STABILITY ANALYSIS TOOL FOR TUNING UNCONSTRAINED DECENTRALIZED MODEL PREDICTIVE CONTROLLERS

Massimo Vaccarini * **Sauro Longhi** * **M. Reza Katebi** **

D.I.I.G.A., Università Politecnica delle Marche, Ancona, Italy
** *Industrial Control Centre, University of Strathclyde, Glasgow, Scotland, UK*

Abstract: Some processes are naturally suitable to be controlled in a decentralized framework: centralized control solutions are often infeasible in dealing with large scale plants and they are technologically prohibitive when the processes are too fast for the available computational resources. In these cases, the resulting control problem is usually split in many smaller subproblems and the global requirements are guaranteed by means of a proper coordination. The unconstrained decentralized case is here considered and a coordination strategy is proposed for improving the global control performances. This paper present a tool for setting up and tuning a nominally stable decentralized Model Predictive Controller. Numerical examples are proposed for testing and validating the developed technique.

Keywords: Decentralized Control, Predictive Control, Communication Networks, Stability Analysis, Optimal Regulators

1. INTRODUCTION

The current diffusion of networks allows the control technologies and methodologies to fully express their potentials in many application fields. By means of the fast communication technologies, nowadays different independent controllers distributed in a wide area can exchange information in order to improve both the local and the global control performances. Robotic applications with multiple autonomous agents pursuing a common goal, control applications in manufacturing and process industry where multiple units cooperatively make a product, supply chains in which multiple actors that influence each other are involved, and large scale power systems represent some typical situations.

A decentralized control solution based on independent agents is here considered for the regulation of different interacting processes. Global objectives, such as closed-loop stability and performance requirements for the global process require coordination among the control agents.

Different solutions have been developed in literature. The required coordination can be introduced by a hierarchical decentralized control scheme, where a supervisor computes the global optimum and coordinates all the control agents (Šiljak, 1996). A decentralized filtering and control architecture managed by a global coordinator has been proposed by Katebi and Johnson (1997) for guaranteing steady state optimality of the control actions. A different solution based on the team theory and asynchronous teams has been proposed in Camponogara (2000). A decomposition of the global process into subsystems, based on the structural properties and physical constraints, has been developed in Jia and Krogh (2001), Cheng and Krogh (2001), Camponogara *et al.* (2002), Jia (2003), where each agent controls a subsystem making use of local model, objectives and constraints. Gudi *et al.* (2004) investigated the identification procedure oriented to decentralized MPC. The effectiveness of the communication based MPC and the importance of the interaction analysis for improving the control performances has been validated in Venkat *et al.* (2005).

A partitioning of both actuators and sensors has been proposed by Motee and Sayyar-Rodsari (2003) as a solution for the problem of computational complexity for the use of MPC technology in large scale dynamic systems. Adopting a proper decentralized solution, the global computational effort can be reduced without significant degradations of the control performances and the fault tolerant issues can be improved (El-Farra *et al.*, 2004).

The functionalities of a Networked Decentralized Model Predictive Control solution (ND-MPC) have been recently tested on real plants with relatively strong interactions. The results obtained with the decentralized MPC of a gasifier seem to be satisfactory if compared with those obtained by the classical centralized MPC (Longhi *et al.*, 2005). Stability analysis tools for testing the stability of the ND-MPC architecture have been also proposed by Vaccarini *et al.* (2006).

This paper summarizes the results on the stability analysis developed by the authors and present a set of simulation tests to validate the proposed tool and to illustrate its use for the synthesis of decentralized controllers with communication networks. After a description of the adopted ND-MPC strategy provided in Section 2, the stability analysis tool is presented in Section 3. Based on the former considerations, the control algorithm is outlined in Section 4 and the tool is applied to the synthesis of decentralized controllers for a set of testing plants in Section 5.

