DISTRIBUTED MPC STRATEGIES FOR AUTOMATIC GENERATION CONTROL

Aswin N. Venkat * **Ian A. Hiskens** **,[1]
James B. Rawlings * **Stephen J. Wright** ***

* *Department of Chemical and Biological Engineering, University of Wisconsin, Madison, WI-53706*
** *Department of Electrical and Computer Engineering, University of Wisconsin, Madison, WI-53706*
*** *Computer Sciences Department, University of Wisconsin, Madison, WI-53706*

Abstract: The paper considers distributed model predictive control (MPC) strategies that are appropriate for controlling large-scale systems such as power systems. The overall system is decomposed into subsystems, each with its own MPC controller. To achieve performance equivalent to centralized MPC, these distributed regulators must work iteratively and cooperatively towards satisfying a common, systemwide control objective. Automatic generator control (AGC) provides a practical example for contrasting the performance of centralized and decentralized controllers.

Keywords: Distributed model predictive control; automatic generation control; power system control.

1. INTRODUCTION

Model predictive control (MPC) is emerging as a sophisticated, yet practical, control technology. This model-based control strategy uses a prediction of system behaviour to establish an appropriate control response. A number of benefits follow from using MPC, including the ability to account systematically for process constraints. The effectiveness of MPC depends on models of appropriate accuracy and on the availability of sufficiently fast computational resources—requirements that limit the application base for MPC. Even so, applications abound in the process industries, and are becoming more widespread (Qin and Badgwell, 2003; Camacho and Bordons, 2004).

Traditionally, control of large, networked systems is achieved by designing local, subsystem-based controllers that ignore the interactions between the different subsystems. These controllers often perform poorly when the subsystem interactions are significant. Centralized MPC, on the other hand, is impractical for control of large-scale, geographically expansive systems, such as power systems. A distributed MPC framework is appealing in this context, but must be designed to take account of interactions between subsystems. Interaction issues are crucial to the success of distributed MPC, and are discussed further in Section 3.

Automatic generation control (AGC) provides a topical example for illustrating the performance of distributed MPC in a power system setting. The purpose of AGC is to regulate the real power output of generators, with the aim of controlling system frequency and tie-line interchange (Wood and Wollenberg, 1996). AGC must account for various limits, including restrictions on the amount and rate of generator power deviations.

Flexible AC transmission system (FACTS) devices allow control of the real power flow over selected paths through a transmission network

[1] Corresponding author. Email: `hiskens@engr.wisc.edu`

(Hingorani and Gyugyi, 2000). As transmission systems become more heavily loaded, such controllability offers economic benefits (Krogh and Kokotovic, 1984). However FACTS controls must be coordinated with each other, and with AGC. Distributed MPC offers an effective means of achieving such coordination, whilst alleviating the organizational and computational burden associated with centralized control.

2. MODELS

Distributed MPC relies on decomposing the overall system model into appropriate subsystem models. A system comprised of M interconnected subsystems will be used to establish these concepts.

2.1 Centralized model

The overall system model is represented as a discrete, linear time-invariant (LTI) model of the form

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k), \end{aligned} \tag{1}$$

in which k denotes discrete time and

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1M} \\ \vdots & \vdots & \ddots & \vdots \\ A_{i1} & A_{i2} & \dots & A_{iM} \\ \vdots & \vdots & \ddots & \vdots \\ A_{M1} & A_{M2} & \dots & A_{MM} \end{bmatrix} \quad B = \begin{bmatrix} B_{11} & B_{12} & \dots & B_{1M} \\ \vdots & \vdots & \ddots & \vdots \\ B_{i1} & B_{i2} & \dots & B_{iM} \\ \vdots & \vdots & \ddots & \vdots \\ B_{M1} & B_{M2} & \dots & B_{MM} \end{bmatrix}$$

$$C = \begin{bmatrix} C_{11} & 0 & \dots & 0 \\ 0 & C_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & C_{MM} \end{bmatrix} \quad u = \begin{bmatrix} u_1{}' & u_2{}' & \dots & u_M{}' \end{bmatrix}'$$

$$x = \begin{bmatrix} x_1{}' & x_2{}' & \dots & x_M{}' \end{bmatrix}' \quad y = \begin{bmatrix} y_1{}' & y_2{}' & \dots & y_M{}' \end{bmatrix}'.$$

For each subsystem $i = 1, \dots, M$, the triplet (u_i, x_i, y_i) represents the subsystem input, state and output vector respectively.

