Development of Distributed Model Predictive Control Tools for Power Generation Systems

Leonardo Escamilla III
Dept. of Mechanical and Aerospace Engineering
Klipsch School of Electrical and Computer Engineering
New Mexico State University
Las Cruces, NM, USA
brwnL30@nmsu.edu

Yang Shi
Dept. of Mechanical, Aerospace, and Biomedical Engineering
University of Tennessee, Knoxville
Knoxville, TN, USA
yshi38@vols.utk.edu

Zhenbo Wang
Dept. of Mechanical, Aerospace, and Biomedical Engineering
University of Tennessee, Knoxville
Knoxville, TN, USA
zwang124@utk.edu

Sijo Augustine
Klipsch School of Electrical and Computer Engineering
New Mexico State University
Las Cruces, NM, USA
sijo@nmsu.edu

Olga Lavrova
Klipsch School of Electrical and Computer Engineering
New Mexico State University
Las Cruces, NM, USA
olavrova@nmsu.edu

Satishkuma Ranade
Klipsch School of Electrical and Computer Engineering
New Mexico State University
Las Cruces, NM, USA
sranade@nmsu.edu

Liang Sun
Dept. of Mechanical and Aerospace Engineering
New Mexico State University
Las Cruces, NM, USA
lsun@nmsu.edu

2021 North American Power Symposium (NAPS) | 978-1-6654-2081-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/NAPS52732.2021.9654508

***Abstract*—One of the most prominent issues facing smart grid development is the Load Frequency Control (LFC) problem. A promising approach to solving this issue is to use distributed optimization techniques. The Distributed Model Predictive Control (DMPC) technology has been leveraged to solve the LFC problem. While the theory of MPC and DMPC has been developed for decades, the lack of easy-to-implement DMPC tools has been a barrier for researchers and practitioners to adopt DMPC technology in real-world applications. In this paper, we apply an open-source DMPC toolbox, namely MPCTools, to solve the LFC problem of a multi-area power generation system. The simulation is established in MATLAB Simulink and the simulation results show the effectiveness of the DMPC strategy using MPCTools.**

***Index Terms*—Distributed Model Predictive Control, MPCTools, Power Systems, Load Frequency Control, Implementation**

I. Introduction

As modern society continues to flourish, the demand for stable and robust power systems has grown. In modern power systems with interconnected areas, the balance between generation and load is crucial to maintain the grid frequency within stability margins [1]. The generation within each area must be controlled so that the system frequency and the scheduled power exchange is maintained [2]. Load frequency control (LFC) technology has been leveraged to regulate the active power and frequency. As the grid size grows with emerging renewable energy sources, independently-controlled and interconnected areas pose new challenges to LFC, making the centralized control impractical. While techniques such as optimization and machine learning have been used for LFC [3], distributed model predictive control (DMPC) has drawn researchers' attention for LFC of emerging power grids [4].

Model predictive control, also called receding horizon control, produces an open-loop control input by solving a discrete-time optimal control problem over a given time horizon [2], [5], [6]. DMPC relies on information exchange among areas to decompose a centralized MPC problem into much smaller local optimization problems [2], [4]. While DMPC technology has been witnessed in a wide range of applications [7]–[9], practical DMPC tools for power-system research (e.g., LFC) have not been well developed.

The contribution of this paper is the application of an open-source DMPC tool [10], namely MPCTools, for the LFC of multiple interconnected power generation systems. MPCTools is designed to run in Octave [11] and MATLAB. It is a complete software package that does not rely on remote servers or additional specific software. It can be installed and function on a stand-alone computer. Additionally, MPCTools is a convenient and simple DMPC tool for research and education. In this paper, a two-area and a four-area LFC problems were adopted to present the implementation of DMPC in MPCTools. The simulation results show the effectiveness of the applied DMPC controllers.

The remaining of the paper is structured as follows. Section II presents key terminologies and the selected example LFC problem. In Section III, the MATLAB Simulink environment for the proposed problem is presented, as well as the introduction of MPCTools. Section IV presents and discusses the numerical results and Section V concludes the paper.

II. Preliminaries and Problem Statement

A. Definitions and Terminologies of DMPC

Some key terminologies and notations associated with the DMPC technology are listed as follows.

- *Prediction Horizon* (T): the amount of time steps an MPC controller will predict over.

