc.) under which the above introduced quantities indeed exist, the objective of statistical learning theory is essentially to study whether or in what sense $\overline{L}(\hat{A})$ converges to $L^*_{\hat{H}}$ when $\hat{M} \to \infty$.

On the other hand, the design of supervised learning algorithms essentially aims at constructing sequences of hypothesis spaces $\hat{H}_n$ and learning algorithms $\hat{A}_n$ with good convergence properties and such that $L^*_{\hat{H}_n} \to L^*$. In particular, much of the research in supervised learning has focused on the design of algorithms which scale well in terms of computational requirements with the sample size and with the dimensionality of the input and output spaces X and Y, and which use "large" hypothesis spaces able to model complex non-linear input-output relations.

7.2.2 Unsupervised learning (adapted from [2])

Given a sample of observations $\{y_{\hat{m}}\}_{\hat{m}=1}^{\hat{M}}$ independently and identically distributed according to a certain distribution $\hat{P}_Y$ over a space Y, the objective of unsupervised learning is essentially to determine an approximation of the sampling distribution. In the most interesting case, Y is a product space $Y_1 \times ... \times Y_{d_y}$ defined by d_y discrete or continuous random variables, and the main objective of unsupervised learning is to identify the relations among these latter (independence relations, colinearity relations) as well as the parameters of their distributions.

Earlier work in this field concerned clustering, principal component analysis and hidden Markov models. More recent research topics, still very active today, concern independent component analysis as well as the very rich field of graphical probabilistic models, such as Bayesian belief networks.

Independent component analysis aims at explaining the observed variables y_{l_y} as linear combinations

$$y_{l_y} = \sum \hat{\beta}_{l_y,l_x} x_{l_x}, \tag{7.5}$$

where the x_{l_x} are independent source variables.

Bayesian networks model the joint distribution of the random variables as a product of conditional distributions:

$$P(y_1, ...y_{d_y}) = \prod_{l_y=1}^{d_y} P(y_{l_y}|Pa(y_{l_y})), \tag{7.6}$$

where $Pa(y_{l_y})$ denotes for each variable a subset of so-called parent variables. The parent-child relation is encoded in the form of a directed acyclic graph, which explicitly identifies conditional independence relationships among subsets of variables. Unsupervised learning of Bayesian networks aims at identifying from a sample of observations the structure of the parent-child relationship and for each variable the parameters defining the conditional probability distribution $P(y_{l_y}|Pa(y_{l_y}))$.

7.2.3 Reinforcement learning

RL must learn itself from "the reinforcement signal (the observed return over a certain time horizon)" how to select control actions. The most important feature distinguishing RL from supervised learning is that training information from input-action samples only evaluates the actions taken rather than instruct the agent by giving correct actions. That is to say, training information only indicates how good the action taken is, but not whether it is the possible best or the possible worst action [50].

We redraw Fig. 7.1 in the form of Fig. 7.2 to illustrate the underlying concept of RL. The agent and environment interact at each of a sequence of discrete time steps, $t = 0, 1, 2, 3....$ At time t, the agent collects the environment's state $s_t \in S$, where S is the set of all possible states, and on that basis selects an action $u_t \in U(s_t)$, where $U(s_t)$ is the set of actions

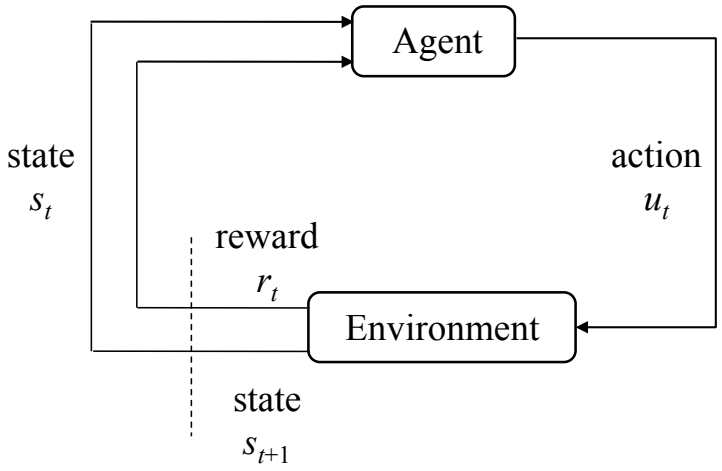

Figure 7.2: Interact between agent and environment. Taken from [50].

available at state s_t. At next time $t+1$, the agent receives a new environment state s_{t+1}, and then observes or calculates a numerical reward $r_t \in \mathbb{R}$.

In the standard RL setting, s_{t+1} is assumed to depend only on (s_t, u_t), not to be related to past states and actions. This is defined as the Markov property. At subsequent control times, new actions are selected and corresponding rewards are obtained. This setting corresponds to the so-called Markov Decision Process (MDP).

RL aims at deriving an optimal control strategy which chooses the sequence of actions based on the observations available at time t, so as to maximize the cumulated reward over a future temporal horizon. The cumulated reward, called the return, is defined (in the case of an infinite horizon) by

$$R_t^{\infty} = \sum_{i=0}^{\infty} \gamma^i r_{t+i}, \tag{7.7}$$

where γ is a discount factor. It determines the present values of future rewards: a reward received at time $t+i$ in the future is worth only γ^i times what it would be worth if it were received immediately. In order to get a finite return over an infinite horizon, it is assumed that $0 \leq \gamma < 1$ and the reward is bounded. If $\gamma = 0$, the agent is "myopic" in being concerned only by maximizing immediate reward r_t. As γ approaches 1, the agent takes future rewards into account more strongly: it becomes more farsighted.

Considering that one part of this thesis is to apply RL to design a model-free supplementary control for power system electromechanical oscillations' damping, we will provide some general background about RL in the next sections.

7.3 RL policies, rewards and returns

A reinforcement learning problem involves three main elements: policies, rewards and returns. We introduce this notions below, according to the terminology and derivations of [50].

7.3.1 Policies

A policy defines the agent's way of behaving at a future time horizon. Roughly speaking, a policy is a mapping from perceived states of the environment to actions to be taken at those states, namely $\pi : s_t \rightarrow u_t$. If a sequence of actions is selected by a policy from a state s_t, a corresponding return will be obtained. The good policy always selects the actions with large control returns. In some cases the policy may be a simple function or lookup table, whereas in others it may involve extensive computation such as a search process. The policy is the core of a RL agent in the sense that it alone is sufficient to determine behaviors of the agent.

7.3.2 Rewards and returns

A reward function maps each state-action pair (x_t, u_t) perceived by the environment to a single number $r_t \in \mathbb{R}$, indicating the intrinsic desirability of that state-action pair. The reward function defines what are good and bad events in an immediate sense for the agent. If an action selected by the policy is followed by a low reward, then the policy may be changed to select some other action in that situation in the future.

Whereas a reward function only indicates what is good in an immediate sense, a return function specifies what is good in the long run. Roughly speaking, the return of a sequence of state-action pairs is the total amount of rewards which an agent accumulates over this sequence. A RL agent's sole objective is to maximize the return that it receives in the long run.

7.3.3 Discussion about rewards and returns

Rewards determine the immediate, intrinsic desirability of state-action pairs, while returns indicate the long-term desirability of state-action pairs after taking into account the states that are likely to follow given the environment dynamics, and the actions available in those states. A state-action pair yielding a low immediate reward still can have a high return only if it is followed by other state-action pairs that yield high rewards. Or the reverse could be true.

Rewards are in a primary sense, whereas returns, as the discounted sum of rewards, are secondary. Without rewards there could be no returns, and the only purpose of estimating returns is to achieve more rewards. Nevertheless, it is the return with which we are most concerned when making action choices. Action choices are made based on the return judgment. We seek the actions that bring about the highest return, not the highest reward, because these actions obtain the greatest amount of rewards for us over the long run.

For example, a chess playing agent should be rewarded only for actually winning, not for achieving subgoals such as taking its opponent's pieces or gaining control of the center of the board. If solely achieving these sorts of subgoals was taken into account, then the agent might find a way to achieve them without achieving the real goal. It might find a way to take the opponent's pieces even at the cost of losing the game.

Unfortunately, it is much harder to determine returns than to determine rewards. Rewards are basically given directly by the environment, but returns must be estimated and re-estimated from the sequence of observations which an agent makes over a given horizon. In fact, the most important component of almost all RL algorithms is a method for efficiently estimating the returns. Next, we will introduce three methods of estimating these returns [50].

7.4 Reinforcement learning solutions

7.4.1 Return function formulation

The return function defines how good it is to perform a given action in a given state in the long run. For an agent, starting from the state s_t, taking the action u_t, and thereafter following the policy π, the expected return is

calculated by

$$Q^{\pi}(s_t, u_t) = \hat{E}_{\pi}\{R_t|(s_t, u_t)\} = \hat{E}_{\pi}\{\sum_{i=0}^{\infty} \gamma^i r_{t+i}|(s_t, u_t)\}. \tag{7.8}$$

The expected return here means that the $Q^{\pi}(s_t, u_t)$ is the average of all possible $R_t|(s_t, u_t)$ values, which correspond to distribution of the different possible subsequent trajectories influenced by any possible sources of stochastic behavior, but assuming that both the initial action u_t and the subsequent behavior π of the agent are fixed.

A fundamental property of $Q^{\pi}(s_t, u_t)$ used throughout RL is that it satisfies a recursive relationship. For any policy π and any state-action pair (s_t, u_t), the following consistency condition holds between $Q^{\pi}(s_t, u_t)$ and Q^{π} of possible following state-action pair:

$$\begin{aligned} Q^{\pi}(s_t, u_t) &= \hat{E}_{\pi}\{\sum_{i=0}^{\infty} \gamma^i r_{t+i}|(s_t, u_t)\} \\ &= \hat{E}_{\pi}\{r_t + \gamma \sum_{i=0}^{\infty} \gamma^i r_{t+1+i}|(s_t, u_t)\} \\ &= \hat{E}_{\pi}\{r_t + \gamma Q^{\pi}(s_{t+1}, \pi(s_{t+1}))|(s_t, u_t)\} \end{aligned} \tag{7.9}$$

where, s_{t+1} and $\pi(s_{t+1})$ are the next reached state and its corresponding action chosen by the policy π. Eqn. (7.9) is called Bellman equation which expresses a relationship between the Q value of a state-action pair and the Q value of its successor. The Bellman equation forms the basis of a number of algorithms to compute, approximate and learn the Q function.

Among all policies π, there is always at least one policy π^* that is better than or equal to all other policies in terms of expected return. $Q(\cdot, \cdot) = Q^{\pi^*}(\cdot, \cdot)$ is termed the optimal return function, and its corresponding policy is termed the optimal policy π^*. Solving a RL problem means finding the $Q(\cdot, \cdot)$ and π^*. This optimization problem can be represented as:

$$\begin{aligned} Q(s_t, u_t) &= \hat{E}\{r_t + \gamma \max_{u_{t+1}} Q(s_{t+1}, u_{t+1})\} \\ \pi^*(s) &= \arg\max_{u} Q(s, u). \end{aligned} \tag{7.10}$$

Normally, it is very difficult to calculate the optimal Q function and its corresponding policies because of computational cost and limits of environment's model. So, three categories of reinforcement learning algorithms are designed to build up the approximation of the optimal Q function and policies.

7.4.2 Dynamic programming

The term Dynamic Programming (DP) refers to a collection of algorithms that can be used to compute the optimal policies given a perfect model of the environment as a Markov Decision Process. A perfect model means that given any current state and action, we can calculate the probability of each possible next state and the expected reward.

The key idea of DP is the use of Q function to organize and structure the search for good policies. It contains two components: policy estimation and policy improvement.

Policy evaluation: the task of policy evaluation is to compute Q^π for a policy π. The existence and uniqueness of Q^π are guaranteed as long as $\gamma<1$ (and in some other cases not relevant for our work, see e.g. [50]).

Starting from an initial approximation Q_0^π, each successive approximation is obtained by using the Bellman equation (7.11) as an update rule:

$$Q_{k+1}^\pi(s_t, u_t) = \hat{E}_\pi\{r_t + \gamma Q_k^\pi(s_{t+1}, \pi(s_{t+1}))\} \tag{7.11}$$

The new approximate Q_{k+1}^π is the expected sum of the old discounted value of next state-action pair $Q_k^\pi(s_{t+1}, \pi(s_{t+1}))$ and the immediate reward r_t. When $k \to \infty$, the sequence Q_k^π will converge to Q^π under the same conditions that guarantee the existence of Q^π.