2.2 Decentralized model

In the decentralized modeling framework, the effect of the external subsystems on the local subsystem is assumed to be negligible. The decentralized model for subsystem i, $i = 1, \dots, M$ is written as

$$\begin{aligned} x_i(k+1) &= A_{ii}x_i(k) + B_{ii}u_i(k) \\ y_i(k) &= C_{ii}x_i(k). \end{aligned} \tag{2}$$

2.3 Partitioned model (PM)

The PM for each subsystem i incorporates the effect of the local subsystem variables as well as the effect of the states and inputs of the interconnected subsystems. The PM for subsystem i, as the name suggests, is obtained by considering an appropriate partition of (1), as follows:

$$\begin{aligned} x_i(k+1) &= A_{ii}x_i(k) + B_{ii}u_i(k) + \sum_{j \neq i}(A_{ij}x_j(k) + B_{ij}u_j(k)) \\ y_i(k) &= C_{ii}x_i(k). \end{aligned} \tag{3}$$

3. DISTRIBUTED MPC FOR POWER SYSTEM CONTROL

3.1 Preliminaries

Given the PM for each subsystem $i = 1, \dots, M$, we consider two formulations for distributed MPC: communication-based MPC and cooperation-based MPC. The suitability of either distributed MPC framework for systemwide control is assessed in the sequel. In both approaches, an optimization and exchange of variables between subsystems is performed during a sample time. We may choose not to iterate to convergence.

The set of admissible controls for subsystem $i, \Omega_i \subseteq \mathbb{R}^{m_i}$ is assumed to be a non-empty, compact, convex set containing the origin in its interior. The set of admissible controls for the whole plant Ω is defined to be the Cartesian product of the admissible control sets of each of the subsystems. For subsystem i at time k, the predicted state vector at time $t > k$ is denoted by $x_i(t|k)$. We have $x_i(k|k) \equiv x_i(k)$.

The *cost function* for subsystem i is defined over an infinite horizon as follows:

$$\phi_i(\boldsymbol{x}_i, \boldsymbol{u}_i; x_i(k)) = \sum_{t=k}^{\infty} x_i(t|k)'Q_i(t)x_i(t|k) + u_i(t|k)'R_i(t)u_i(t|k), \tag{4}$$

where $Q_i > 0$, $R_i > 0$ are symmetric weighting matrices and $\boldsymbol{x}_i(k) = [x_i(k+1|k)', x_i(k+2|k)', \dots \dots]'$, $\boldsymbol{u}_i(k) = [u_i(k|k)', u_i(k+1|k)', \dots \dots]'$ denote the infinite horizon state and input trajectories, respectively, for subsystem i.

3.2 Communication-based MPC [2]

For communication-based MPC, the optimal state-input trajectory $(\boldsymbol{x}_i^p(k), \boldsymbol{u}_i^p(k))$ for subsystem i at iteration p is obtained as the solution to the following optimization problem

$$\begin{aligned} \min_{\boldsymbol{x}_i, \boldsymbol{u}_i} \quad & \phi_i(\boldsymbol{x}_i, \boldsymbol{u}_i; x_i(k)) \\ \text{s.t.} \quad & x_i(t+1|k) = A_{ii}x_i(t|k) + B_{ii}u_i(t|k), \quad k \le t \\ & \qquad + \sum_{j \neq i}[A_{ij}x_j^{p-1}(t|k) + B_{ij}u_j^{p-1}(t|k)] \\ & u_i(t|k) \in \Omega_i, \ k \le t \le k+N-1 \\ & u_i(t|k) = 0, \ k+N \le t. \end{aligned} \tag{5}$$

The integer N denotes the control horizon. For notational simplicity, we drop the time dependence of $(\boldsymbol{x}_i^p(k), \boldsymbol{u}_i^p(k))$ and represent it as $(\boldsymbol{x}_i^p, \boldsymbol{u}_i^p)$. For each subsystem i at iteration p, only the subsystem input sequence $\boldsymbol{u}_i^p$ is optimized and updated. The other subsystems' inputs are not altered during the solution of (5); they remain at their values from iteration $p-1$. The objective function is the one for subsystem i only.

In the communication-based MPC framework, each subsystem's MPC has no information about

[2] Similar strategies have been proposed by (Jia and Krogh, 2001; Camponogara *et al.*, 2002)

the objectives of the interconnected subsystems' MPCs. Convergence of the exchanged state and input trajectories must therefore be assumed—a drawback of this formulation. Not uncommonly, this MPC formulation leads to unstable closed-loop behavior, so it is an unreliable strategy for systemwide control.