- *Control Horizon* (T_c): the number of optimal control input moves generated during the time horizon [12].
- *Sampling time* (T_s): the time between measurement updates.
- *Integration step* (δ_{int}): the step size used to solve the differential algebraic equations for the dynamics model within an MPC controller.
- *Controlled Variables* (y): the variables that are being regulated by the controller.
- *Manipulated Variables* (u): input variables to the system that regulate the dynamics to a set-point. The values of controlled variables are determined by the optimization solver given the objective function.
- *State Variables*: the variables used to describe the system dynamics represented in state-space forms.
- *Constraints*: limitations imposed on the system dynamics and controlled variables. These limitations desctibe the real-world limitations of physical systems, such as rotor speed and power output.

B. Example System

A common setup to analyze the Load Frequency Control problem can be viewed in Fig. 1. Within the setup, N generators are connected to each other through a tie-line. The characteristics of the generator located at the i^{th} area are described by its voltage, E_i, phase angel, δ_i, and internal reactance, X_i. The tie-line connecting generators i and j is represented by its reactance $X_{tie_{ij}}$. To simplify the problem, the reactance of the i^{th} and j^{th} generators, and their respective tie-lines are summed and represented as $X_{T_{ij}}$. The power being transmitted between the i^{th} and j^{th} generator locations, in p.u., is represented by P_{ij}.

The goal of the LFC problem is to maintain the scheduled delivery of power between generator sites while zeroing the frequency difference between generators. To complete this task, each area is outfitted with an MPC controller, which directly controls the power output of its generator. Each MPC controller has access to the full state information of the attached generator site.

The state-space equations for the i^{th} area are given by [2]

$$\Delta\dot{\delta}_i = 2\pi\Delta f_i(t) \tag{1}$$

$$\Delta\dot{f}_i = \frac{-\Delta f_i(t)}{T_{P_i}} + \frac{K_{P_i}\Delta P_{g_i}(t)}{T_{P_i}} - \frac{K_{P_i}\Delta P_{d_i}(t)}{T_{P_i}} - \frac{K_{P_i}}{2\pi T_{P_i}}\left(\sum_{j\in N} K_{S_{ij}}\left[\Delta\delta_i(t) - \Delta\delta_j(t)\right]\right) \tag{2}$$

where

- $\Delta\delta_i(t)$: the continuous-time incremental change in the phase angle of the i^{th} generator bus in units of radians.
- $\Delta f_i(t)$: the continuous-time incremental change in the frequency of the i^{th} generator bus in units of Hz.
- $\Delta P_{g_i}(t)$: the continuous-time incremental change in the generator output of the i^{th} area in units per unit (p.u.). It is directly determined by the i^{th} MPC controller.

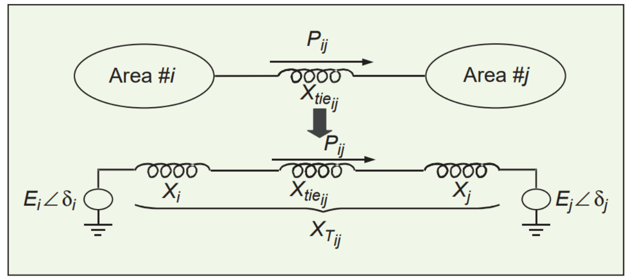

Fig. 1: A multi-area distributed power system [2].

- ΔP_{d_i}: the continuous-time incremental load disturbance of the i^{th} area in units of per unit (p.u.).
- T_{P_i}: the system model constant of the i^{th} generator area in units of seconds.
- K_{P_i}: the tuneable system gain of the i^{th} generator area.
- $K_{S_{ij}}$: the tuneable synchronizing coefficient of the connection line between the i^{th} and j^{th} generator areas.

In addition to the dynamic model of the system, an MPC controller requires an objective function to be minimized/maximized with respect to certain constraints of the system. The result of the optimization is the optimal control input to the system at a given time. The selection of the objective function is problem-dependent and also user-dependent, and different objective functions will result in different performances of the controller. For the proposed LFC problem with the dynamics defined in Equations (1) and (2), the objective function is selected as [2]

$$J_i(t) = \int_t^{t+T}\left[\sum_{\substack{j=1\\ j\neq i}}^{M} p_{ij}(\Delta\delta_i(\tau) - \Delta\delta_j(\tau))^2 + q_i\Delta f_i^2(\tau) + r_i\Delta P_{gi}^2(\tau)\right]d\tau \tag{3}$$

where p_{ij}, q_i, and r_i are tuneable weighting parameters.