Policy improvement: after we have determined the Q^π for an arbitrary policy π, we would like to know how to exploit this function Q^π, in order to construct another policy π' closer to optimality. One can show that the policy defined by

$$\pi'(s) = \arg\max_u Q^\pi(s, u) \tag{7.12}$$

is at least as good as π, and that if this π' is identical to π than we have found a globally optimal policy.

Once a policy π has been improved using Q^π to yield a better policy π', we can then compute $Q^{\pi'}$ and improve it again to yield an even better π'' until the process converges to an optimal policy π^* and optimal Q^{π^*}. This way of finding an optimal policy is called policy iteration. But one drawback to policy iteration is that each of its iterations involves policy evaluation, which is itself an iterative computation. A more efficient method is named

Q value iteration. In particular, as soon as one sweep of policy evaluation is finished, one improvement making the policy greedy with respect to the current Q function is made.

Classical DP algorithms are of limited utility in RL because of their assumption of a perfect environment model and their great computational expense, but DP algorithms provide an essential foundation for the understanding of other RL methods. In fact, all of these methods can be viewed as attempts to achieve the same effects as DP algorithms, only with less computation and without assuming a perfect model of the environment.

7.4.3 Monte Carlo methods

Unlike DP methods that need complete knowledge of the environment, Monte Carlo (MC) methods require only experiences–sample sequences of states, actions, and rewards from on-line or simulated interaction with the environment. These experiences are divided into episodes, and all episodes eventually terminate no matter what actions are selected.

Each occurrence of a state-action pair (s_t, u_t) in an episode is called a visit to (s_t, u_t). Given a set of episodes obtained by following π, MC methods estimate the expected return $Q^\pi(s_t, u_t)$ by averaging simply the returns observed after all visits to (s_t, u_t). As more returns are observed, the average return should converge to the expected value .

Within a given episode, the first time a state-action pair (s_t, u_t) is visited is called a first-visit to (s_t, u_t). The first-visit MC methods average just the returns following the first-visits to (s_t, u_t). However, the every-visit methods average the returns following all the visits to (s_t, u_t). Please note that the estimate for each state-action pair in MC methods are independent. The estimate for one state-action pair does not build upon the estimate of any other state-action pair, which can reduce computational expense of estimating when only the Q values of a subset of states are estimated.

One problem that MC Q estimation faces is that some relevant state-action pairs may never be visited, which will jeopardize the precision of Q estimation. Two kinds of methods are proposed to assure that all state-action pairs are encountered: on-policy MC methods and off-policy MC methods. One classical on-policy method is $\varepsilon - greedy$ method. It gives all nongreedy actions the minimal probability of selection, $\frac{\varepsilon}{U(s_t)}$, and the greedy action the remaining bulk of the probability, $1 - \varepsilon + \frac{\varepsilon}{U(s_t)}$. In off-policy methods, the policy used to generate episodes, called the behavior policy, may in fact be

unrelated to the policy that is evaluated and improved, called the estimation policy. It is the behavior policy with a nonzero probability of selecting all actions that assures all state-action pairs are encountered.

7.4.4 Temporal Difference Learning

Temporal Difference (TD) learning is a combination of MC methods and DP methods. Like MC methods, TD methods can learn directly from the raw samples without a model of the environment's dynamics. Like DP, TD methods update the estimates based in part on other learned estimates, and they are also bootstrapping methods.

On the other hand, DP, MC and TD methods all adhere to an essentially same idea: the Q function is repeatedly estimated to more closely approximate the real return function for a given policy, and the policy is improved with respect to the current Q function; the above steps are carried on iteratively to converge to the optimal policy and the optimal return function. The differences in these three methods are primarily differences in their approaches to policy estimation.

Based on some experiences following a policy π, TD learning estimates a new $Q^{\pi}_{k+1}(s_t, u_t)$ using the observed one-step reward r_t and the current estimate $Q^{\pi}_{k}(s_{t+1}, \pi(s_{t+1}))$, as shown by Eqn. (7.13). Here, $\hat{\alpha}$ is a constant step-size parameter.

$$Q^{\pi}_{k+1}(s_t, u_t) \leftarrow Q^{\pi}_{k}(s_t, u_t) + \hat{\alpha}[r_t + \gamma Q^{\pi}_{k}(s_{t+1}, \pi(s_{t+1})) - Q^{\pi}_{k}(s_t, u_t)]. \quad (7.13)$$

From Eqn. (7.13), we can see that on the one hand, TD methods sample the one-step reward r_t, like MC methods sampling a return R_t; on the other hand, they use the current Q estimate to calculate $Q^{\pi}_{k}(s_{t+1}, \pi(s_{t+1}))$. Therefore, TD methods combine the sampling of MC methods with the bootstrapping of DP methods.

Correspondingly, TD methods obtain the advantages over both MC and DP methods. Firstly, TD methods have an advantage over DP methods in which they do not require a model of the environment. Secondly, the most obvious advantage of TD methods over MC methods is that they update immediately the Q function after obtaining an one-step reward, not necessarily waiting until the end of an episode like MC methods.

7.5 Conclusions

In this chapter, learning definition is introduced at the beginning and next three learning forms, supervised learning, unsupervised learning and reinforcement learning are mentioned. As the focus of this chapter, reinforcement learning's policies, rewards and returns are detailedly discussed, and several primary RL solution approaches are given. These theoretical discussions about RL derive from [2, 50, 109]. Based on the prepared knowledge about RL, a RL-based supplementary damping control will be designed in the next chapter.

Chapter 8

RL applications for electromechanical oscillations damping

8.1 Introduction

Several RL applications have been proposed to damp electromechanical oscillations of power systems using its model-free characteristic and adaptability to varying system dynamics [54, 56, 59, 110]. Work presented in [54] takes the combinations of PSSs' gains as states, and then attempts to increase or decrease PSSs' gains with a fixed step to improve damping level. If one increase or decrease in PSSs' gains causes the damping ratio to be larger than a specific limit, the action will obtain a reward of 100. And otherwise, it will get a reward of -1. By this way, RL seeks for a near optimal control policy to coordinate different PSSs in a same power system. In order to improve the efficiency of Q-learning, a least worst action Q-learning algorithm is designed in paper [56]. In paper [110], RL is applied to adaptively tune the gain of SVC stabilizer following changes in load characteristics in order to provide the better electromechanical oscillation damping.

Actually, RL and MPC both can be formulated as an optimal control problem having discrete-time dynamics and costs as discussed in Chapter 2. Work [59] compares them in a unified framework. Different with MPC, RL infers in a model-free way closed-loop policies from a set of system trajectories and instantaneous rewards gathered from interaction with the real system or

from simulations. In a deterministic case and when a system model and cost function are available, the results of paper [59] about damping control in a single-machine infinite-bus system show that RL may be competitive with MPC.

Different with other works using RL for oscillations damping [52, 54–56] where Q-learning or some of its variants have been suggested, we use a model-free tree-based batch mode RL algorithm [57–59]. Furthermore, we propose to design a multi-agent system of heterogeneous non-communicating RL-based agents through separate sequential learning of individual agents [109, 111]. In addition, along some suggestions of the work in [59], this chapter explores possibilities to combine the proposed RL-based control with MPC.

The content of the present chapter is mainly based on our paper

[61] *D. Wang, M. Glavic, and L. Wehenkel. Trajectory-based supplementary damping control for power system electromechanical oscillations. IEEE Trans. Power Syst., Under second review.*

8.2 Tree-based batch mode RL (adapted from [57] and [58])

The tree-based batch mode RL method that we propose to use calculates in an iterative way an approximation of the optimal Q-function over a temporal horizon of T_h from a set of dynamic and reward four-tuples (s_t, u_t, r_t, s_{t+1}) (observe the state s_t at time t, take an action u_t, receive the next state s_{t+1} and the instantaneous reward r_t) [57]. It has two components: the extra-tree ensemble supervised learning method and the fitted Q iteration principle [57].

8.2.1 Extra-Tree ensemble based supervised learning

The supervised learning algorithm named Extra-Trees [58] builds each tree from the complete original training set. To determine a splitting at a node, it selects $\hat{K}$ cut-directions at random and for each cut-direction, a cut-point at random. It then computes a score for each of the $\hat{K}$ cut-directions and chooses the one that maximizes the score. All state-action pairs related to the node are split into the left subset or the right subset according to the

chosen splitting. The same procedure is repeated at the next node to be split. The algorithm stops splitting a node until stopping conditions are met.

Three parameters are associated to this algorithm: the number M of trees to be built to compose the ensemble model, the number $\hat{K}$ of candidate cut-directions at each node and the minimal leaf size $\hat{n}_{\min}$. The detailed extra-tree algorithm is given in Fig. 8.1. Inputting a state-action pair, each extra-tree outputs a Q value by averaging all samples' Q in the finally reached leaf node, and the output of an ensemble of extra-trees is the average of outputs of all extra-trees.

8.2.2 Fitted Q iteration principle

The fitted Q iteration algorithm calculates an approximation of the Q-function over a given temporal horizon by iteratively extending the optimization horizon:

- At the first iteration, it produces an approximation of Q_1-function corresponding to a 1-step optimization. Since the true Q_1-function is the conditional expectation of the instantaneous reward given by the state-action pair (i.e., $Q_1(s_t, u_t) = E\{r_t|(s_t, u_t)\}$), an approximation of it can be constructed by applying a batch mode regression algorithm, namely an ensemble of extra-trees whose inputs are state-action pairs (s_t, u_t) and whose outputs are instantaneous rewards r_t.

- The j-th iteration derives an approximation of Q_j-function corresponding to a j-step optimization horizon. The training set at this step is obtained by merely refreshing the outputted returns of the training set of the previous step by:

$$\hat{Q}_j(s_t, u_t) = r_t(s_t, u_t) + \gamma \max_{u_{t+1}} \hat{Q}_{j-1}(s_{t+1}, u_{t+1}) \tag{8.1}$$

Details about the tree-based batch mode RL are given in [57]. With the increase of iteration number, the $\hat{Q}_j$ function will gradually converge to a $\hat{Q}$ function over an infinite horizon. Out of considerations about computational complexity, a $\hat{Q}_j$ over a finite horizon of T_h is used in our experiments to substitute for the $\hat{Q}$ function over an infinite horizon.

Build_a_tree ($\mathcal{TS}$)
Input: a training set $\mathcal{TS}$, namely $\{(\hat{i}^{\hat{m}}, \hat{o}^{\hat{m}})\}_{\hat{m}=1}^{\#\mathcal{F}}$
Output: a tree $\hat{T}$.

- If

  (i) $\#\mathcal{TS} < \hat{n}_{min}$, or

  (ii) all input variables are constant in $\mathcal{TS}$, or

  (iii) the output variable is constant over the $\mathcal{TS}$.

  return a leaf labeled by the average value $\frac{1}{\#\mathcal{TS}} \sum_{\hat{m}} \hat{o}^{\hat{m}}$.

- Otherwise

  1. Let $[\hat{a}_{\hat{j}} < \hat{t}_{\hat{j}}]$=Find_a_test ($\mathcal{TS}$).
  2. Split $\mathcal{TS}$ into $\mathcal{TS}_l$ and $\mathcal{TS}_r$ according to the test $[\hat{a}_{\hat{j}} < \hat{t}_{\hat{j}}]$.
  3. Build from these subsets T_l =Build_a_tree ($\mathcal{TS}_l$) and T_r =Build_a_tree ($\mathcal{TS}_r$) ;
  4. Create a node with the test $[\hat{a}_{\hat{j}} < \hat{t}_{\hat{j}}]$, attach $\hat{T}_l$ and $\hat{T}_r$ as left and right subtrees of this node and return the resulting tree.

Find_a_test ($\mathcal{TS}$)
Input: a training set $\mathcal{TS}$, namely $\{(\hat{i}^{\hat{m}}, \hat{o}^{\hat{m}})\}_{\hat{m}=1}^{\#\mathcal{F}}$
Output: a test $[\hat{a}_{\hat{j}} < \hat{t}_{\hat{j}}]$.

1. Select $\hat{K}$ inputs, $\{\hat{a}_1, ..., \hat{a}_{\hat{K}}\}$, at random, without replacement, among all (non constant) input variables.
2. For $\hat{k}$ going from 1 to $\hat{K}$:

   (b) Compute the maximal and minimal value of $\hat{a}_{\hat{k}}$ in $\mathcal{TS}$, denoted respectively $\hat{a}_{\hat{k},min}^{\mathcal{TS}}$ and $\hat{a}_{\hat{k},max}^{\mathcal{TS}}$.