3.3 *Feasible cooperation-based MPC (FC-MPC)*

To arrive at a reliable, distributed, systemwide MPC framework, we modify the objectives of the subsystems' MPCs to provide a means for cooperative behaviour among the controllers. Each local controller objective ϕ_i is replaced by one that measures the systemwide impact of local control actions. Here, we choose the simplest such measure - a strong convex combination of the individual subsystems' objectives *i.e.*, $\phi = \sum w_i\phi_i$, where w_i are the weights. We set $w_i = 1/M$ for all $i = 1, \ldots, M$ in our discussion below, but our results hold for any combination of weights satisfying $w_i > 0$, $\sum w_i = 1$.

In large-scale implementations, the system sampling interval may be insufficient for the convergence of the iterative, cooperation-based algorithm. In such cases, the algorithm has to be terminated prior to convergence of the state and input trajectories and the last calculated input trajectories used to compute a suitable control law. To facilitate intermediate termination, it is imperative that all iterates generated by the cooperation-based algorithm are systemwide feasible (*i.e.*, satisfy all model and inequality constraints) and the resulting distributed control law is closed-loop stable.

We define the finite horizon state and input trajectories for subsystem i as $\overline{\boldsymbol{x}}_i(k)' = [x_i(k+1|k)', \ldots, x_i(k+N|k)']$ and $\overline{\boldsymbol{u}}_i(k)' = [u_i(k|k)', u_i(k+1|k)', \ldots, u_i(k+N-1|k)']$. For convenience, we drop the k dependence of $\overline{\boldsymbol{x}}_i$ and $\overline{\boldsymbol{u}}_i$ in the following discussion. It is shown in Appendix A that for each $i = 1, \ldots, M$, $\overline{\boldsymbol{x}}_i$ can be expressed as follows:

$$\overline{\boldsymbol{x}}_i = E_{ii}\overline{\boldsymbol{u}}_i + f_{ii}x_i(k) + \sum_{j\neq i}[E_{ij}\overline{\boldsymbol{u}}_j + f_{ij}x_j(k)]. \quad (6)$$

The infinite horizon input trajectory $\boldsymbol{u}_i$ is obtained by augmenting $\overline{\boldsymbol{u}}_i$ with the input sequence $u_i(t|k) = 0$ for all $t \geq k+N$. The infinite horizon state trajectory $\boldsymbol{x}_i$ is derived from $\overline{\boldsymbol{x}}_i$ by propagating the terminal state $x_i(k+N|k)$ using (3) and $u_i(t|k) = 0$, $t \geq k+N$, $\forall\, i = 1, \ldots, M$. For subsystem i, the FC-MPC optimization problem is

$$\boldsymbol{u}_i^{p(*)}(k) \in \arg(\text{FC-MPC}_i) \quad (7a)$$

in which

$$\text{FC-MPC}_i \triangleq$$

$$\min_{\boldsymbol{u}_i} \frac{1}{M}\sum_{r=1}^{M}\Phi_r\left(\boldsymbol{u}_1^{p-1}, \ldots, \boldsymbol{u}_{i-1}^{p-1}, \boldsymbol{u}_i, \boldsymbol{u}_{i+1}^{p-1}, \ldots, \boldsymbol{u}_M^{p-1}; x(k)\right) \quad (7b)$$

$$\text{s.t.} \quad u_i(j|k) \in \Omega_i, \quad k \leq j \leq k+N-1 \quad (7c)$$

$$u_i(j|k) = 0, \quad k+N \leq j, \quad (7d)$$

in which each cost function $\Phi_i(\cdot)$ is obtained by eliminating the state trajectory $\boldsymbol{x}_i$ from (4), using (3). For this case, the FC-MPC optimization problem for each subsystem $i = 1, \ldots, M$ can be explicitly written as the finite horizon optimization

$$\text{FC-MPC}_i \triangleq$$

$$\min_{\overline{\boldsymbol{u}}_i} \frac{1}{2}\overline{\boldsymbol{u}}_i'\mathcal{R}_i\overline{\boldsymbol{u}}_i + \left(\sum_{j=1}^{M}E_{ji}'\sum_{s\neq j}\mathbb{M}_{js}\sum_{l\neq i}\left(E_{sl}\overline{\boldsymbol{u}}_l^{p-1} + \boldsymbol{g}_s(x(k))\right)\right.$$
$$\left. + \sum_{j=1}^{M}E_{ji}'\mathbb{Q}_j\sum_{l\neq i}\left(E_{jl}\overline{\boldsymbol{u}}_l^{p-1} + \boldsymbol{g}_j(x(k))\right)\right)'\overline{\boldsymbol{u}}_i$$
$$\text{s.t.} \quad u_i(j|k) \in \Omega_i, \quad k \leq j \leq k+N-1, \quad (8)$$