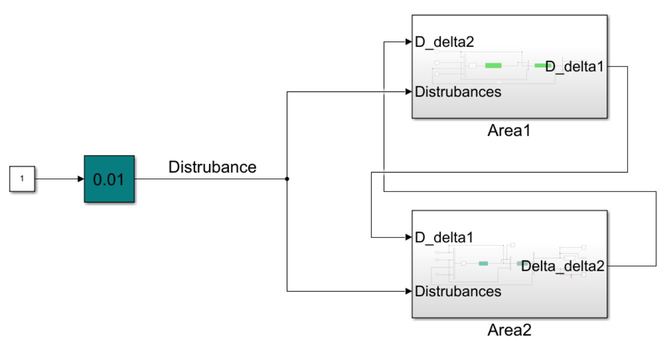

Fig. 2: A two-area LFC structure.

III. Methodology

In this section, we present the development of a DMPC system using MPCTools in MATLAB Simulink.

A. Structure of the Platform

Following the interconnected power generation systems in Figure 1, the overview of the two-area system to be regulated is shown in Fig. 2. We assume a constant disturbance of 0.01 p.u. in this paper. Each "Area" block represents a generator sytem that contains the dynamics and MPC controller components. Each "Area" is assuemd to have direct access to $\Delta\delta$ of the other site through a communication network.

B. Components of the Platform

Figure 3 shows the internal structure of each area within the LFC model. The major components of each area are listed as follows:

- *Zero-order hold*: Model Predictive Controllers are discrete controllers, therefore the system must discretize the dynamics. The zero-order hold is responsible for taking the continuous-time signal and converting it into discrete-time values. The sampling time of the block is T_s.
- *MPC Controller*: The MPC Controller block of the i^{th} area takes in $\Delta\delta$ from all other areas in the network, the area disturbance, and the local state variables. It finds the optimal ΔP_{g_i} by minimizing the objective function within this block. The controller then makes an incremental change in the power output of the generator. This block was implemented utilizing MPCTools.
- *Dynamic sytem*: This block is responsible for emulating the dynamics of the respective generator area.

C. MPCTools

MPCTools [10], is a model predictive control toolbox for MATLAB and Octave. MPCTool set up and solve an MPC problem by utilizing CasADi, which is a free, open-source nonlinear optimization and algorithmic differentiation platform [13] and unique from other similar platforms as it offers an intuitive symbolic framework for approaching optimization problems, such as Nonlinear control problems. MPCTools is designed to allow for researchers and engineers to intuitively code linear and nonlinear MPC problems. There are many built-in functions that can be directly called or modified according to problem specifications.

Within this work, MPCTools was utilized to take advantage of CasADi, to leverage its ability to handle MPC problems, and to extend its use to a distributed system. CasADi was utilized to solve the dynamics model used by the MPC controller, as well as to minimized the proposed objective function in Equation(3). The Runge–Kutta technique (rk4) was utilized to solve the differential algebraic equations to describe the system dynamics. The Interior Point OPTimizer (IPOPT) was utilized to find the solution that minimizes the objective function at each time step.

IV. Simulation Results

A. Two-Area Problem

To start off, we used the MPCTools to solve a two-area LFC control problem. Each system has an MPC controller that has access to the states of its area and the phase angle change ($\Delta\delta$) of the other site. A constant disturbance of 0.01 p.u. is assumed. The output of the generators were constrained to be within [-3, 3] p.u.. No constraints were imposed on the system states. The goal of each site's MPC controller is to minimized the difference between the phase angle changes, i.e., $(\Delta\delta_i - \Delta\delta_j)^2$, and the frequency change, Δf_i.

For this experiment, the sampling time step, T_s, was tuned and selected as 0.01 s, 0.1 s, and 0.2 s, respectively, to compare the representative system performance. These values were selected to best represent the behavior of the system as the sampling frequency decreased. With a T_s selected, the prediction horizon, T, of the MPC controllers was varied. The results of the experiment are shown in Figs. 4 through 6.