   (b) Draw a discretization threshold $\hat{t}_{\hat{k}}$ uniformly in $[\hat{a}_{\hat{k},min}^{\mathcal{TS}}, \hat{a}_{\hat{k},max}^{\mathcal{TS}}]$

   (b) Compute the score $\hat{S}_{\hat{k}}$ =Score $([\hat{a}_{\hat{k}} < \hat{t}_{\hat{k}}], \mathcal{TS})$
3. Return a test $[\hat{a}_{\hat{j}} < \hat{t}_{\hat{j}}]$ such that $\hat{S}_{\hat{j}} = \max_{\hat{k}=1,...,\hat{K}} \hat{S}_{\hat{k}}$.

Figure 8.1: Procedure used by the extra-tree algorithm to build a tree. Adapted from [57]

Inputs: a set $\mathcal{F}$ of four-tuples $\{(s_t^{\hat{m}}, u_t^{\hat{m}}, r_t^{\hat{m}}, s_{t+1}^{\hat{m}})\}_{\hat{m}=1}^{\#\mathcal{F}}$.
Outputs: an approximation $\hat{Q}$ of the Q-function.
Initialization:
Set j to 0.
Let $\hat{Q}_{T_h}$ be a function equal to zero everywhere on $S \times U$.
Iterations:

1. Repeat until stopping conditions are reached
   (a) -$j \leftarrow j + 1$.
   (b) -Build the training set $\mathcal{TS} = \{(\hat{i}^{\hat{m}}, \hat{o}^{\hat{m}})\}_{\hat{m}=1}^{\#\mathcal{F}}$ based on the function $\hat{Q}_{j-1}$ and on the set of four-tuples $\mathcal{F}$:

$$\begin{aligned} \hat{i}^{\hat{m}} &= (s_t^{\hat{m}}, u_t^{\hat{m}}), \\ \hat{o}^{\hat{m}} &= r_t^{\hat{m}} + \gamma \max_{u \in U} \hat{Q}_{j-1}(s_{t+1}^{\hat{m}}, u). \end{aligned} \tag{8.2}$$

   (c) Use the regression algorithm to induce from $\mathcal{TS}$ the function $\hat{Q}_{T_h}(s_t, u_t)$.
2. Return the function $\hat{Q}_{T_h}$

Figure 8.2: Fitted Q iteration algorithm. Adapted from [57]

Once we have the $\hat{Q}_{T_h}(s_t, u_t)$ function, we can use it to make damping control decision: for a state s_t, calculate all candidate actions' expected return by using the function $\hat{Q}_{T_h}(s_t, u_t)$, and select the action with the largest one as supplementary input to the existing controller. Details of fitted Q iteration algorithm, as applied for the problem studied in this thesis, are given in Fig. 8.2. In our simulations we have $s_t = (\delta_t, \omega_t, E'_{q_t}, \psi''_{d_t}, E'_{d_t}, \psi''_{q_t})$ and $u_t \in [-0.015, 0.015]$.

8.3 Test system and scenario

We test the tree-based batch mode RL still using the same test system in Fig. 4.2 and the same short-circuit scenario at bus 1, as in the simulations

reported in Part II of this thesis. Please notice that at the beginning, there is no MPC controller in this system. To reduce the amount of computations, we analyze system response only over a period of 8 seconds. Correspondingly, we also adjusted the power flow of the system to reduce the severity of this fault.

When controlled only through existing PSSs and TCSC the system exhibits poorly damped oscillations, as shown by dashed lines in Fig. 8.3 corresponding to the temporal evolution over a period of 8s of the power flow through line 1-2 and the angular speed of generator 1.

Next, the tree-based batch mode RL on a single generator and multiple generators is investigated together with the combined control effects between MPC and RL, all in the same medium size power system model [1].

8.4 RL-based control of a single generator

8.4.1 Sampling four-tuples

To collect a set of four-tuples (s_t, u_t, r_t, s_{t+1}), 500 system trajectories of 8 seconds under a series of random actions for different fault durations ranging from 0.01 to 0.05 seconds (this provides different initial conditions for each trajectory) are simulated. Every 0.1 seconds after the disturbance, the current states of a selected generator are sampled and a random action from an action space $[-0.015, 0.015]$ is applied. A candidate action space includes all possible supplementary inputs for the PSS on this generator. The action space is discretized at a step of 0.005. The system state s_{t+1} reached at next time is observed and the one-step local reward r_t by Eqn. (8.3) evaluated.

$$r_t = -\int_t^{t+1} |w - w_{ref}| \mathrm{d}t \tag{8.3}$$

where w and w_{ref} are the angular speed of the controlled generator and its rated angular speed. All in all, 2500 simulations (each over 8 seconds) are run, and a total of about 200000 four-tuples are thus collected.

8.4.2 Building extra-trees

Based on a four-tuple set, an extra-tree ensemble consisting of 100 trees ($M = 100$) is built using the method of [57]. We use the following tree

parameters: discount factor $\gamma = 0.95$; leave size $\hat{n}_{min} = 1$; splitting attribute number $\hat{K} = 7$ (six generator state attributes and one action attribute).

8.4.3 Q function based greedy decision making

Firstly, the Q approximation is iteratively calculated over an optimization horizon of $T_h = 10$. The obtained results show that the Q function over a finite horizon could produce good control effects.

At each control time, a damping action is determined as follows: collect the current states of controlled generator, select an action from the action space [-0.015, 0.015] and then calculate the Q value of this state-action pair by recursively searching all 100 trees and averaging their outputs. All candidate actions at current states are thus probed and the action with the largest Q value is selected as the optimal supplementary input to the PSS on the controlled generator.

Fig. 8.3 displays system response (solid lines) when the tree-based batch mode RL is applied only on generator 1. We observe that the introduction of this single supplementary control already improves the damping. Next, the same method is also applied on generator 2 and 3, and similar further improvements in damping are observed, as shown in Fig. 8.4. Clearly, the use of different generators would produce different, and hopefully complementary contributions to oscillations damping. Notice, however, that the optimal placement of the supplementary controllers is not considered in this work.

8.5 RL-based control of multiple generators

Interaction among RL-based controllers is a key problem to be solved when using them on multiple generators. Design of RL controllers on multiple generators should thus be carefully approached, because good or optimal solution of each individual controller when acting alone does not imply good or optimal solution to the system when all the controllers act together. To ensure the design of multiple controllers, three approaches can be adopted:

- Learn each controller's policy individually, in the absence of the other supplementary controllers, and benefit from the learned control policies by using them simultaneously. While this approach does not always ensure a good collective performance, it was nevertheless illustrated successfully in the previous section.

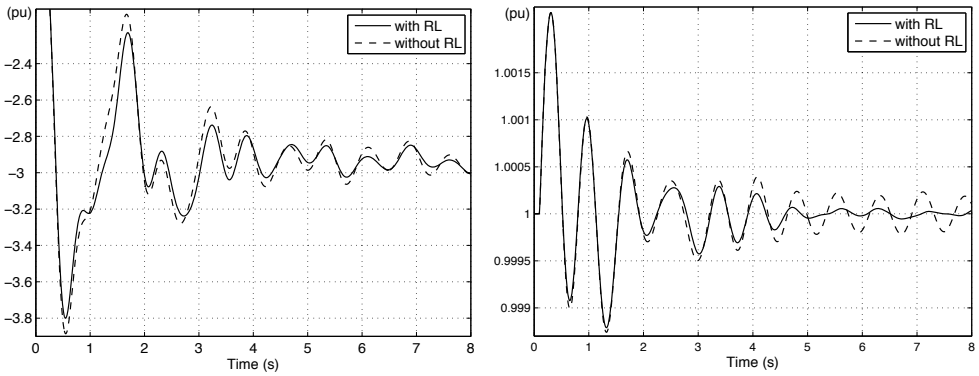

Figure 8.3: Generator 1 under RL control: active power of line 1-2 (left); angular speed of generator 1 (right)

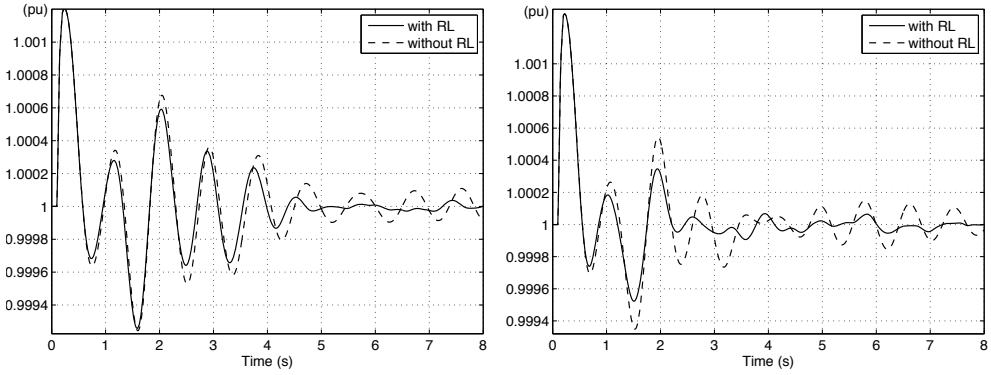

Figure 8.4: Generators 1, 2 and 3 under RL control: angular speeds of generator 2 (left) and generator 3 (right)

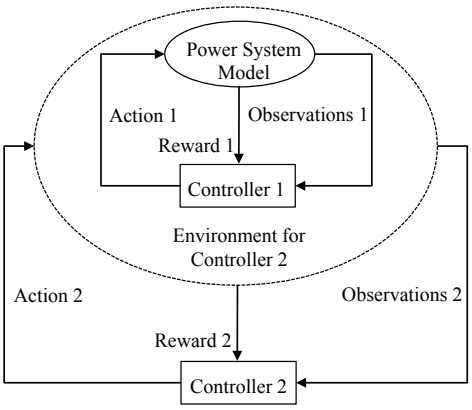

Figure 8.5: Separate sequential learning of controllers: 1 pass, in the order 1-2

- Separate sequential learning of the controllers (agents): a sequence of random actions to a first generator is first applied in order to yield its training samples while using the current control strategy for all other generators; the generator's states are recorded and corresponding one-step rewards computed; this is repeated until enough four-tuple samples about this generator are obtained and a new control policy is determined for this generator. Subsequently, the new control policy of this generator is used while applying the same sampling procedure on a second generator, and so on. In other words, one controller learns at a time and for each additional controller its working environment is considered to be the system together with all existing controllers that already learned to solve the task they are responsible for. In this way a multi-agent system of heterogeneous non-communicating agents [109, 111] is formed.

- Simultaneous asynchronous learning: four-tuple sets of all controlled generators come from the same trajectories and random actions are applied to multiple generators simultaneously to generate these trajectories.

We believe that the sequential (i.e. separate but coordinated) learning approach is most natural to create a well coordinated multi-controller system. This approach is illustrated in Fig. 8.5.

The intuition behind this approach is as follows. Let us imagine that there are 100 individual controllers in the system acting together. If one

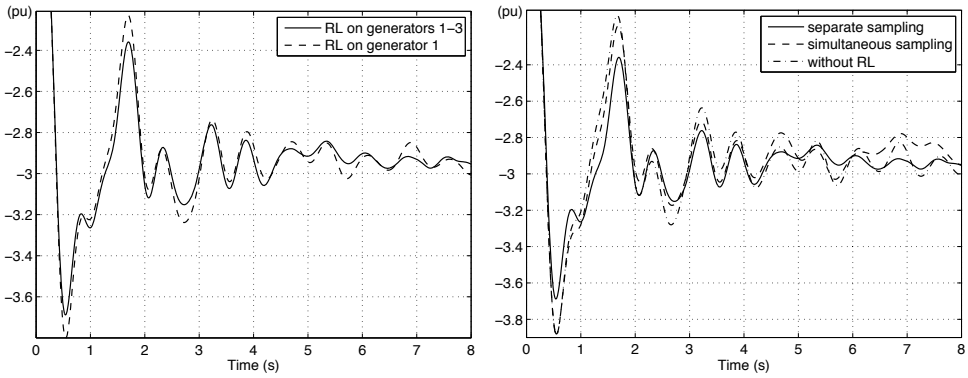

Figure 8.6: Multi-generator RL with a single 1-2-3 pass sequential learning (left); separate sequential sampling vs simultaneous asynchronous sampling (right). The curves represent the active power in line 1-2.

intends to add one more controller it is reasonable to adapt this controller to the system together with the already tuned other 100 controllers instead of re-learning all the 100 existing ones. Simulation results included in this section illustrate and support advantages of this intuition

The left figure of Fig. 8.6 compares the results when the batch mode RL is applied only on generator 1 and then together on generators 1, 2 and 3. We can see that more RL controllers bring indeed better damping effects.