in which

$$\mathcal{R}_i = \mathbb{R}_i + \sum_{j=1}^{M}E_{ji}'\mathbb{Q}_jE_{ji} + \sum_{j=1}^{M}E_{ji}'\sum_{s\neq j}\mathbb{M}_{js}E_{si},$$
$$\mathbb{Q}_i = \text{diag}\left(Q_i(1), \ldots, Q_i(N-1), \overline{Q}_{ii}\right),$$
$$\mathbb{M}_{ij} = \text{diag}\left(0, \ldots, 0, \overline{Q}_{ij}\right),$$
$$\mathbb{R}_i = \text{diag}\left(R_i(0), R_i(1), \ldots, R_i(N-1)\right),$$
$$\boldsymbol{g}_i(x(k)) = \sum_{j=1}^{M}f_{ij}x_j(k),$$

while

$$\overline{\mathbb{Q}} = \begin{bmatrix} \overline{Q}_{11} & \overline{Q}_{12} & \cdots & \cdots & \overline{Q}_{1M} \\ \overline{Q}_{21} & \overline{Q}_{22} & \cdots & \cdots & \overline{Q}_{2M} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \overline{Q}_{M1} & \overline{Q}_{M2} & \cdots & \cdots & \overline{Q}_{MM} \end{bmatrix} \quad (9)$$

is a suitable terminal penalty matrix. Restricting attention to open-loop stable systems simplifies the choice of $\overline{\mathbb{Q}}$. For each $i = 1, \ldots, M$, let $Q_i(0) = Q_i(1) = \cdots = Q_i(N-1) = Q_i$. The terminal penalty $\overline{\mathbb{Q}}$ can be obtained as the solution to the centralized Lyapunov equation

$$A'\,\overline{\mathbb{Q}}\,A - \overline{\mathbb{Q}} = -\mathbb{Q} \quad (10)$$

in which $\mathbb{Q} = \text{diag}(Q_1, Q_2, \ldots, Q_M)$.

3.4 *FC-MPC algorithm and properties*

The state trajectory for subsystem i generated by the set of subsystem input trajectories $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_M$ and initial state z is represented as $\boldsymbol{x}_i(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_M; z)$. For notational simplicity, we drop the functional dependence of the state trajectory and write $\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_M; z)$. The following algorithm is employed for cooperation-based distributed MPC.

Algorithm 1.
Given $\left(\boldsymbol{u}_i^0, x_i(k)\right), \mathbb{Q}_i \geq 0, \mathbb{R}_i \geq 0, i = 1, \ldots, M$
$p_{\max}(k) \geq 0$ and $\epsilon > 0$
$p \leftarrow 1, \rho_i \leftarrow \Gamma\epsilon, \Gamma \gg 1$
while $\rho_i > \epsilon$ for some $i = 1, \ldots, M$ and $p \leq p_{\max}$
for $i = 1, \ldots, M$
$\boldsymbol{u}_i^{p(*)} \in \arg(\text{FC-MPC}_i)$, (see (7), (8))
$\boldsymbol{u}_i^p = \frac{1}{M}\boldsymbol{u}_i^{p(*)} + \left(1 - \frac{1}{M}\right)\boldsymbol{u}_i^{p-1}$
$\rho_i = \|\boldsymbol{u}_i^p - \boldsymbol{u}_i^{p-1}\|$

end (for)
Transmit $\boldsymbol{u}_i^p$, $i = 1, \ldots, M$ to interconnected subsystems.
Calculate $\boldsymbol{x}_i(\boldsymbol{u}_1^p, \ldots, \boldsymbol{u}_M^p; x(k))$, $i = 1, \ldots, M$.
$p \leftarrow p + 1$
end (while)

After p iterates, denote the cooperation-based cost function by $\Phi(\boldsymbol{u}_1^p, \ldots, \boldsymbol{u}_M^p; x(k))$. The following properties can be established for the FC-MPC formulation (8) employing Algorithm 1.

Lemma 1. Given the distributed MPC formulation FC-MPC$_i$ defined in (7), (8), $\forall\ i = 1, \ldots, M$, the sequence of cost functions $\{\Phi(\boldsymbol{u}_1^p, \ldots, \boldsymbol{u}_M^p; x(k))\}$ generated by Algorithm 1 is a non-increasing function of the iteration number p.