From Figs. 4 and 5, it can be seen that both Δf_1 and Δf_2 are successfully regulated to a constant near-zero value and $\Delta\delta_1$ and $\Delta\delta_2$ are maintained with a near-constant difference. It can also be seen that a smaller sampling time results in smoother stability curves. Additionally, increasing the prediction horizon improves the convergence speed until the value of $T = 3$, after which increasing T results in poorer performance.

From Fig. 6, it can be seen that a sample time of 0.2 s results in instability of the system. In the event of a large sampling time, a larger prediction horizon value results in rapid instability. Meanwhile, smaller time horizon values, while still unstable, diverge at a slower rate.

B. Four-Area Problem

With the results obtained from the two-area problem, we expanded to a four-area problem. This was completed by introducing two more generators operating under similar conditions to the previous two areas. Once again, the sampling time, T_s, was varied within the previously selected values. Then, the prediction-horizon values of the controllers was varied.

From Figs. 7 through 9, it can be seen that a smaller sampling time results in significantly better performance. By observing Fig. 7, there once again appears to be a point at which a larger prediction horizon begins to negatively affect the performance of the controller. In the case of $T_s = 0.01$ s, the largest prediction horizon value that should be used is $T = 3$ s.

Unlike the two-area LFC problem, there is a noticeable difference in the performance of the DMPC strategy when T_s is increased from 0.01 s to 0.1 s. Figure 8 shows that increasing the sampling time value causes Δf of each area to converge to increasingly non-zero values. In this case, the best performance is obtained by selecting a smaller time horizon, $T = 2$ s. Anything larger than this value negatively impacts both the rise time and steady-state value.

Finally, for $T_s = 0.2$ s, it can be seen from Fig. 9 that the performance of the DMPC control strategy is degraded. Unlike the two-area system, stability is still possible, but there is much oscillation left in the system. Further increasing the sampling time will surely lead to instability. Increasing the prediction horizon negatively affects the performance of the control strategy.

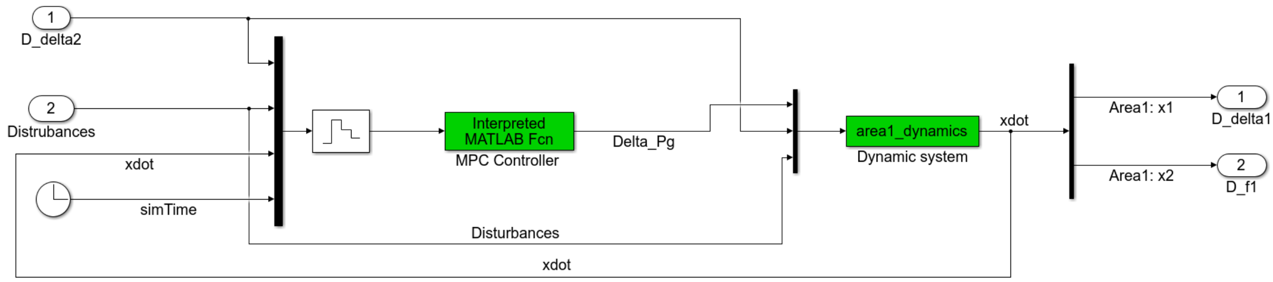

Fig. 3: Internal components of Area 1.

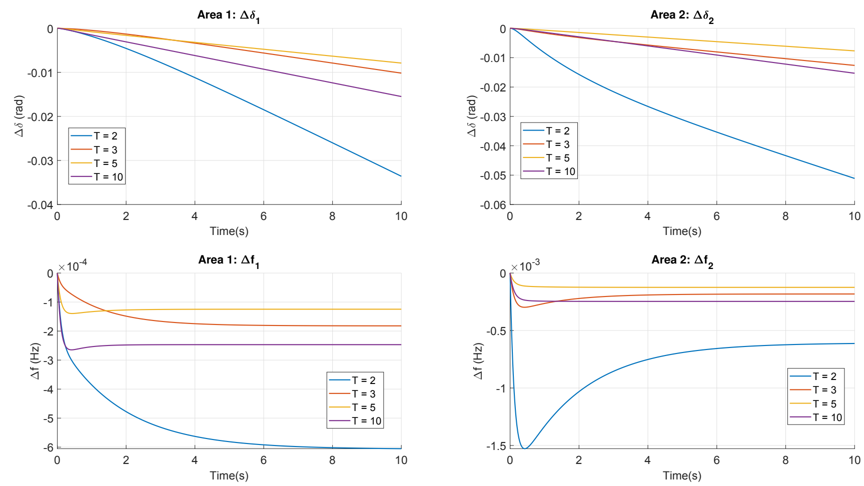

Fig. 4: Two-area LFC result: Ts = 0.01 s.