In order to simplify the mathematical expressions, some notations are here introduced. Given the numbers $k \in \mathbb{Z}$, $h \in \mathbb{Z}$, $m \in \mathbb{N}$, $j \in \mathbb{N}$, $i \in \mathbb{N}$, $n \in \mathbb{N}$ such that $k \geq h$, $j \geq 1$, $m \geq h - k + 1$, $i = 1, \ldots, n$:

- $\|v\|_A \triangleq A^T v A$ is the norm of vector v induced by matrix A;
- $\lambda_j\{A\}$ is the j-th eigenvalue of a square matrix A.
- $\hat{x}(k|h) \triangleq \mathscr{E}\left[x(k)\left|\mathfrak{Y}^h\right.\right]$ is the $(h - k)$-step ahead prediction of x, given the measurements $\mathfrak{Y}^h$ up to time k;
- $u(k|h)$ is the value of $u(k)$, computed at time h;
- $X_i(k, m|h)$ is a stacked vector made by the vectors $x_i(k|h), \ldots, x_i(k + m - 1|h)$;
- $X(k, m|h)$ is a stacked vector made by the vectors $X_1(k, m|h), \ldots, X_n(k, m|h)$;

2. NETWORKED DECENTRALIZED MPC

For achieving global performance objectives, a coordination scheme based on communication among control agents is developed. The control actions are computed by a set of subcontrollers which are independent agents able to dynamically exchange a restricted set of information. In the proposed control architecture, each agent $\mathscr{A}_i$ implements an MPC algorithm for the subsystem $\mathscr{S}_i$ using both local information acquired

on $\mathscr{S}_i$ and the estimate of the interactions among $\mathscr{S}_i$ and the other subsystems $\mathscr{S}_j$, $j = 1, \ldots, n$, $j \neq i$. The resulting optimal sequence and the future prediction of the state over the prediction horizon, have to be exchanged among subsystems through a Local Area Network (LAN).

As well known, Model Predictive Control acts according to the receding horizon principle: at each sampling time, using a predictive model of the system dynamics, the response of the process to changes in manipulated variables over a fixed horizon is predicted. Based on a proper cost function, a finite-horizon optimal control problem is solved to obtain current and future moves of the manipulated variables. Only the first computed move is applied to the real system. The same procedure is repeated at the next control step based on the new measurement. Although this computation is an open-loop control problem, the receding horizon principle allows MPC to generate a feedback-control law.

Let consider a linear, discrete-time representation of a plant $\mathscr{P}$:

$$x(k + 1) = Ax(k) + Bu(k) \quad (1a)$$
$$y(k) = Cx(k). \quad (1b)$$

and denote with $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$, $y(k) \in \mathbb{R}^{n_y}$ and $y^d \in \mathbb{R}^{n_y}$, its state, control input, output and desired output, respectively. Suppose that $\mathscr{P}$ is composed by n subsystems $\mathscr{S}_i$ whose state-space representation is:

$$x_i(k + 1) = A_{ii}x_i(k) + B_{ii}u_i(k) + w_i(k) \quad (2a)$$
$$y_i(k) = C_{ii}x_i(k) + v_i(k). \quad (2b)$$

where vectors x_{ii}, u_i, y_i and y_i^d are the local state, control input, output and desired output respectively, and vectors w_i and v_i, named state and output interaction vectors, are given by:

$$v_i(k) \triangleq \sum_{j=1(j \neq i)}^{n} A_{ij}x_j(k) + \sum_{j=1(j \neq i)}^{n} B_{ij}u_j(k), \quad (3a)$$

$$w_i(k) \triangleq \sum_{j=1(j \neq i)}^{n} C_{ij}x_j(k). \quad (3b)$$

Definition 1. (ND-MPC). Given the plant $\mathscr{P}$ composed by n subsystems $\mathscr{S}_i$, $i = 1, \ldots, n$ the unconstrained Networked Decentralized Model Predictive Control problem with prediction horizon p and control horizon m consists of finding, at time k, a set of independent agents $\mathscr{A}_i$, $i = 1, \ldots, n$, such that each $\mathscr{A}_i$ minimizes the local cost function

$$J_i \triangleq \sum_{l=1}^{p} \left\|\hat{y}_i(k + l|k) - y_i^d(k + l|k)\right\|_{Q_i} + \\ + \sum_{l=1}^{m} \left\|\Delta u_i(k + l - 1|k)\right\|_{R_i}, \quad (4a)$$

subject, for $l = 1, \ldots, p$, to the model constraint:

$$\hat{y}_i(k+l|k) = \hat{v}_i(k+l|k-1) + C_{ii}A_{ii}^l\hat{x}_i(k|k) +$$
$$+ \sum_{s=1}^{l} C_{ii}A_{ii}^{s-1}B_{ii}u_i(k+l-s|k) +$$
$$+ \sum_{s=1}^{l} C_{ii}A_{ii}^{s-1}\hat{w}_i(k+l-s|k-1). \quad \text{(4b)}$$