Using the same number of four-tuples and the same extra-tree parameters, control effects of two ways (sequential vs asynchronous) of sampling are shown in the right figure of Fig. 8.6, when the tree based batch mode RL is applied to generator 1-3. The RL based on simultaneous sampling performs better than the scheme only using existing PSSs and TCSC in the first 6 seconds, but it brings large oscillations in the last 2 seconds. However, when compared with the RL based on the separate sequential sampling, the damping effects of the asynchronously learned controllers are clearly worse.

Specifically, in the four-tuple samples obtained using the simultaneous sampling, the dynamics and rewards of a generator are decided not only by the actions applied to itself, but also by the dynamics and actions of the other generators. However, when utilizing these samples to select an optimal action for one generator at a control time, real dynamics and actions of the two other generators are normally different from those of collected learning samples, because they then switch from random actions to their learned policy. This leads to the wrong estimation about the return of one action,

and possibly the wrong choice of action, which may jeopardize combined control effects. However, in the sequential sampling scheme, the rewards of a generator only represent consequences of its own actions, given the already tuned control policy used by the other generators, and hence it should lead to an improvement in damping each time a new controller is retrained. We hence use this approach (with a single 1-2-3 pass).

Remark. We do not consider related problems of determining optimal number of RL controllers and the effect of the order in which controllers are designed. These considerations are left for future research.

8.6 Combination of RL and MPC

The use of MPC to damp electromechanical oscillations has been investigated in our previous chapters. In this section, a possibility to combine control between MPC and the tree-based batch mode RL is further investigated. This possibility has been suggested in [59] through the comparison of MPC and RL in a unified framework. The results of [59] show that RL may certainly be competitive with MPC even when a good deterministic model is available.

When MPC is infeasible due to the limits of communications, measurements and models in real power systems, RL controllers based on local dynamics and local rewards are set to complete MPC's control effects. Here, we consider a combination of an MPC controller acting at the level of Area 2 control center to control generators 4-6 in this area with three RL-based controllers installed on generators 1-3 and obtained by sequential separate sampling (a single pass is used to train them in the order 1, 2, 3). The MPC state vector x includes generator, exciter, PSS, and turbine governor states. Output variables are angular speeds of generators 4-6. The input u is a vector of supplementary signals for PSSs on generators 4-6, which is subject to $-0.015 \leq u \leq 0.015$. Prediction horizon of $T_h = 10$ and control horizon of $T_c = 3$ steps are chosen. In the objective function (2.6), all deviations of the predicted outputs from references are weighted uniformly and independently, i.e. W_{y_i} is the identity matrix. The MPC controller considers $\pm 10\%$ state estimation errors and a 0.05s delay. It updates every 0.1 seconds supplementary damping signals for PSSs on generators 4-6.

Fig. 8.7 shows control response in terms of tie line 1-2 active power flow of the MPC controller alone, the RL controller alone, and when the two schemes are used in combination (while the RL-controllers have been trained

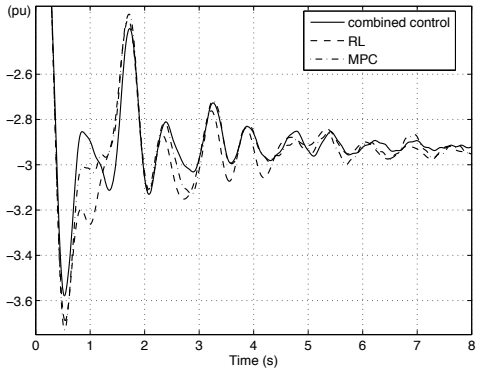

Figure 8.7: Combination of RL-based control and MPC in a control center. Active power flow in line 1-2.

with taking into account the effect of the MPC controller on generators 4-6). We observe that the combined scheme indeed shows better performances with respect to the sole use of either MPC on generators 4-6 or RL on generators 1-3.

8.7 The use of a global reward signal

The proposed RL controller can also incorporate some remote information which represents, to some extent, system-wide dynamics to define its reward, in order to explicitly target global control effects.

In order to illustrate this possibility, we make the following modifications with respect to the previous setting:

1. The active power deviation from steady state of tie-line 1-2 is used as global reward signal for the training of the RL controller on generator 1. The state vector of the controller is however kept unchanged, and so still uses only the local measurements.

2. The controller of generator 1 is re-trained based on a sample of four-tuples obtained by repeatedly simulating system response under a series of random actions and collecting corresponding active power in line 1-2 together with the local state variables of generator 1, while the two other controllers (in generators 2 and 3) where previously trained by using their local reward signals. Specifically, we still consider the

three-phase short circuit to ground at bus 1, and the RL controller on generator 1 applies a random action every 0.1 seconds after the fault.

3. At every time step, it collects the active power of line 1-2, and calculates its one-step instantaneous reward as:

$$r_t = -\int_t^{t+1} |P_{1-2} - P_{ref}| \mathrm{d}t, \tag{8.4}$$

where P_{1-2} is the measured active power of line 1-2, and P_{ref} is its reference which is the steady state exchange power before the occurrence of the disturbance (it could also be a new post-disturbance exchange power determined by off-line simulations).

Fig. 8.8 displays the results when the RL controller installed on generator 1 is learnt based on the global reward signal (solid line) while the other two RL controllers (at generators 2 and 3) use local reward signals (solid line) and when all three RL controllers use local reward signals as inputs (dashed line). Remind that the local signal here means the generator angular speed. Better damping of the inter-area electromechanical oscillations is clearly observed with the RL controller installed on generator 1 learnt based on the global reward signal. This illustrates the flexibility of the proposed control in enhancing the damping of different oscillation modes by focusing on dominant ones depending on system prevailing conditions. RL signals of generator 1 corresponding to different rewards are given in Fig. 8.9

Notice that in addition to using remote measurements for the reward definition, one could as well use such remote measurements to extend the definition of the state-vector of local controllers. However, while rewards are used only at the (off-line) training stage of the RL controller when computing the Q-function approximation from samples, state variables must also be used in real-time to actually compute the optimal decisions, and hence real-time communication delays would thus have to be taken into account to evaluate this option in a realistic way. On the other hand, the use of remote signals to define the reward only requires to collect time-tagged remote measurements during a certain period of time together with the time-tagged local measurements so as to form a set of four-tuples that can be exploited off-line for learning the Q-function approximation. This latter approach is hence less demanding in terms of communication infrastructure requirements, and also less sensitive to possible losses of communication channels in real-time.

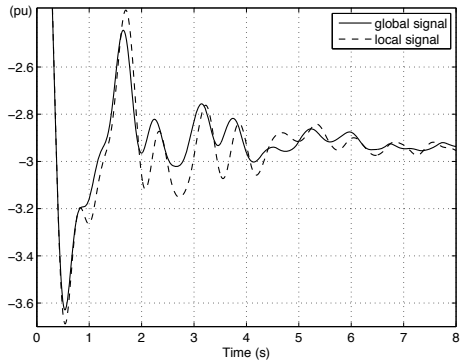

Figure 8.8: Active power of line 1-2: global reward vs. local reward

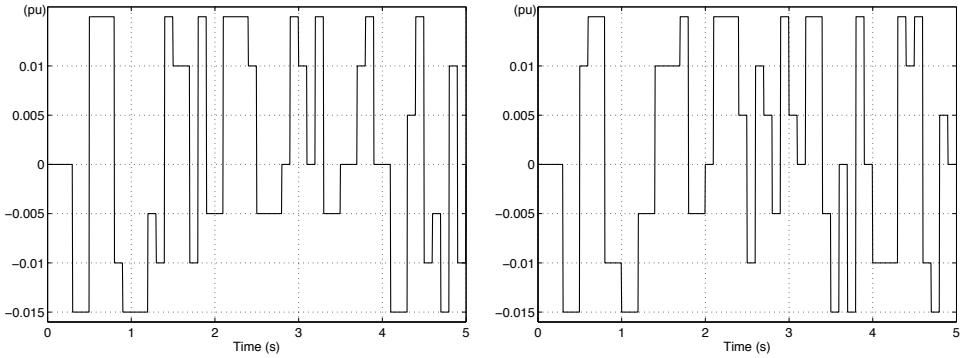

Figure 8.9: RL signals of generator 1: local reward (left), and global reward (right)

8.8 Comparison with an existing method

Finally, the proposed RL supplementary control is compared with existing damping control methods, using the modal analysis method from [1] as an example. All generators, except generators 7 and 14, are assumed to have a PSS and their gains are optimized using root locus to obtain a damping ratio larger than 0.05, and the time constants are calculated according to the needed phase compensation. When a same three phase short circuit fault as above occurs, the response of active power in tie-line 1-2 is shown by the dashed line in Fig. 8.10.

Next, three RL-based controllers are installed on generators 1, 2, and 3. These controllers use the global signal of active power of tie-line 1-2. The

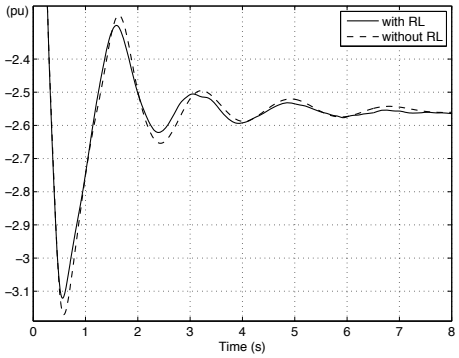

Figure 8.10: Comparison with existing methods

solid line of Fig. 8.10 shows that the supplementary control signals on three generators could further improve the effects of existing controllers even if they are optimized in some way. Although the improvement is small, it is significant especially considering the difficulty of optimizing PSS parameters in practice. Moreover, the comparison also shows the robustness of the proposed method with respect to different operation conditions and damping levels.

8.9 Conclusions

The chapter focuses on applying a tree-based batch mode RL to learn supplementary damping controls from samples of observed trajectories. The results on a single generator show that the supplementary inputs calculated by using this method can further improve damping effects of existing controllers. When the tree-based batch mode RL on multiple generators is used, a separate sequential sampling and learning for each generator's supplementary control is the most appropriate solution so as to effectively coordinate the different supplementary controls. This method can also be combined with MPC to complete its control effects and cope with its modeling errors.

One of the main advantages of the learning-based strategy is that it does not rely on accurate analytical models of the system dynamics, but rather exploits directly measurements about the past performance of the system based on already observed system trajectories. It is therefore a promising approach to cope with the emerging features of power systems, whose dynamics more and more depend on the dynamics of loads and dispersed generation which

incorporation into global dynamic models would be a daunting task.

One practical problem about the tree-based batch mode RL approach is that it needs important computational resources to build and exploit large enough ensembles of extremely randomized trees. So, the future work would attempt to use some more efficient learning algorithms.

One essential advantage of the learning-based approach proposed in this paper is its very generic nature, so that it could be used even at the very local level of load and dispersed generation control, in order to "smarten" the whole power system control strategy at any layer.

Part IV

Conclusions and future work

Chapter 9

Conclusions and future work

9.1 Conclusions

At the beginning of this thesis, we discussed oscillation modes, mechanisms, analysis methods and controls to understand better electromechanical oscillations. Next, we presented research motivations, and elaborated the proposed trajectory-based supplementary damping control in terms of its feasibility, overall principle, mathematical formulation of the control objective, solution approaches and implementation strategy.

Different with the conventional damping control, this thesis proposed a trajectory-based supplementary damping control. It is formulated as a T_h-step optimal control problem with discrete dynamics and rewards, which is solved using MPC and RL methods to obtain some optimized supplementary signals for existing damping controllers. These supplementary signals are superimposed on the outputs of existing controllers, to make post-disturbance generator speeds return to and remain near the reference speed. When all generators run at the reference speed, electromechanical oscillations are damped. On the one hand, these supplementary inputs can coordinate efficiently different damping controllers to get better global effects. On the other hand, they are continuously updated at each control time to improve the adaptivity of damping control with respect to power system evolution.