Using Lemma 1 and the fact that $\Phi(\cdot)$ is bounded below assures convergence with iteration number p.

Lemma 2. All limit points of Algorithm 1 are optimal.

Lemma 2 implies that the solution obtained at convergence of Algorithm 1 is Pareto optimal *i.e.*, the solution at convergence is identical to the centralized MPC solution.

3.5 Distributed MPC control law

Let $\mathcal{X}$ represent the constrained stabilizable set for the system under the set of input constraints $\Omega_1 \times \Omega_2 \times \ldots \times \Omega_M$. At time k, let the FC-MPC algorithm (Algorithm 1) be terminated after $p(k) = l$ iterates, with

$$\boldsymbol{u}_i^l(x(k)) = \left[u_i^l(k; x(k))', u_i^l(k+1; x(k))', \ldots, \right]', \quad i = 1, \ldots, M \tag{11}$$

representing the solution to Algorithm 1 after l cooperation-based iterates. The distributed MPC control law is obtained through a receding horizon implementation of optimal control whereby the input applied to subsystem i at time k, $u_i(k)$, is

$$u_i(k) = u_i^l(k; x(k)). \tag{12}$$

Lemmas 1 and 2 lead to the following theorem on closed-loop stability of the nominal system.

Theorem 3. Let Algorithm 1, the distributed MPC formulation (7), (8) with $N \geq 1$, and the distributed control law defined in (12) be given. If A is stable, $\overline{\mathbb{Q}}$ is obtained from (10), and

$$Q_i(0) = Q_i(1) = \ldots = Q_i(N-1) = Q_i > 0$$
$$R_i(0) = R_i(1) = \cdots = R_i(N-1) = R_i > 0$$
$$i = 1, \ldots, M$$

then the origin is an exponentially stable equilibrium for the closed-loop system

$$x(k+1) = Ax(k) + Bu(x(k))$$

in which

$$u(x(k)) = \left[u_1^{p(k)}(k; x(k))', \ldots, u_M^{p(k)}(k; x(k))'\right]'$$

for all $x(k) \in \mathcal{X}$ and any $p(k) = 1, 2, \ldots$

Remark 4. If $(A, \mathbb{Q}^{\frac{1}{2}})$ is detectable and $Q_i \geq 0$ for all $i = 1, \ldots, M$, then the closed-loop system is asymptotically stable under the distributed MPC control law.

4. EXAMPLES

4.1 Performance comparison

The examples use the cumulative stage cost as an index for comparing the performance of different controller paradigms. Accordingly, define

$$\Lambda = \frac{1}{t} \sum_{k=0}^{t-1} \sum_{i=1}^{M} \frac{1}{2} \left[x_i(k)' Q_i x_i(k) + u_i(k)' R_i u_i(k)\right]. \tag{13}$$

4.2 Two area power system network

We consider an example with two control areas interconnected through a tie line. For a 25% load increase [3] in area 2, the load disturbance rejection performance of the FC-MPC formulation is evaluated and compared against the performance of centralized MPC (cent-MPC), decentralized MPC (decent-MPC), communication-based MPC (comm-MPC) and the standard automatic generation control (AGC) with anti-reset windup. The load reference setpoint in each area is constrained between ± 0.3.

The relative performance of standard AGC, cent-MPC and FC-MPC (terminated after just 1 cooperation-based iteration) is depicted in Fig. 1, where the transient responses of the tie-line power flow and the area 2 load reference setpoint are shown. Under standard AGC, the system takes more than 400 sec to drive the tie-line power flow deviation to zero. With cent-MPC or FC-MPC (terminated after just 1 iteration), the tie-line power flow disturbance is rejected in less than 100 sec. The closed-loop performances of the various control formulations are compared in Table 1.

Table 1. Performance of different control formulations w.r.t. cent-MPC, $\Delta\Lambda\% = \frac{\Lambda_{\text{config}} - \Lambda_{\text{cent}}}{\Lambda_{\text{cent}}} \times 100$.

	Λ	$\Delta\Lambda\%$
standard AGC	39.26	158.32
decent-MPC	17.683	16.4
comm-MPC	17.42	14.62
FC-MPC (1 iterate)	15.24	0.24
FC-MPC (5 iterates)	~ 15.2	~ 0
cent-MPC	15.2	–

[3] In practice, such a large load change would result in curtailment of AGC, and activation of other, more drastic controls such as load shedding. This exaggerated disturbance is useful, however, for exploring the influence of constraints on the various control strategies.