Definition 2. (Interaction). When a changes in input, state or output variables of a subsystem $\mathscr{S}_i$ produces variations of input, state or output variables of a subsystem $\mathscr{S}_j$, it is said that $\mathscr{S}_i$ interacts with $\mathscr{S}_j$.

Definition 3. (Connection). When the agent $\mathscr{A}_i$ sends information about the future behavior of subsystem $\mathscr{S}_i$ to the agent $\mathscr{A}_j$, it is said that $\mathscr{A}_i$ is connected with $\mathscr{A}_j$.

Definition 4. (Neighboring Agent). If the agent $\mathscr{A}_i$ is connected with $\mathscr{A}_j$, $\mathscr{A}_i$ is called input neighboring agent of $\mathscr{A}_j$ and $\mathscr{A}_j$ is called output neighboring agent of $\mathscr{A}_i$. $\mathscr{A}_i$ and $\mathscr{A}_j$ are said neighboring agents.

Definition 5. (Neighborhood of an Agent). The input (output) neighborhood of an agent $\mathscr{A}_i$ is the set of its input (output) neighboring agents.

Each agent $\mathscr{A}_i$, solution to the ND-MPC problem, can be decomposed in three parts: an optimizer, a state predictor and an interaction predictor. At time k, based on the exchanged information, the interaction prediction together with the local measurement is used by the optimizer to solve the MPC optimization problem. Once computed the optimal sequence $\{\Delta u_i(k|k),\ldots,\Delta u_i(k+m-1|k)\}$, which minimize the cost function (4a), the first element $\Delta u_i(k|k)$ is selected and $u_i(k) = u_i(k-1) + \Delta u_i(k|k)$ is applied as control input to $\mathscr{S}_i$. Then the state predictor computes an estimate of the future state trajectory and broadcasts the optimal control sequence over the control horizon and the state predictions over the prediction horizon to its output neighborhood.

The following assumptions are here considered:

Assumptions 1.

- prediction and control horizons are the same for each agent;
- control agents are synchronous;
- control agents communicate only once within a sampling interval;
- the communication channel introduces a delay of a single sampling time.

Lemma 1. (Quadratic Program). Under the Assumptions 1, at step k, each agent $\mathscr{A}_i$ solution to the ND-MPC problem, has to solve the following optimization problem:

$$\min_{\Delta U_i(k,m|k)} J_i = \Delta U_i(k,m|k)^T H_i \Delta U_i(k,m|k) - G_i^T \Delta U_i(k,m|k).$$
$$\text{(5)}$$

where
$$H_i = N_i^T \bar{Q}_i N_i + \bar{R}_i, \quad \text{(6a)}$$
$$G_i = 2N_i^T \bar{Q}_i[Y_i^d(k+1,p|k) - L_i\hat{x}_i(k|k) - M_i u_i(k-1) +$$
$$- S_i\hat{W}_i(k,p|k-1) - T_i\hat{V}_i(k,p|k-1)]. \quad \text{(6b)}$$

The introduced matrices are used for computing the output predictions: L_i, M_i, N_i, S_i depend only on the system matrices A_i, B_i and C_i, matrix T_i introduces a unit delay in the interaction vector and matrices $\bar{Q}_i$, $\bar{R}_i$ are made by a block replication of the local weighting matrices Q_i, R_i. Equation (5) defines an unconstrained Quadratic Program which has to be solved on-line at each sampling instant.

For the sake of brevity, all the proofs of the results stated in this paper are are omitted. Refer to Longhi *et al.* (2005) and Vaccarini *et al.* (2006) for further details.

3. STABILITY ANALYSIS OF ND-MPC

The main idea is to find an explicit solution for the ND-MPC problem and to use it for obtaining a mathematical representation of the closed-loop system. Once the closed-loop dynamic is known, the stability condition can be verified by analyzing the dynamic matrix.