As the first part of this research, MPC is applied to realize the proposed supplementary damp control. A centralized MPC scheme based on a linearized discrete-time state space model of a power system, is first designed as a baseline. A distributed MPC scheme is introduced next in which each

MPC controller solves its optimization subproblem using a detailed model of its own area and a rough model of the remaining areas. It then sends the supplementary signals to local damping controllers under its responsibility. The common control target is used to implicitly coordinate these MPC controllers. Finally, a two-level hierarchical multi-area MPC scheme is introduced with the aim to enhance control robustness. The upper MPC controllers optimize supplementary signals of damping controllers in their own areas and send them to the lower MPC controllers as their input bases. Using local models and measurements, the lower MPC controllers correct these input bases.

The performances of the proposed control schemes are tested using a 16-generator, 70-bus test system, both in ideal conditions and considering realistic state estimation errors and communication delays. The centralized MPC scheme can effectively and quickly damp inter-area oscillations, work with less available controllers and incorporate different types of controllers. The distributed scheme appears to be a viable control strategy for large-scale systems while the hierarchical MPC further improves control effects of the distributed one, and offers at the same time much better robustness.

Considering the possible influence of model imprecision on MPC control effects in practice, RL is considered to design the proposed damping control in the second part of the thesis. Specifically, a tree-based batch mode RL is applied to calculate supplementary signals for existing controllers. Based on a set of dynamic and reward samples, the method approximately calculates the control return over a temporal horizon for each state-action pair, and then applies the action with the largest control return at current state. Our results on a single generator show that the supplementary signals calculated in such a way can improve further damping effects of existing controllers. A separate sequential sampling and learning for each generator's supplementary control is showed to be the most appropriate to deal with the interaction among multiple RL-based controllers. Finally, the RL-based control is combined with MPC to complement their advantages in oscillations damping.

9.2 Future work

There is still room for many improvements in the proposed supplementary damping control schemes for their considerations as viable practical solutions in real power systems. As far as MPC-based schemes are concerned, several improvements could be made from the following aspects:

- Models and measurements: building the models that could exactly represent real power system dynamics and updating them following operation changes are important for MPC control effects. But on the other hand, more exact models involve more state measurements. Considering the limits of current state estimation technologies, one compromise is to use the reduced-order models that could approximate enough system dynamics and at the same time only use those states measurable. In case it is necessary to use some states unmeasurable, artificial intelligence may be used to estimate their value distributions.

- Solvability: a useful MPC scheme must assure that it can find a control policy in any condition, while meeting practical operation constraints and control constraints. In all cases involved by this thesis, the MPC optimization is solvable. If unsolvable a primary approach should be temporary relaxation of the least-important constraints.

- Stability and robustness: we have verified the stability and robustness of proposed schemes in different operation conditions and failure scenarios in a medium-size equivalent power system. But some theoretical investigations in MPC stability are possible in the future work, like introducing terminal state constraints, or using the infinite-horizon MPC formulations. In addition, MPC robustness is to be further investigated under more operation conditions and failure scenarios in large-scale power systems.

- Coordination between individual MPC controllers: a distributed MPC scheme is more viable in practical power systems, considering the limits on the measurement, communication and computation, reliability and robustness. The coordination between individual MPC controllers is very important for obtaining good global control effects. In the thesis, one kind of implicit coordination based on subsystem-wise MPC is used. Possible further coordination could be made as follows: determine in advance the post-fault exchange power limits between subsystems by simulating possible oscillation scenarios and then impose them as coherency constraints between subsystems on individual MPC optimization.

- Coupling between two layers of MPC: in the proposed hierarchical MPC scheme, the upper MPC controller calculates supplementary sig-

nal bases for all lower MPC controllers under its authority, and the lower MPC controllers are responsible to calculate some corrections to these bases. In some sense, the corrections of lower MPC controllers could be considered kind of disturbances to the upper MPC controller. If the magnitude of disturbances is too large, it maybe destroys the current control effects of the upper MPC controller. For the time being, we determine the magnitude of lower corrections with the help of repeated simulation trials. Another possible solution dealing with the coupling is to integrate these corrections to the state space model used by the upper MPC optimization, namely.

$$\begin{aligned} \hat{x}[k+1|k] &= A\hat{x}[k|k] + B\hat{u}[k|k] + B_c\hat{c}[k|k]; \\ \hat{y}[k|k] &= C\hat{x}[k|k]. \end{aligned} \tag{9.1}$$

where $\hat{c}[k|k]$ is a vector of corrections at time k. Normally, the upper MPC controller does not know the lower corrections, so we could use some random noises to represent these corrections.

- Practical constraints: in the current work, random measurement errors and a common delay are used. More considerations should be given to some other combinations of SE errors (more in accordance with devices' performances) and delays (different for different measurements and controls). Moreover, the performances of proposed MPC schemes under other oscillations forms, like the ambient oscillations excited by continuous load variations, are to be further tested.

RL selects supplementary signals by learning from the collected dynamic and reward samples. Therefore, the completeness of collected samples with respect to possible operation conditions and fault scenarios will influence RL control effects in practice. The sample set used in the thesis is to be further completed to make RL enable to deal with more different operation conditions and fault scenarios. However, with the increase of samples, computation becomes a pressing problem for the used tree-based batch mode RL algorithm. Investigating more efficient learning algorithms is significant in the future work. Moreover, an alternative is to attempt to reduce the number of states included in samples on the premise of assuring control effects, in order to reduce RL computation. How to select the necessary states that are enough to represent system dynamics will be explored in the future work.

Part V

Appendices

Appendix A

Test system

We use a 16 generators, 70 buses system to illustrate our proposed supplementary damping control schemes, which is a reduced-order equivalent model of the New England/New York interconnected system. A single line diagram of the system is shown in Fig. A.1. There is a DC exciter, a Power System Stabilizer (PSS) and a Turbine Governor (TG) on each generator. Generators 13-16 are equivalent generators. Compared with the original system of book [1], out of research need, a Thyristor Controlled Series Compensator (TCSC) is installed between bus 69 and bus 70 to adjust active power of tie-line 8-9, and a tie-lie 1-27 is deleted. We simply divide the system into two areas according to its geographical structure: area A1 and area A2. The TCSC is assigned to area A1 as its control resource. We also adjust power flow at steady state. Fig. A.2 gives injection powers at load and generator nodes. Fig. A.3 indicates line transmission powers.