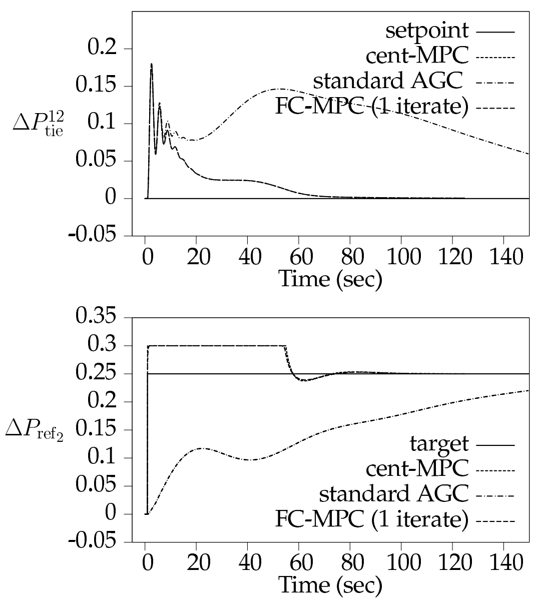

Fig. 1. Change in tie line power flow ($\Delta P_{\mathrm{tie}}^{12}$) and load reference setpoint ($\Delta P_{\mathrm{ref}_2}$).

4.3 *Four area power system network*

An example with four control areas is shown in Fig. 2. Power flows through tie-line connections $1-2$, $2-3$, and $3-4$ are the sources of interactions between the control areas. The relative performance of cent-MPC, comm-MPC and FC-MPC is analyzed for a 25% load increase in area 2 and a simultaneous 25% load drop in area 3. In the comm-MPC and FC-MPC formulations, the load reference setpoint (P_{ref}) in each area is manipulated independently to reject the load disturbances and drive the deviations in frequencies ($\Delta\omega_i$) and tie-line power flows ($\Delta P_{\mathrm{tie}}^{ij}$) to zero. In the cent-MPC framework, a single MPC controls the entire power network. The load reference setpoint for each area is constrained between ± 1.

Fig. 3 shows the performance of cent-MPC, comm-MPC and FC-MPC (terminated after 1 cooperation-based iterate.) Based on calculated closed-loop control costs, the performance of comm-MPC is worse than that of cent-MPC by about 25%. The closed-loop performance of the FC-MPC formulation, terminated after just 1 cooperation-based iterate, is within 3.2% of cent-MPC performance. Performance of the FC-MPC framework can be driven to within any pre-specified tolerance of cent-MPC performance by allowing the cooperation-based iterative process to converge.

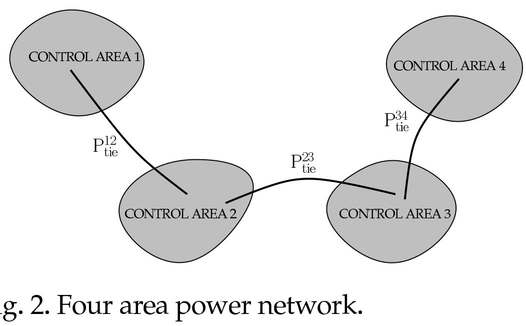

Fig. 2. Four area power network.

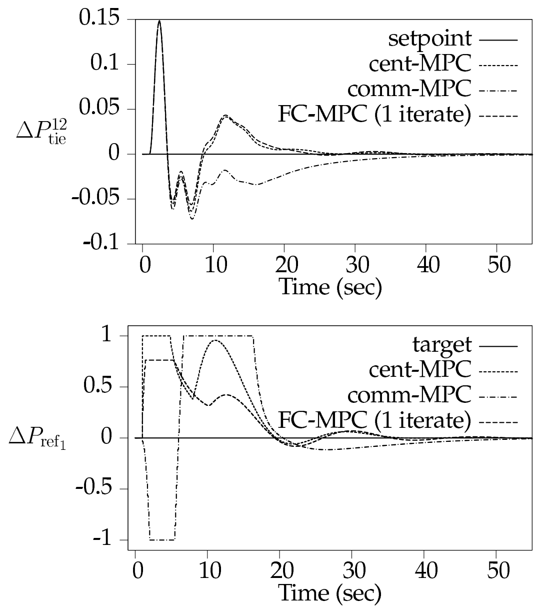

Fig. 3. Change in tie line flow ($\Delta P_{\mathrm{tie}}^{12}$), and load reference setpoint ($\Delta P_{\mathrm{ref}_1}$).