For this purpose, denote with $\tilde{A}_i$, $\tilde{B}_i$, $\tilde{C}_i$, $\bar{A}_i$ and $\bar{B}_i$ proper local interaction matrices and with $\tilde{A}$, $\tilde{B}$, $\tilde{C}$, $\bar{A}$ and $\bar{B}$ their corresponding global stacked versions. The gain of each MPC agent for the control effort movement is represented by K_i whereas matrix κ_i represents the gain for the magnitude of the local control effort over the whole control horizon m. Define I^0 as the matrix for selecting the first computed movements from the optimal control sequence. Denote with $\bar{L}_i$, $\bar{M}_i$ two state prediction matrices composed by system matrices, and introduce I_i' as an auxiliary identity block matrix. Assume that L, M, $\bar{L}$, $\bar{M}$, S, T, I', κ are diagonalizations of the local matrices L_i, M_i, $\bar{L}_i$, $\bar{M}_i$, S_i, T_i, I_i', κ_i respectively and define:

$$\theta \triangleq -\kappa L, \qquad \phi \triangleq -\kappa\left(S\tilde{A} + T\tilde{C}\right), \quad \text{(7a)}$$
$$\rho \triangleq I'I^0 - \kappa\left(MI^0 + S\tilde{B}\right). \quad \text{(7b)}$$

Lemma 2. (Interaction Predictor). Under the Assumptions 1, at step k, for each agent $\mathscr{A}_i$ solution to the ND-MPC problem, the predictions of the interaction vectors are given by:

$$\hat{W}_i(k,p|k-1) = \tilde{A}_i\hat{X}(k,p|k-1) + \tilde{B}_iU(k-1,m|k-1),$$
$$\text{(8a)}$$
$$\hat{V}_i(k,p|k-1) = \tilde{C}_i\hat{X}(k,p|k-1), \quad \text{(8b)}$$

and the global prediction vectors take the following form:

$$\hat{W}(k,p|k-1) = \tilde{A}\hat{X}(k,p|k-1) + \tilde{B}U(k-1,m|k-1),$$
$$\text{(9a)}$$
$$\hat{V}(k,p|k-1) = \tilde{C}\hat{X}(k,p|k-1). \quad \text{(9b)}$$

The stacked vectors $X(k,p|h)$ and $U(k-1,m|h)$ are built with both the local estimations and the information collected from the input neighborhood by agent $\mathscr{A}_i$. Null entries will correspond to the subsystems which don't belong to the input neighborhood. Note that at time k, agent $\mathscr{A}_i$ uses the predictions computed and broadcasted at time $k-1$ $\{\hat{v}_i(k|k-1),\ldots,\hat{v}_i(k+p-1|k-1)\}$ and $\{\hat{w}_i(k|k-1),\ldots,\hat{w}_i(k+p-1|k-1)\}$.

Lemma 3. (State Predictor). Under the Assumptions 1, at step k, for each agent $\mathscr{A}_i$ solution to the ND-MPC problem, the decentralized state prediction for the agent $\mathscr{A}_i$ is expressed by:

$$\hat{X}_i(k+1,p|k) = \bar{L}_i\hat{x}_i(k|k) + \bar{M}_iU_i(k,m|k) + \\ + \bar{A}_i\hat{X}(k,p|k-1) + \bar{B}_iU(k-1,m|k-1). \quad (10)$$

and the decentralized prediction equation for the overall system is given by the matrix form:

$$\hat{X}(k+1,p|k) = \bar{L}\hat{x}(k|k) + \bar{M}U(k,m|k) + \\ + \bar{A}\hat{X}(k,p|k-1) + \bar{B}U(k-1,m|k-1). \quad (11)$$

Lemma 4. (Explicit Solution). Under the Assumptions 1, at step k, for each agent $\mathscr{A}_i$ solution to the ND-MPC problem, the explicit form of the control action applied by the agent $\mathscr{A}_i$ to the subsystem $\mathscr{S}_i$ is given by:

$$u_i(k) = (I - K_iM_i)\,u_i(k-1) + \\ + K_i\left[Y_i^d(k+1,p|k) - L_i\hat{x}_i(k|k)\right] + \\ - K_iS_i\hat{W}_i(k,p|k-1) - K_iT_i\hat{V}_i(k,p|k-1). \quad (12)$$