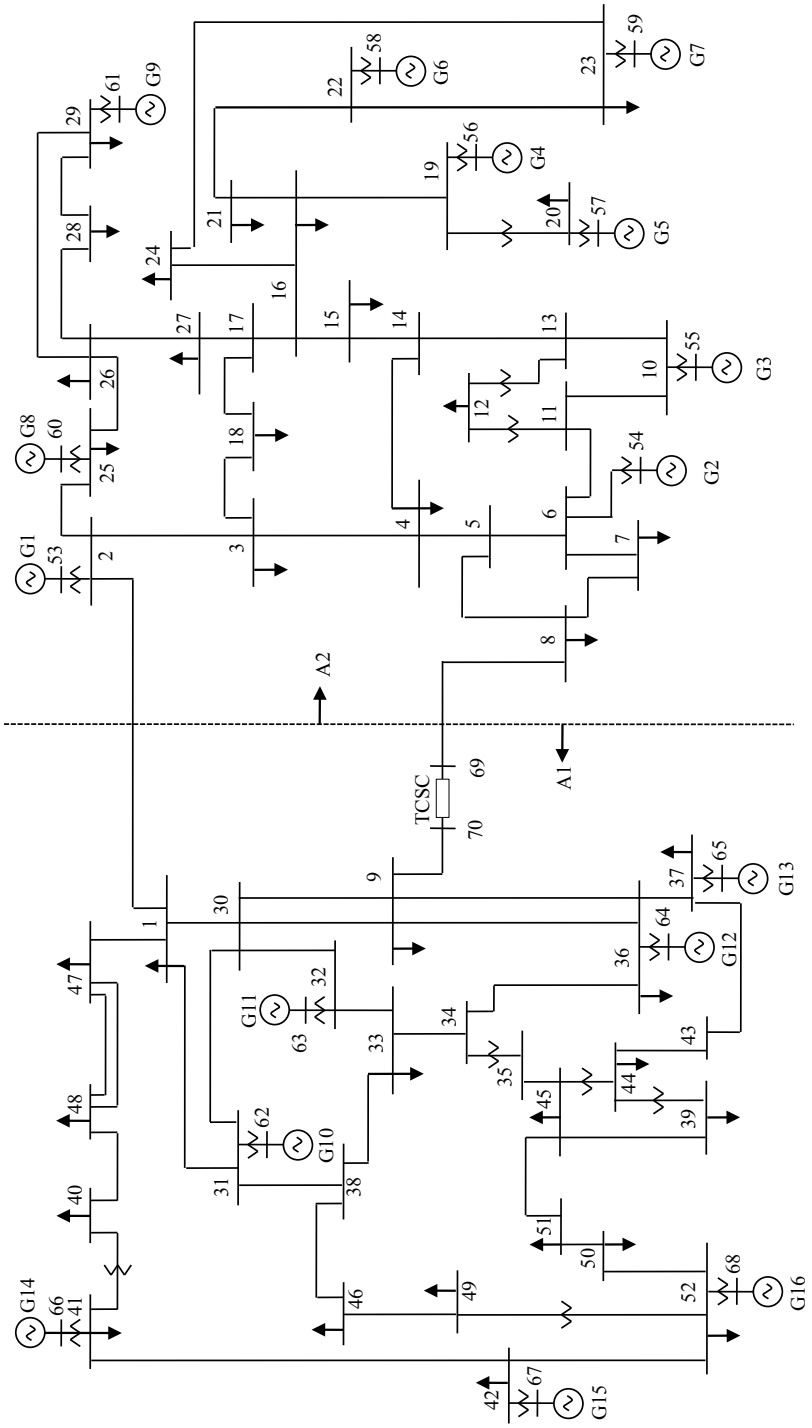

Figure A.1: 16 generators / 2 areas / 70 buses system

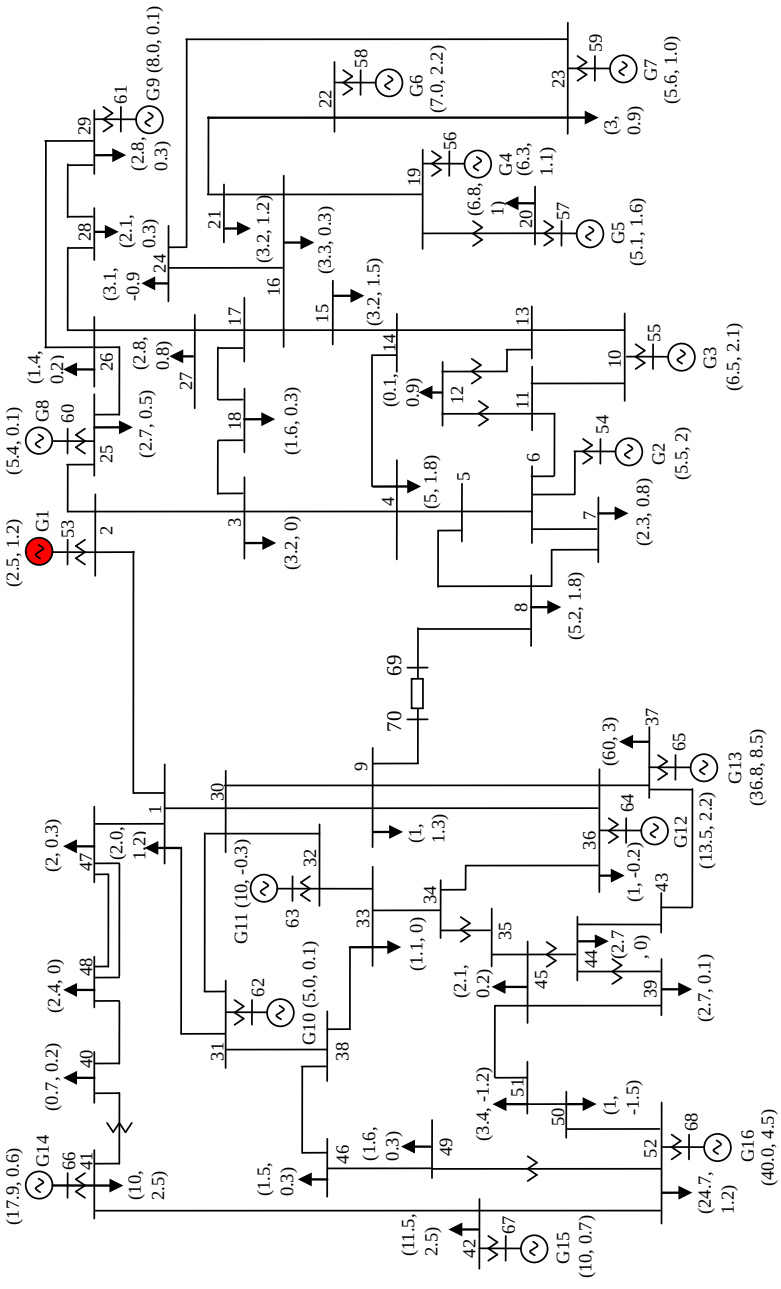

Figure A.2: Load and generation

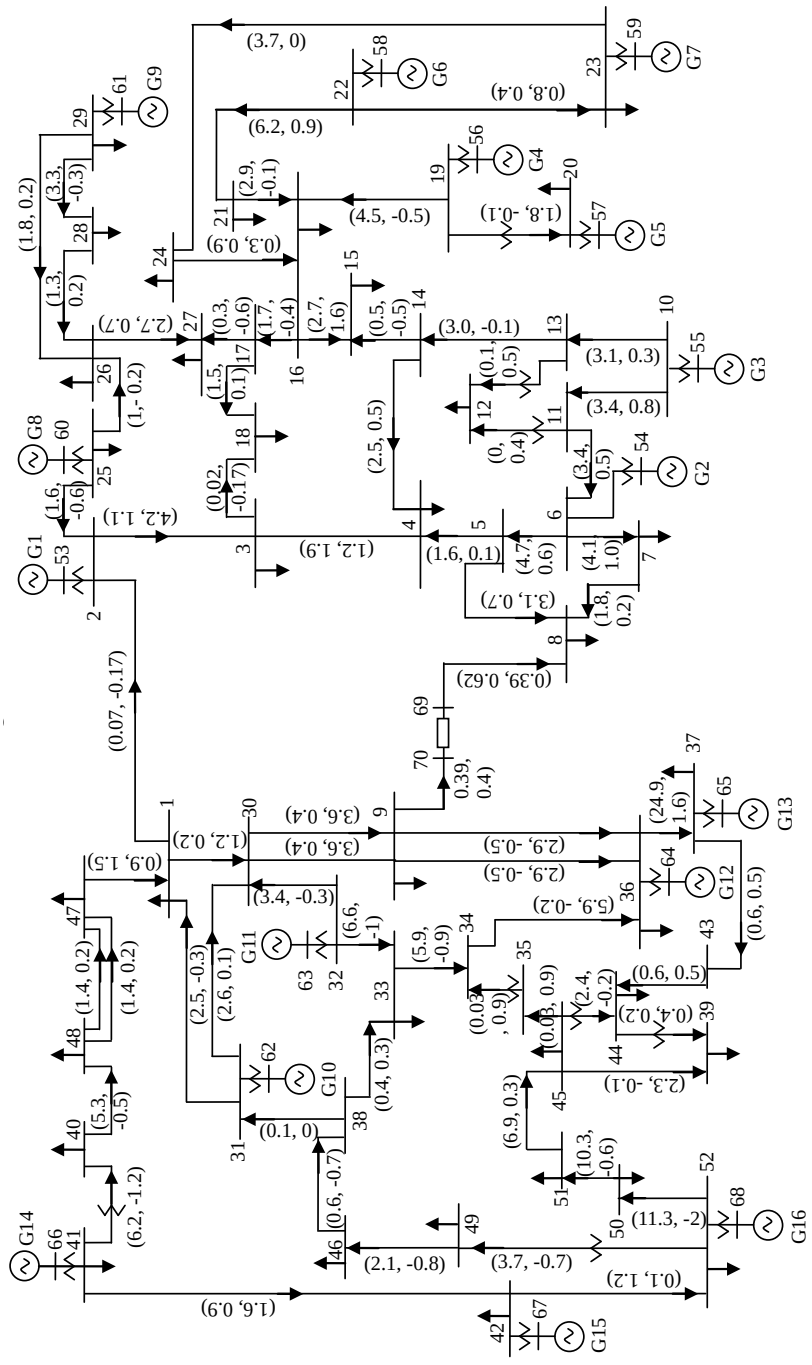

Figure A.3: Transmission line power flow

Appendix B

Power System Toolbox (adapted from [112])

Power System Toolbox (PST) was conceived and initially developed by Dr. Kwok W. Cheung and Prof. Joe Chow at the Rensselaer Polytechnic Institute during the early 1990s. From 1993 to 2009, it was marketed, and further developed by Graham Rogers (formerly Cherry Tree Scientific Software). Now, it is in use by utilities, consultants and universities world-wide.

PST consists of a set of coordinated MATLAB m-files which model the power system components necessary for power system flow and stability studies. The toolbox comes with the m-files, demo examples of how the models can be used, several sets of dynamic data and a user's manual. Since source codes for all functions are provided, a user may create easily and efficiently new dynamic models or modify existing models complying with the given conventions, in order to meet special modeling or simulation requirements. He/she also can connect PST conveniently with other MATLAB programs. This is the main reason that we selected the PST as simulation tool. Next, we introduce briefly three functions of PST: load flow calculation, transient stability simulation and small signal stability analysis.

B.1 Load flow calculation

In power systems, load flow study is performed to obtain a set of feasible steady operation conditions which obey certain power balance and security constraints. Type "lfdemo"(an AC load flow driver) or "lfdc" (an load flow

driver involving HVDC links) in the MATLAB command window, and then choose a data file that specifies system structure together with generators' power outputs, and system's active and reactive power loads. Bus voltage magnitudes and angles are calculated by solving the nonlinear algebraic network equations so that the specified loads are supplied. If the voltages at buses are found to be out of secure limits during load flow iterations, the corresponding transformer taps are adjusted to bring load voltages back in range. Finally, either load flow has converged, or the number of allowed iterations has been exceeded. In either case, the user is given a list of solutions viewing options.

B.2 Transient stability simulation

PST provides the models of machines and control systems in the form of MATLAB functions, for performing transient stability simulations, and for small signal stability analysis and damping controller design. A solved load flow case is required to set the operating condition used to initialize dynamic device models. A fault is defined in the data specification card 'sw_con'. The driving function 's_simu' provides a transient simulation environment which requires the data file specifying system structure, controller and faults, like stand-alone transient programs. It calculates transient response by solving a set of differential equations determined by the dynamic models and a set of algebraic equations determined by the power system network.

One thing that must be mentioned is that PST provides a way of modulating controller reference inputs in transient simulations. Our researches just make use of it to superimpose supplementary control signals on existing controllers, in order to improve damping control effects.

B.3 Small signal stability analysis

The stability of operating point of a dynamic system to small disturbances is termed small signal stability. In order to analyze small signal stability, nonlinear dynamic differential equations must be linearized about a steady state operating point to get a set of linearized state space equations, like:

$$\begin{aligned} x[k+1|k] &= Ax[k|k] + Bu[k|k]; \\ y[k|k] &= Cx[k|k] + Du[k|k]. \end{aligned} \tag{B.1}$$

where A is the state matrix; B is the input matrix; C is the output matrix; D is the feed forward matrix; x is the state vector, y is the output vector and u is the input vector.

In PST, the linearization is performed by calculating numerically Jacobian matrix. That is to say, starting from the states determined by model initialization, a small perturbation is applied to each state in turn. The changes in the rates of change of all states divided by the magnitude of the perturbation give a column of state matrix corresponding to the disturbed state. A permutation matrix p_mat is used to arrange the disturbed states in a logical order. After each perturbation, the perturbed state is returned to its original equilibrium values. The input matrix B, the output matrix C and the feed forward matrix D can be determined in a similar manner. A single driver, svm_mgen is provided for small signal stability, which could calculate eigenvalues, damping ratios, participation factors, and so on.

More details about PST functions and algorithms can be found in manual [112].

Appendix C

Power system models (adapted from [112])

This section presents linearized models of relevant power system components used in MPC.

C.1 Generator

$$
\begin{aligned}
\tfrac{d\delta}{dt} &= \Delta\omega \\
J\tfrac{d\omega}{dt} &= T_m - T_e - \hat{D}\Delta\omega \\
\Psi''_d &= \tfrac{E'_q(X''_d - X_l) + \Psi_{1d}(X'_d - X''_d)}{X'_d - X_l} - X''_d I_d \\
\tfrac{d\Psi_{1d}}{dt} &= \tfrac{1}{T''_{d0}}(E'_q - \Psi_{1d}) \\
\tfrac{dE'_q}{dt} &= \tfrac{1}{T'_{d0}}(E_{fd} - X_{ad} I_{fd}) \\
\Psi''_q &= \tfrac{E'_d(X''_q - X_l) + \Psi_{1q}(X'_q - X''_q)}{X'_q - X_l} - X''_q I_q \\
\tfrac{d\Psi_{1q}}{dt} &= \tfrac{1}{T''_{q0}}(E'_d - \Psi_{1q}) \\
\tfrac{dE'_d}{dt} &= -\tfrac{1}{T'_{q0}} X_{aq} I_{1q} \\
X_{ad} I_{fd} &= \tfrac{(X'_d - X''_d)(X_d - X'_d)}{(X'_d - X_l)^2}[E'_q - \Psi_{1d} \\
&\quad + \tfrac{(X'_d - X_l)(X''_d - X_l)}{(X'_d - X''_d)} I_d] + f_{sat}(E'_q) \\
X_{aq} I_{1q} &= \tfrac{(X'_q - X''_q)(X_q - X'_q)}{(X'_q - X_l)^2}[E'_d - \Psi_{1q} \\
&\quad + \tfrac{(X'_q - X_l)(X''q_- X_l)}{(X'_q - X''_q)} I_q] + E'_d
\end{aligned}
\tag{C.1}
$$

where J is the moment of inertia; T_m is the mechanical torque; T_e is the electromechanical torque; $\hat{D}$ is the damping coefficient; Ψ''_d and Ψ''_q are the d axis and q axis components of stator flux linkage; E'_d and E'_q are the d axis and q axis transient stator voltages; Ψ_{1d} and Ψ_{1q} are the amortisseur circuit flux linkages; X_l is the leakage reactance; X_d, X'_d and X''_d are the d axis synchronous reactance, transient reactance and subtransient reactance; T'_{d0} and T''_{d0} are the open circuit time constant and open circuit subtransient time constant of d axis; X_q, X'_q and X''_q are the q axis synchronous reactance, transient reactance and subtransient reactance; T'_{q0} and T''_{q0} are the open circuit time constant and open circuit subtransient time constant of q axis; I_d and I_q are the d axis and q axis stator currents; X_{ad} and X_{aq} are the mutual reactances; E_{fd} and I_{fd} are the excitation voltage and current; f_{sat} is saturation coefficient.

C.2 Exciter

$$\begin{array}{rl} V_{err} & = E_{sig} + V_{ref} + V_{pss} - V_{ter} \\ \frac{dV_{As}}{dt} & = \frac{V_{err} - V_{As}}{T_b} \\ V_A & = \frac{T_c}{T_b} V_{err} + (1 - \frac{T_c}{T_b}) V_{As} \\ \frac{dE_{fd}}{dt} & = \frac{K_a V_A - E_{fd}}{T_a} \end{array} \tag{C.2}$$

where V_{err} is the voltage deviation; E_{sig} is the supplementary input signal; V_{ref} is the voltage reference; V_{pss} is the output of the PSS; V_{ter} is the terminal voltage; V_{As} and V_A are the regulator states; K_a is the voltage regulator gain; T_a is the voltage regulator time constant; T_b and T_c are the transient gain reduction time constants.

C.3 Turbine governor

$$\begin{array}{rl} \frac{dTG_1}{dt} & = \frac{(TG_{in} - TG_1)}{T_s} \\ \frac{dTG_2}{dt} & = \frac{(1 - \frac{T_3}{T_c}) TG_1 - TG_2}{T_c} \\ \frac{dTG_3}{dt} & = \frac{(TG_2 + \frac{T_3}{T_c} TG_1)(1 - \frac{T_4}{T_5}) - TG_3}{T_5} \\ P_m & = TG_3 + \frac{T_4}{T_5}(TG_2 + \frac{T_3}{T_c} TG_1) \end{array} \tag{C.3}$$

where TG_{in} is the input of turbine governor; TG_1, TG_2 and TG_3 are the state variables; P_m is the mechanical power; T_s is the servo time constant; T_c is the HP turbine time constant; T_3 is the transient gain time constant; T_4 is the time constant to set HP ratio; T_5 is the reheater time constant.