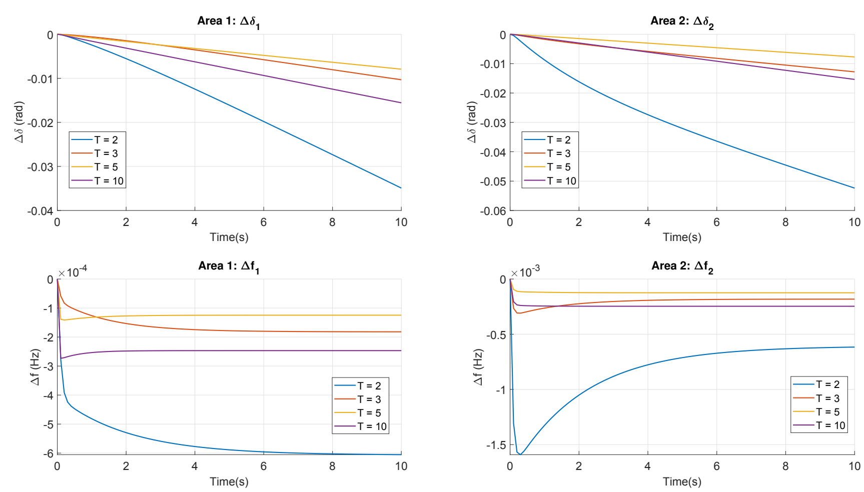

Fig. 5: Two-area LFC result: Ts = 0.1 s.

V. Conclusion

The MPCTools, an open-source MPC software package tool, was applied to the proposed distributed Model Predictive Control (DMPC) structure in MATLAB simulink for the two-area and four-area load frequence control (LFC) problems. The simulation setup process and its results show the effectiveness of the applied DMPC strategy and MPCTools to the LFC problem. In both cases, provided a small sampling time, the DMPC system was capable of achieving the control objective. It was also shown that there exists an optimal prediction

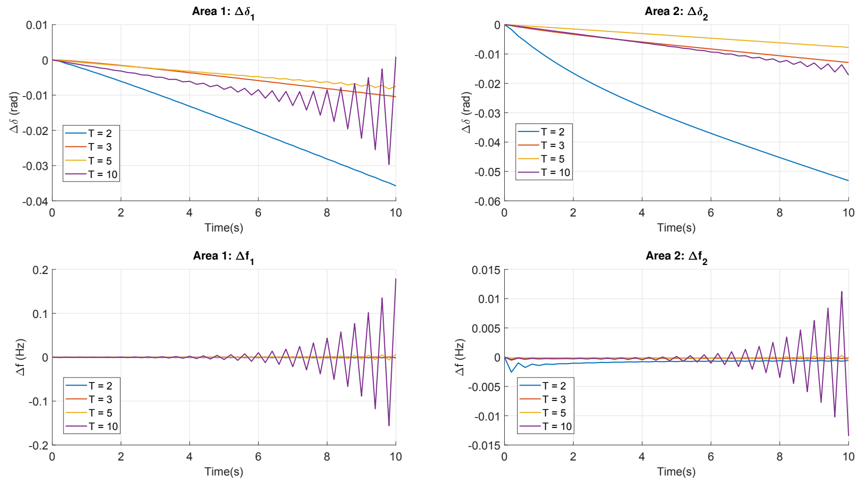

Fig. 6: Two-area LFC result: Ts = 0.2 s.