Lemma 5. (Optimal Control Sequence). Under the Assumptions 1, at step k, for each agent $\mathscr{A}_i$ solution to the ND-MPC problem, the expression of the optimal control sequence $U_i(k,m|k)$ is:

$$U_i(k,m|k) = I_i'u_i(k-1) + \\ \kappa_i\left[Y_i^d(k+1,p|k) - L_ix_i(k) - M_iu_i(k-1) + \\ - S_i\hat{W}_i(k,p|k-1) - T_i\hat{V}_i(k,p|k-1)\right]. \quad (13)$$

and its global expression is:

$$U(k,m|k) = \theta\hat{x}(k|k) + \phi\hat{X}(k,p|k-1) \\ + \rho U(k-1,m|k-1) + \\ + \kappa Y^d(k+1,p|k). \quad (14)$$

Theorem 1. (ND-MPC Stability).
The closed-loop system given by the feedback connection of the plant $\mathscr{P}$ with a solution of the ND-MPC problem, composed by a set of independent agents $\mathscr{A}_i$, $i=1,\ldots,n$, is asymptotically stable if and only if:

$$\left|\lambda_j\left\{\begin{bmatrix} A & 0 & BI^0 & 0 \\ \bar{L} & \bar{A} & \bar{M} & \bar{B} \\ \theta A + \phi\bar{L} & \phi\bar{A} & \rho + \theta BI^0 + \phi\bar{M} & \phi\bar{B} \\ 0 & 0 & I_{mn_u} & 0 \end{bmatrix}\right\}\right| < 1,$$

$$\forall j \in [1,\ldots,n_{ND}], \quad n_{ND} = pn_x + n_x + 2mn_u, \quad (15)$$

where the n_{ND} is the order of the global closed-loop system.

The first two block rows of the global closed loop dynamic matrix in Equation (15) are formed by elements of matrix A (in the first two block columns) and matrix B (in the remaining two block columns). The Third block row is made by all the process matrices A, B and C and the weighting matrices Q_i and R_i and depends also on the horizons p and m.

4. CONTROL ALGORITHM

At sample time k, each agent $\mathscr{A}_i$, solution to the ND-MPC problem:

(i) Acquires the measures.
(ii) Acquires the predicted future state trajectories $X_j(k,p|k-1)$ and control inputs $U_j(k,m|k-1)$ from the neighboring agents and, once combined with the local state trajectory $X_i(k,p|k-1)$ and the control input $U_i(k,m|k-1)$, it builds $X(k,p|k-1)$ and $U(k,m|k-1)$ and computes the corresponding predictions of the interactions (8a).
(iii) Computes the optimal control sequence (13).
(iv) Applies the first element $u_i(k) = I_i^0U_i(k,m|k)$ of the optimal sequence $U_i(k,m|k)$ as control input to $\mathscr{S}_i$.
(v) Computes the future state trajectory (10) of the subsystem $\mathscr{S}_i$ over the prediction horizon p where $\hat{x}_i(k|k) = x_i(k)$ is given by the measures.
(vi) Broadcasts the optimal sequence $U_i(k,m|k)$ and the predicted state trajectory $X_i(k+1,p|k)$ to the neighboring agents.
(vii) Iterates.

In the previous equations, the state prediction $\hat{x}_i(k|k)$ has been replaced with the actual state $x_i(k)$ because of the hypothesis of fully accessible state.

The desired output $Y_i^d(k+1,p|k)$ for the agent $\mathscr{A}_i$ is provided by a proper reference generator $\mathscr{R}_i$ that can assume either known or unknown future desired output.

5. NUMERICAL TESTS

In this section the proposed distributed control strategy is applied to a testing process. Although different weighting matrices can be used for each controller and the control horizon can be smaller than the prediction horizon, for simplifying the graphical representations, equal prediction and control horizons ($p = m$) and weighting matrices $R = \gamma I_{n_u}$ and $Q = I_{n_y}$ will be used.

The following unstable, non-minimum phase plant $\mathscr{P}$ is considered for testing the presented tools.