C.4 PSS

$$\begin{aligned}
\frac{dPSS_1}{dt} &= \frac{(PSS_{in}-PSS_1)}{T_w} \\
\frac{dPSS_2}{dt} &= \frac{(1-\frac{T_{n1}}{T_d1})G_{pss}\frac{dPSS_1}{dt}-PSS_2}{T_{d1}} \\
\frac{dPSS_3}{dt} &= \frac{(1-\frac{T_{n2}}{T_d2})\frac{dPSS_2}{dt}-PSS_3}{T_{d2}} \\
V_{pss} &= \frac{T_{n2}}{T_d2}(\frac{T_{n1}}{T_{d1}}G_{pss}\frac{dPSS_1}{dt}+PSS_2)+PSS_3
\end{aligned} \tag{C.4}$$

where PSS_{in} is the input of PSS, PSS_1, PSS_2, and PSS_3 are the PSS states; G_{pss} is the PSS gain; T_w is the washout time constant; T_{n1} and T_{n2} are the lead time constants; T_{d1} and T_{d2} are the lag time constants.

C.5 TCSC

$$\frac{dX_{tcsc}}{dt} = \frac{K_r TCSC_{in} - X_{tcsc}}{T_r} \tag{C.5}$$

where X_{tcsc} is the TCSC output; $TCSC_{in}$ is the TCSC input signal; K_r is the TCSC gain; T_r is its time constant.

More details about the used power system models can be found in manual [112].

Bibliography

[1] G. Rogers. *Power system oscillations*. Power electronics and power systems series. Kluwer academic publishers, Massachusetts, USA, 2000.

[2] L. Wehenkel, M. Glavic, P. Geurts, and D. Ernst. Automatic learning for advanced sensing, monitoring and control of electric power systems. In *Proceedings of the second Carnegie Mellon conference in electric power systems*, January 2006.

[3] M. Klein, G. Rogers, and P. Kundur. A fundamental study of inter-area oscillations in power systems. *IEEE Trans. Power Syst.*, Vol. 6(3):914–921, August 1991.

[4] Annual report 2012. Technical report, ENTSO-E, Available online: https://www.entsoe.eu/publications/general-publications/annual-reports/, 2012.

[5] Yearly statistics & adequacy retrospect 2012. Technical report, ENTSO-E, Available online: https://www.entsoe.eu/publications/statistics/yearly-statistics-adequacy-retrospect/, 2012.

[6] E. Grebe, J. Kabouris, S. L. Barba, W. Sattinger, and W. Winter. Low frequency oscillations in the interconnected system of continental Europe. In *Proceedings of PES general meeting*, Minneapolis, Minnesota, USA, July 2010.

[7] J. Lehner, T. Weissbach, and G. Scheffknecht. Oscillation behavior of the enlarged UCTE power system including the Turkish power system. In *Proceedings of the 17th IFAC world congress*, Seoul, Korea, July 2008.

[8] G. P. Liu, Z. Xu, Y. Huang, and W. L. Pan. Analysis of inter-area oscillations in the South-China interconnected power system. *Electr. Pow. Syst. Res.*, Vol. 70(1):38–45, 2004.

[9] V. Venkatasubramanian and Y. Li. Analysis of 1996 western American electric blackouts. *Bulk power system dynamics and control*, Vol. 6:685–721, August 2004.

[10] P. Kundur. *Power system stability and control*. McGraw Hill Professional, New York, USA, 1994.

[11] G. Rogers. Power system structure and oscillations. *IEEE Comput. Appl. Power*, Vol. 12(2):14, 16, 18, 20–21, April 1999.

[12] F. P. Demello and C. Concordia. Concepts of synchronous machine stability as affected by excitation control. *IEEE Trans. Power Appar. Syst.*, Vol. 99(4):316–329, April 1969.

[13] J. V. Milanović and I. A. Hiskens. Effects of load dynamics on power system damping. *IEEE Trans. Power Syst.*, Vol. 10(2):1022–1028, May 1995.

[14] I. A. Hiskens and J. V. Milanović. Locating dynamic loads which significantly influence damping. *IEEE Trans. Power Syst.*, Vol. 12(1):255–261, February 1997.

[15] C. D. Vournas, N. Krassas, and B. C. Papadias. Analysis of forced oscillations in a multimachine power system. In *Proceedings of international conference on control*, Edinburgh, UK, March 1991.

[16] I. Dobson, J. F. Zhang, S. Greene, H. Engdahl, and P. W. Sauer. Is strong modal resonance a precursor to power system oscillations ? *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, Vol. 48(3):340–349, March 2001.

[17] V. Auvray, I. Dobson, and L. Wehenkel. Modifying eigenvalue interactions near weak resonance. In *Circuits and Systems, 2004. ISCAS'04. Proceedings of the 2004 International Symposium on*, volume 5, pages 992–995. IEEE, 2004.

[18] V. Ajjarapu and B. Lee. Bifurcation theory and its application to nonlinear dynamical phenomena in an electrical power system. *IEEE Trans. Power Syst.*, Vol. 7(1):424–431, February 1992.

[19] F. Howell and V. Venkatasubramanian. Transient stability assessment with unstable limit cycle approximation. *IEEE Trans. Power Syst.*, Vol. 14(2):667–677, May 1999.

[20] J. J. Sanchez-Gasca, V. Vittal, M. J. Gibbard, A. R. Messina, and et al. Inclusion of higher order terms for small-signal (modal) analysis: committee report-task force on assessing the need to include higher order terms for small signal analysis. *IEEE Trans. Power Syst.*, Vol. 20(4):1886–1904, May 2005.

[21] N. Pariz, H. M. Shanechi, and E. Vaahedi. Explaining and validating stressed power systems behavior using modal series. *IEEE Trans. Power Syst.*, Vol. 18(2):778–785, May 2003.

[22] M. Bronzini, S. Bruno, M. D. Benedictis, and M. L. Scala. Power system modal identification via wavelet analysis. In *Proceedings of IEEE PowerTech*, Lausanne, Switzerland, July 2007.

[23] D. J. Trudnowski, J. R. Smith, T. A. Short, and D. A. Pierre. An application of Prony methods in PSS design for multimachine systems. *IEEE Trans. Power Syst.*, Vol. 6(1):118–126, February 1991.

[24] A. R. Messina and V. Vittal. Nonlinear, non-stationary analysis of interarea oscillations via Hilbert spectral analysis. *IEEE Trans. Power Syst.*, Vol. 21(3):1234–1241, August 2006.

[25] J. R. Hauer, C. J. Demeure, and L. L. Scharf. Initial results in Prony analysis of power system response signals. *IEEE Trans. Power Syst.*, Vol. 14(1):226–231, February 1999.

[26] D. J. Trudnowski, J. M Johnson, and J. F. Hauer. Making Prony analysis more accurate using multiple signals. *IEEE Trans. Power Syst.*, Vol. 14(1):226–231, February 1999.

[27] D. H. Li and Y. J. Cao. An online identification method for power system low-frequency oscillation based on fuzzy filtering and Prony

algorithm. *Automation of electric power systems*, Vol. 31(1):14–20, January 2007.

[28] C. W. Therrien and C. H. Velasco. An iterative Prony method for ARMA signal modeling. *IEEE Trans. Signal Processing*, Vol. 43(1):358–361, January 1995.

[29] The IEEE special stability controls working group, the dynamic performance, and modeling of DC systems joint working group. HVDC controls for system dynamic performance. *IEEE Trans. Power Syst.*, Vol. 6(2):743–752, May 1991.

[30] H. Breulmann, E. Grebe, M. Lösing, W. Winter, and et al. Analysis and damping of inter-area oscillations in the UCTE/CENTREL power system. In *Proceedings of CIGRÈ*, Paris, France, August 2000.

[31] B. Pal and B. Chaudhuri. *Robust Control in Power Systems*. Springer, 2005.

[32] J. H. Eto. Final report on the August 14, 2003 blackout in the United States and Canada: causes and recommendations. Technical report, US-Canada power system outage task force, 2004.

[33] Final report: system disturbance on 4 November, 2006. Technical report, Union for the co-ordination of transmission of electricity, 2007.

[34] J. Turunen, J. Thambirajah, M. Larsson, B. C. Pal, and et al. Comparison of three electromechanical oscillation damping estimation methods. *IEEE Trans. Power Syst.*, Vol. 26(4):2398–2407, November 2011.

[35] A. A. Hashmani and I. Erlich. Mode selective damping of power system electromechanical oscillations using supplementary remote signals. *IET Gener. Transm. Distrib.*, Vol. 4(10):1127–1138, October 2010.

[36] R. Preece, J. V. Milanović, A. M. Almutairi, and O. Marjanović. Damping of inter-area oscillations in mixed AC/DC networks using WAMS based supplementary controller. *IEEE Trans. Power Syst.*, Vol. 28(2):1160–1169, May 2013.

[37] R. Majumder, B. Chaudhuri, and B. C. Pal. A probabilistic approach to model-based adaptive control for damping of interarea oscillations. *IEEE Trans. Power Syst.*, Vol. 20(1):367–374, February 2005.

[38] A. P. S. Meliopoulos, G. J. Cokkinides, F. Galvan, and P. Myrda. Delivering accurate and timely data to all: model-based substation automation applications for advanced data availability. *IEEE Power Energy Mag.*, Vol. 5(3):74–86, May/June 2007.

[39] I. Kamwa, R. Grondin, and Y. Hebert. Wide-area measurement based stabilizing control of large power systems— a decentralized/hierarchical approach. *IEEE Trans. Power Syst.*, Vol. 16(1):136–153, February 2001.

[40] M. Abido. Pole placement technique for PSS and TCSC-based stabilizer design using simulated annealing. *Int. J. Elec. Power.*, Vol. 19(1):543–554, February 2000.

[41] T. Boehme, G. P. Harrison, and A. R. Wallace. Assessment of distribution network limits for non-firm connection of renewable generation. *IET Renew. Power Gen.*, Vol. 4(1):66–74, May 2010.

[42] D. Z. Fang, X. D. Yang, T. S. Chung, and K. P. Wong. Adaptive fuzzy-logic SVC damping controller using strategy of oscillation energy descent. *IEEE Trans. Power Syst.*, Vol. 19(3):1414–1421, August 2004.

[43] X. M. Mao, Y. Zhang, L. Guan, and X. C. Wu. Coordinated control of interarea oscillation in the China southern power grid. *IEEE Trans. Power Syst.*, Vol. 21(2):845–852, May 2006.

[44] Y. Li, C. Rehtanz, S. Ruberg, L. F. Luo, and et al. Wide-area robust coordination approach of HVDC and FACTS controllers for damping multiple interarea oscillations. *IEEE Trans. Power Del.*, Vol. 27(3):1096–1105, July 2012.

[45] A. G. Phadke and J. S. Thorp. *Synchronized phasor measurements and their application.* Power electronics and power systems series. Springer, 2008.

[46] A. Chakrabortty. Wide-area damping control of power systems using dynamic clustering and TCSC-based redesigns. *IEEE Trans. Smart Grid.*, Vol. 3(3):1503–1514, September 2012.

[47] M. Zima, M. Larsson, P. korba, C. Rehtanz, and et al. Design aspects for wide-area monitoring and control systems. *Proceedings of the IEEE.*, Vol. 93(5):980–996, May 2005.

[48] P. Korba and K. Uhlen. Wide-area monitoring of electromechanical oscillations in the Nordic power system: practical experience. *IET Gener. Transm. Distrib.*, Vol. 4(10):1116–1126, May 2010.

[49] J. Maciejowski. *Predictive control with constraints*. Prentice Hall, Harlow, England, 2002.

[50] R. Sutton and A. Barto. *Reinforcement Learning: an introduction*. MIT Press, Cambridge, Massachusetts, USA, 1998.

[51] L. Busoniu, R. Babuska, B. D. Schutter, and D. Ernst. *Reinforcement learning and dynamic programming using function approximators*. Automation and Control Engineering Series. CRC Press, 2010.

[52] D. Ernst, M. Glavic, and L. Wehenkel. Power system stability control: reinforcement learning framework. *IEEE Trans. Power. Syst.*, Vol. 19(1):427 – 435, February 2004.