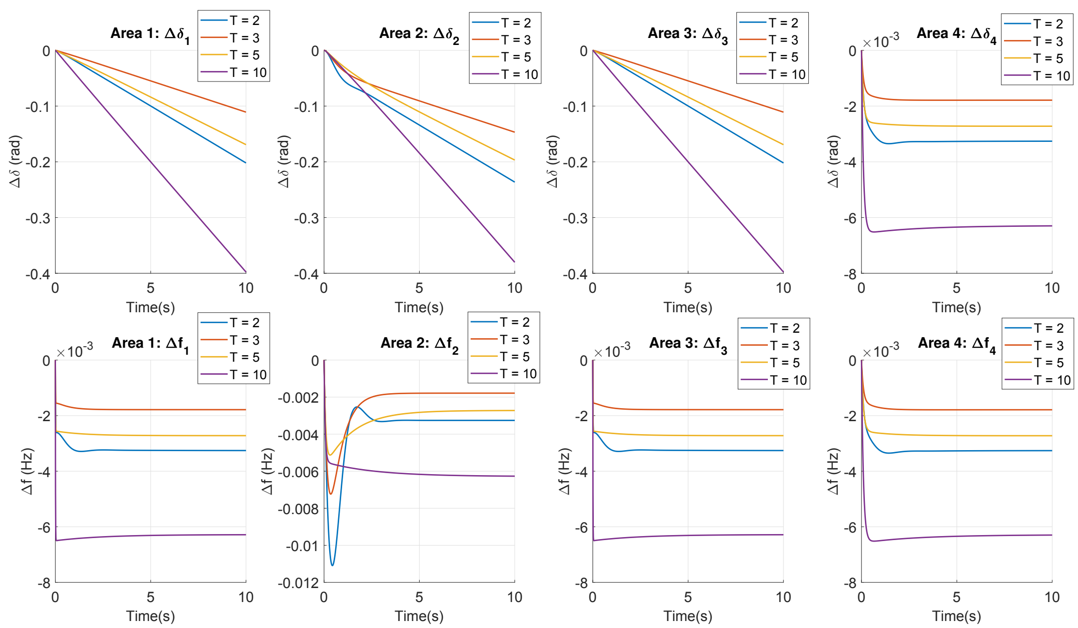

Fig. 7: Four-area LFC result: Ts = 0.01 s.

horizon value that improves the rise time and steady-state value of the system.

Acknowledgment

This work was partially supported by the NSF Grants #OIA-1757207 (NM EPSCoR), HRD-1345232, HRD-1914635 and funding from the Electric Utility Management Program (EUMP) at the New Mexico State University.

References

[1] A. Kannan, M. Nuschke, and D. Strau-Mincu, "Lfc model for frequency stability analysis of prospective power systems with high shares of inverter based generation," in *2019 IEEE Milan PowerTech*. IEEE, 2019, pp. 1–6.

[2] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar, "Distributed model predictive control," *IEEE control systems magazine*, vol. 22, no. 1, pp. 44–52, 2002.

[3] D. Sharma, "Load frequency control: a literature review," *Int J Sci Technol Res*, vol. 9, no. 2, pp. 6421–6437, 2020.

[4] I. E. Uyioghosa and A. K. Saha, "DMPC scheme for load frequency control with application to interconnected power system," in *International SAUPEC/RobMech/PRASA Conference*. IEEE, 2020, pp. 1–6.

[5] C. Estrada and L. Sun, "Trajectory tracking control of a drone-guided hose system for fluid delivery," in *AIAA Scitech 2021 Forum*, 2021, p. 1003.

[6] L. Sun, J. D. Hedengren, and R. W. Beard, "Optimal trajectory generation using model predictive control for aerially towed cable systems," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 2, pp. 525–539, 2014.

[7] H. Li, B. Jin, and W. Yan, "Distributed model predictive control for linear systems under communication noise: Algorithm, theory and