4.4 *Two area power system with FACTS device*

This example returns to the two area network. In this case, the interconnection between the two areas incorporates a FACTS device that is employed by area 1 to manipulate the effective impedance of the tie line. We investigate the relative performance of the cent-MPC, comm-MPC, and FC-MPC formulations, in response to a 25% increase in the load of area 2.

Under the comm-MPC formulation, the system takes about 300 sec to reject the load disturbance. The comm-MPC formulation incurs a performance loss of 192.51% relative to cent-MPC. Under the FC-MPC formulation, terminated after 1 iterate, the performance loss is only 6.2% compared to cent-MPC. The system rejects the load disturbance in less than half the time required by comm-MPC. Fig. 4 shows the relative phase deviation in the two areas, and the change in impedance due to the FACTS device, for the different MPC frameworks.

5. CONCLUSIONS

Centralized MPC is not well suited for control of large-scale, geographically expansive systems such as power systems. However, the performance benefits obtained with centralized MPC can be realized through distributed MPC strategies. Such strategies rely on decomposition of the overall system into interconnected subsystems, and iterative exchange of information between these subsystems. An MPC optimization problem is solved within each subsystem, using local measurements and the latest available external information.

Various forms of distributed MPC have been defined. Feasible cooperation-based MPC (FC-MPC) assigns a common, system-wide objective

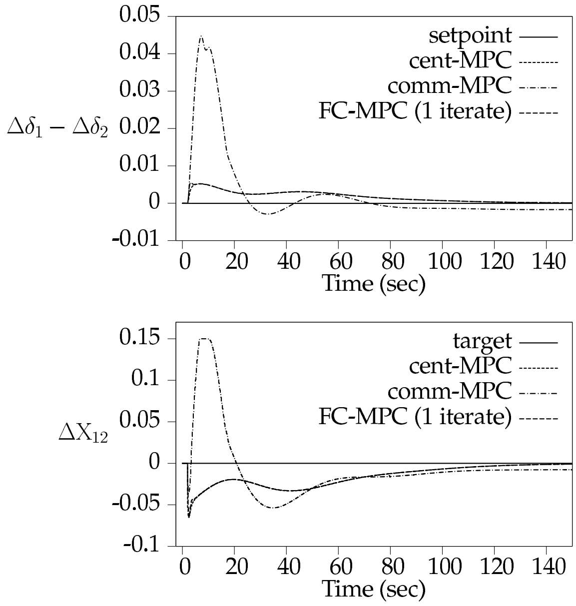

Fig. 4. Relative phase difference $(\Delta\delta_1 - \Delta\delta_2)$, and change in FACTS impedance (ΔX_{12}).

to all subsystem problems, and has the property that the converged solution is identical to centralized MPC. In addition, the FC-MPC algorithm can be terminated prior to convergence without compromising feasibility or closed-loop stability of the resulting distributed controller. This feature allows the practitioner to terminate the algorithm at the end of the sampling interval, even if convergence is not achieved.

The paper has presented a number of power system examples that have applied distributed MPC to automatic generation control (AGC). MPC outperforms standard AGC, due to its ability to account for process constraints. FC-MPC achieves performance that is equivalent to centralized MPC, and superior to other forms of distributed MPC. The FC-MPC framework also allows coordination of FACTS controls with AGC. In this case, the cooperative aspect of FC-MPC was very important for achieving acceptable response.

6. ACKNOWLEDGMENT

The authors gratefully acknowledge the financial support of the industrial members of the Texas-Wisconsin Modeling and Control Consortium, and NSF through grant #CTS-0456694.

Appendix A. MODEL MANIPULATION

To simplify the development of the FC-MPC algorithm, it is convenient to eliminate the states $\boldsymbol{x}_i$, $i = 1, \ldots, M$ using the PM (3). Propagating the model for each subsystem through the control horizon N gives

$$\overline{\boldsymbol{x}}_i = \overline{E}_{ii}\overline{\boldsymbol{u}}_i + \overline{f}_{ii}x_i(k) + \sum_{j \neq i}[\overline{E}_{ij}\overline{\boldsymbol{u}}_j + \overline{g}_{ij}\overline{\boldsymbol{x}}_j + \overline{f}_{ij}x_j(k)] \qquad \forall\, i = 1, \ldots, M \tag{A.1}$$