[53] M. Glavic. Design of a resistive brake controller for power system stability enhancement using reinforcement learning. *IEEE Trans. Control Syst. Technol.*, Vol. 13(5):743 – 751, September 2005.

[54] R. Hadidi and B. Jeyasurya. Near optimal control policy for controlling power system stabilizers using reinforcement learning. In *Proceedings of PES general meeting*, Calgary, Alberta, Canada, July 2009.

[55] A. Karimi, S. Eftekarnejad, and A. Feliachi. Reinforcement learning based backstepping control of power oscillations. *Electr. Pow. Syst. Res.*, Vol 79(11):1511–1520, November 2009.

[56] R. Hadidi and B. Jeyasurya. Reinforcement learning based real-time wide-area stabilizing control agents to enhance power system stability. *IEEE Trans. Smart Grid.*, Vol. 4(1):489 – 497, March 2013.

[57] D. Ernst, P. Geurts, and L. Wehenkel. Tree-based batch mode reinforcement learning. *J. Mach. Learn. Res.*, Vol 6:503–556, 2005.

[58] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomised trees. *Mach. Learn.*, Vol 63(2):3–42, April 2006.

[59] D. Ernst, M. Glavic, F. Capitanescu, and L. Wehenkel. Reinforcement learning versus model predictive control: a comparison on a power system problem. *IEEE Trans. Syst., Man, Cybern.*, Vol. 39(2):517–529, April 2009.

[60] D. Wang, M. Glavic, and L. Wehenkel. A new MPC scheme for damping wide-area electromechanical oscillations in power systems. In *Proceedings of the 2011 IEEE PES PowerTech*, Trondheim, Norway, June 2011.

[61] D. Wang, M. Glavic, and L. Wehenkel. Distributed MPC of wide-area electromechanical oscillations of large-scale power systems. In *Proceedings of the 16th Intelligent System Applications to Power Systems (ISAP)*, Crete, Greece, September 2011.

[62] D. Wang, M. Glavic, and L. Wehenkel. Considerations of model predictive control for electromechanical oscillations damping in large-scale power systems. *International Journal of Electric Power and Energy Systems*, Accepted for publication, 2014.

[63] D. Wang, M. Glavic, and L. Wehenkel. Trajectory-based supplementary damping control for power system electromechanical oscillations. *IEEE Trans. Power Syst.*, Under second review.

[64] S. J. Qin and T. A. Badgwell. A survey of industrial model predictive control technology. *Control Eng. Pract.*, Vol. 11(7):733–764, July 2003.

[65] R. Scattolini. Architectures for distributed and hierarchical model predictive control-a review. *Journal of Process Control*, Vol. 19(5):723–731, May 2009.

[66] F. Delbos and J. C. Gilbert. Global linear convergence of an augmented Lagrangian algorithm for solving convex quadratic optimization problems. *J. Convex Anal.*, Vol. 12(1):45–69, April 2005.

[67] Mathworks Inc. *Large-Sclae quadprog algorithm.* Available online: http://www.mathworks.nl/help/optim/ug/quadprog.html, 2009.

[68] A. Bemporad and M. Morari. Robust model predictive control: a survey. *Robustness in Identification and Control*, Vol. 245:207–226, 1999.

[69] L. Acar. Some examples for the decentralized receding horizon control. In *Proceedings of the 31st IEEE conference on Decision and Control*, Tucson, Arizona, USA, December 1992.

[70] D. D. Siljak. Decentralized control and computations: status and prospects. *Annu. Rev. Control*, Vol. 20:131–141, 1996.

[71] A. Monticelli, M. V. F. Pereira, and S. Granville. Security-constrained optimal power flow with post-contingency corrective rescheduling. *IEEE Trans. Power Syst.*, Vol. 2(1):175–180, February 1987.

[72] F. Capitanescu, M. Glavic, D. Ernst, and L. Wehenkel. Contingency filtering techniques for preventive security-constrained optimal power flow. *IEEE Trans. Power Syst.*, Vol. 22(4):1690 –1697, November 2007.

[73] F. Capitanescu and L. Wehenkel. A new iterative approach to the corrective security-constrained optimal power flow problem. *IEEE Trans. Power Syst.*, Vol. 23(4):1533 –1541, November 2008.

[74] F. Capitanescu, J. L. Martinez Ramos, P. Panciatici, D. Kirschen, and et al. State-of-the-art, challenges, and future trends in security constrained optimal power flow. *Electr. Pow. Syst. Res.*, Vol. 81(8):1731 –1741, August 2011.

[75] B. Marinescu and H. Bourles. Robust predictive control for the flexible coordinated secondary voltage control of large-scale power systems. *IEEE Trans. Power Syst.*, Vol. 14(4):1262–1268, November 1999.

[76] A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright. Distributed MPC strategies with application to power system automatic generation control. *IEEE Trans. Control Syst. Technol.*, Vol. 16(6):1192–1206, November 2008.

[77] S. R. Yamchi, M. Cychowski, R. R. Negenborn, B. D. Schutter, and et al. Kalman filter-based distributed predictive control of large-scale multi-rate systems: application to power networks. *IEEE Trans. Control Syst. Technol.*, Vol. 21(1):27–39, January 2013.

[78] B. Otomega, A. Marinakis, M. Glavic, and T. V. Cutsem. Model predictive control to alleviate thermal over loads. *IEEE Trans. Power Syst.*, Vol. 22(3):1384–1385, August 2007.

[79] J. S. A. Carneiro and L. Ferrarini. Preventing thermal overloads in transmission circuits via model predictive control. *IEEE Trans. Control Syst. Technol.*, Vol. 18(6):1406–1412, November 2010.

[80] M. Larsson and D. Karlsson. Coordinated system protection scheme against voltage collapse using heuristic search and predictive control. *IEEE Trans. Power Syst.*, Vol. 18(3):1001–1006, August 2003.

[81] J. Y. Wen, Q. H. Wu, S. J. Cheng, and J. Fitch. Optimal coordinated voltage control for power system voltage stability. *IEEE Trans. Power Syst.*, Vol. 19(2):1115–1122, May 2004.

[82] I. A. Hiskens and B. Gong. Voltage stability enhancement via model predictive control of load. *Intell. Autom. Soft. Co.*, Vol. 12(1):117–124, January 2006.

[83] R. R. Negenborn and B. S. Rodsai. *Multi-agent model predictive control with applications to power networks*. PhD thesis, Delft university of Technology, 2007.

[84] L. C. Jin, R. Kumar, and N. Elia. Model predictive control-based real-time power system protection schemes. *IEEE Trans. Power Syst.*, Vol. 25(2):988–998, May 2010.

[85] M. Glavic, M. Hajian, W. Rosehart, and T. V Cutsem. Receding - horizon multi-step optimization to correct nonviable or unstable transmission voltages. *IEEE Trans. Power Syst.*, Vol. 26(3):1641–1650, August 2011.

[86] J. J. Ford, G. Ledwich, and Z. Y. Dong. Efficient and robust model predictive control for first swing transient stability of power systems using flexible AC transmission systems devices. *IET Gener. Transm. Distrib.*, Vol. 2(5):73–742, September 2008.

[87] S. R. Wagh, A. K. Kamath, and N. M. Singh. Non-linear model predictive control for improving transient stability of power system using

TCSC controller. In *Proceedings of the 7th Asian control conference*, Hong Kong, China, August 2009.

[88] T. A. Short, D. A. Pierre, and J. R. Smith. Self-tuning generalized predictive control for switched capacitor damping of power system oscillations. In *Proceedings of the 28th conference on decision and control*, Florida, USA, December 1989.

[89] N. P. Johansson, H. P. Nee, and L. Angquist. An adaptive model predictive approach to power oscillation damping utilizing variable series reactance FACTS devices. In *Proceedings of the 41th Int. universities power engineering conference*, Newcastle, UK, September 2006.

[90] L. Wang, H. Cheung, A. Hamlyn, and R. Cheung. Model prediction adaptive control of inter-area oscillations in multi-generators power systems. In *Proceedings of the IEEE PES general meeting*, Calgary, Canada, July 2009.

[91] B. Chaudhuri, R. Majumder, and B. C. Pal. Application of multiple-model adaptive control strategy for robust damping of interarea oscillations in power system. *IEEE Trans. Control Syst. Technol.*, Vol. 12(5):727–736, September 2004.

[92] J. H. Chow and K. W. Cheung. A toolbox for power system dynamics and control engineering education and research. *IEEE Trans. Power Syst.*, Vol. 7(4):1559–1564, November 1992.

[93] A. P. S. Meliopoulos, G. J. Cokkinides, F. Galvan, and B. Fardanesh. Distributed state estimator-advances and demonstration. In *Proceedings of 41st Hawaii Int. Conf. System Sciences*, Waikoloa, USA, January 2008.

[94] H. Wu, K. S. Tsakalis, and G. T. Heydt. Evaluation of time delay effects to wide-area power system stabilizer design. *IEEE Trans. Power Syst.*, Vol. 19(4):1935–1941, November 2004.

[95] D. D. Siljak and A. I. Zecevic. Control of large-scale systems: beyond decentralized feedback. *IEEE Trans. Power Syst.*, Vol. 20(1):367–374, February 2005.

[96] S. Talukdar, D. Jia, P. Hines, and B. H. Krogh. Distributed model predictive control for the mitigation of cascading failures. In *Proceedings of the 44th IEEE conference on decision and control*, Seville, Spain, December 2005.

[97] P. Hines, D. Jia, and S. Talukdar. Distributed model predictive control for electric grids. In *Proceedings of the Carnegie Mellon transmission conference*, Pittsburgh, USA, December 2004.

[98] P. Li, B. H. Zhang, L. Y. Cheng, Z. Q. Bo, and et al. Study on the coordinated voltage control with the distributed model prediction. In *Proceedings of the IEEE PES general meeting 2010*, Minneapolis, USA, July 2010.

[99] A. G. Beccuti and M. Morai. A distributed solution approach to centralized emergency voltage control. In *Proceedings of the 2006 American control conference*, Minneapolis, USA, July 2006.

[100] N. Motee and B. S. Rodsai. Optimal partitioning in distributed model predictive control. In *Proceedings of the 2003 American control conference*, Colorado, USA, June 2003.

[101] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar. Distributed model predictive control. *IEEE Control Syst. Mag.*, Vol. 22(1):44–52, February 2002.

[102] S. Corsi, P. Marannino, N. Losignore, G. Moreschini, and et al. Coordination between the reactive power scheduling function and the hierarchical voltage control of the EHV ENEL system. *IEEE Trans. Power Syst.*, Vol. 10(2):686–694, May 1995.

[103] A. Rubaai and J. A. Momoh. Three-level control scheme for load - frequency control of interconnected power system. In *Proceedings of TENCON '91*, August 1991.

[104] M. Rahmani and N. Sadati. Hierarchical optimal robust load-frequency control for power systems. *IET Gener. Transm. Distrib.*, Vol. 6(4):303–312, April 2012.

[105] F. Okou, L. A. Dessaint, and O. Akhrif. Power systems stability enhancement using a wide-area signals based hierarchical controller. *IEEE Trans. Power Syst.*, Vol. 20(3):1465–1477, August 2005.

[106] M. H. Raouf, E. R. Anarmarzi, Y. H. Lesani, and J. Olamaei. Power system damping using hierarchical fuzzy multi-input power system stabilizer and static var compensator. *International journal of electrical and electronics engineering*, Vol. 5(2):117–123, May 2011.

[107] A. Rubaai and F. E. Villaseca. Transient stability hierarchical control in multimachine power systems. *IEEE Trans. Power Syst.*, Vol. 4(4):1438–1442, November 1989.

[108] R. Nadira. A hierarchical interactive approach to electric power system restoration. *IEEE Trans. Power Syst.*, Vol. 7(3):1123–1131, August 1992.

[109] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach.* Prentice hall, New Jersey 07458, USA, 2003.

[110] M. Rashidi and F. Rashidi. Damping enhancement in the presence of load parameters uncertainty using reinforcement learning based SVC controller. In *Proceedings of IEEE international conference on systems, man and cybernetics.*, October 2003.

[111] P. Stone and M. Veloso. Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots*, 18(3):345–383, July 2000.

[112] J. Chow and G. Rogers. *Power system toolbox version 3.0.* Available online: http://www.ecse.rpi.edu/pst/PST.html/, 2008.