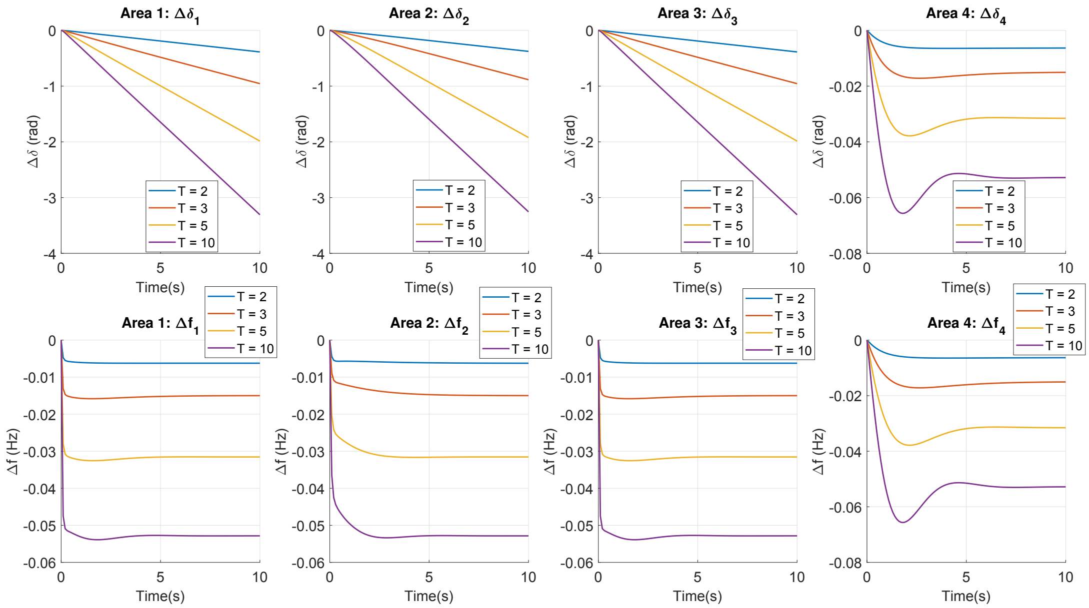

Fig. 8: Four-area LFC result: Ts = 0.1 s.

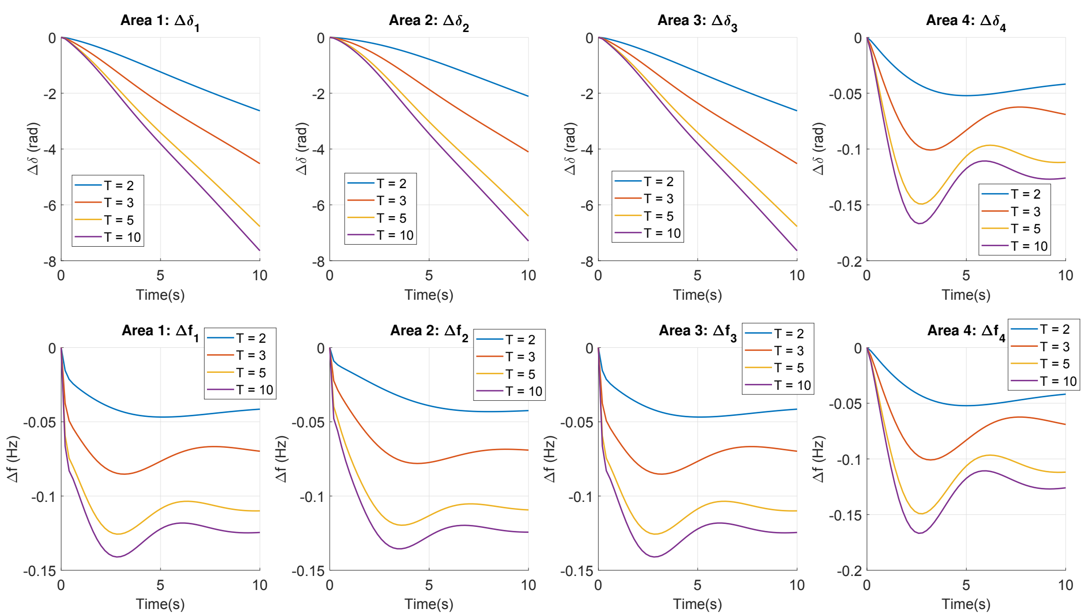

Fig. 9: Four-area LFC result: Ts = 0.2 s.

implementation," *Automatica*, vol. 125, p. 109422, 2021.

[8] G. Darivianakis, A. Eichler, and J. Lygeros, "Distributed model predictive control for linear systems with adaptive terminal sets," *IEEE Transactions on Automatic Control*, vol. 65, no. 3, pp. 1044–1056, 2019.

[9] Y. Zhou, M. Wang, and S. Ahn, "Distributed model predictive control approach for cooperative car-following with guaranteed local and string stability," *Transportation research part B: methodological*, vol. 128, pp. 69–86, 2019.

[10] M. Risbeck and J. Rawlings, "MPCTools: Nonlinear model predictive control tools for CasADi (Octave interface), 2018a," *URL: https://bitbucket. org/rawlings-group/octave-mpctools*.

[11] GNU Octave, *URL: https://www.gnu.org/software/octave/*.

[12] Choose Sample Time and Horizons, *URL: https://www.mathworks.com/help/mpc/ug/choosing-sample-time-and-horizons.html#buj0ytu*.

[13] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.