in which

$$\overline{E}_{ij} = \begin{bmatrix} B_{ij} & 0 & \ldots & \ldots & 0 \\ A_{ii}B_{ij} & B_{ij} & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ A_{ii}^{N-1}B_{ij} & \ldots & \ldots & \ldots & B_{ij} \end{bmatrix} \qquad \overline{f}_{ij} = \begin{bmatrix} A_{ij} \\ A_{ii}A_{ij} \\ \vdots \\ A_{ii}^{N-1}A_{ij} \end{bmatrix}$$

$$\overline{g}_{ij} = \begin{bmatrix} 0 & 0 & \ldots & \ldots & 0 \\ A_{ij} & 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ A_{ii}^{N-2}A_{ij} & A_{ii}^{N-3}A_{ij} & \ldots & \ldots & 0 \end{bmatrix}.$$

Combining the models in (A.1), $\forall\, i = 1, \ldots, M$, gives the following system of equations

$$\mathcal{A}\widetilde{\boldsymbol{x}} = \mathcal{E}\widetilde{\boldsymbol{u}} + \mathcal{G}x(k) \tag{A.2}$$

in which

$$\mathcal{G} = \begin{bmatrix} \overline{f}_{11} & \overline{f}_{12} & \ldots & \overline{f}_{1M} \\ \overline{f}_{21} & \overline{f}_{22} & \ldots & \overline{f}_{2M} \\ \ddots & \ddots & \ddots & \ddots \\ \overline{f}_{M1} & \ldots & \ldots & \overline{f}_{MM} \end{bmatrix} \qquad \mathcal{E} = \begin{bmatrix} \overline{E}_{11} & \overline{E}_{12} & \ldots & \overline{E}_{1M} \\ \overline{E}_{21} & \overline{E}_{22} & \ldots & \overline{E}_{2M} \\ \ddots & \ddots & \ddots & \ddots \\ \overline{E}_{M1} & \ldots & \ldots & \overline{E}_{MM} \end{bmatrix}$$

$$\mathcal{A} = \begin{bmatrix} I & -\overline{g}_{12} & \ldots & -\overline{g}_{1M} \\ -\overline{g}_{21} & I & \ldots & -\overline{g}_{2M} \\ \ddots & \ddots & \ddots & \ddots \\ -\overline{g}_{M1} & \ldots & \ldots & I \end{bmatrix} \quad \widetilde{\boldsymbol{x}} = \begin{bmatrix} \overline{\boldsymbol{x}}_1 \\ \overline{\boldsymbol{x}}_2 \\ \vdots \\ \overline{\boldsymbol{x}}_M \end{bmatrix} \quad \widetilde{\boldsymbol{u}} = \begin{bmatrix} \overline{\boldsymbol{u}}_1 \\ \overline{\boldsymbol{u}}_2 \\ \vdots \\ \overline{\boldsymbol{u}}_M \end{bmatrix} \tag{A.3}$$

Since the system is LTI, a solution to the system (A.2) exists for each permissible RHS. Matrix $\mathcal{A}$ is therefore invertible and consequently, we can write for each $i = 1, \ldots, M$

$$\overline{\boldsymbol{x}}_i = E_{ii}\overline{\boldsymbol{u}}_i + f_{ii}x_i(k) + \sum_{j \neq i}[E_{ij}\overline{\boldsymbol{u}}_j + f_{ij}x_j(k)]. \tag{A.4}$$

REFERENCES

Camacho, E.F. and C. Bordons (2004). *Model Predictive Control, Second Edition*. Springer Verlag, New York, NY.

Camponogara, Eduardo, Dong Jia, Bruce H. Krogh and Sarosh Talukdar (2002). Distributed model predictive control. *IEEE Ctl. Sys. Mag.* pp. 44–52.

Hingorani, Narain G. and Laszlo Gyugyi (2000). *Understanding FACTS*. IEEE Press, New York, NY.

Jia, Dong and Bruce H. Krogh (2001). Distributed model predictive control. In: *Proceedings of the American Control Conference*. Arlington, Virginia.

Krogh, Bruce and Petar V. Kokotovic (1984). Feedback control of overloaded networks. In: *IEEE Transactions on Automatic Control*. Vol. AC-29, No. 8, pp. 704–711.

Qin, S. Joe and Thomas A. Badgwell (2003). A survey of industrial model predictive control technology. *Control Eng. Prac.* **11**(7), 733–764.

Wood, A.J. and Bruce F. Wollenberg (1996). *Power Generation Operation and Control*. John Wiley and Sons, New York, NY.