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \dfrac{-0.75}{(s-1)(s+1)^2} & \dfrac{\alpha}{(s+1)} \\ \dfrac{\alpha}{(s+5)} & \dfrac{-0.375(s-2)}{(s+1)^2} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix}$$
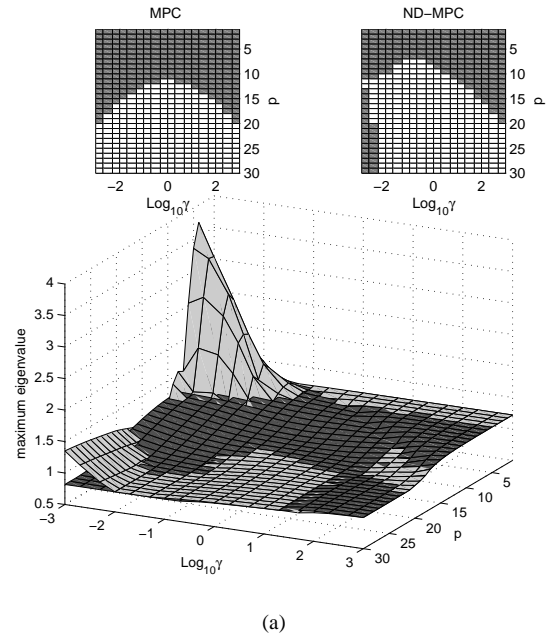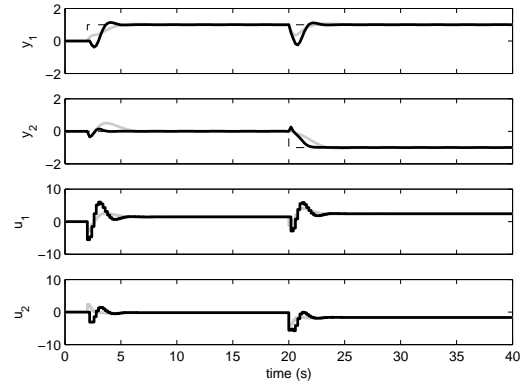
$$(16)$$
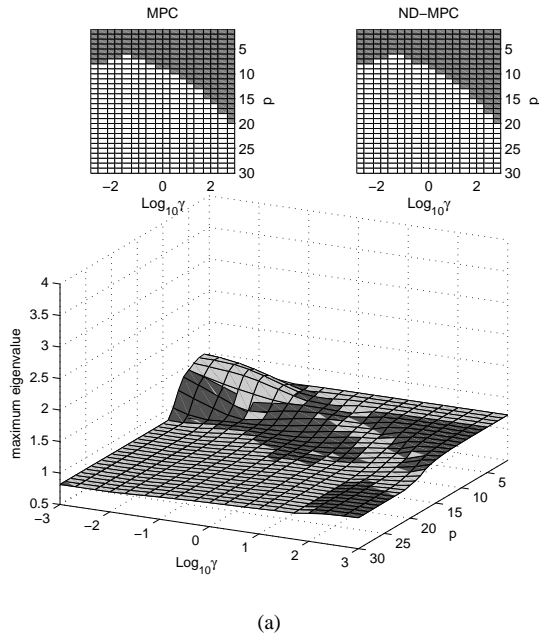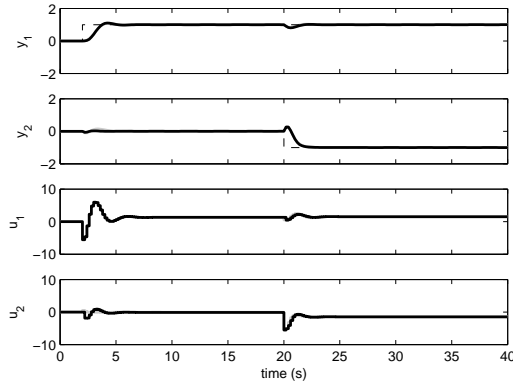
Fig. 1. Stability regions and maximum closed-loop eigenvalues for $\alpha = 0.1$ (a) and control performances with $\gamma = 0.01$ and $p = 20$ (b).

Fig. 2. Stability regions and maximum closed-loop eigenvalues for $\alpha = 0.5$ (a) and control performances with $\gamma = 0.01$ and $p = 20$ (b).

The corresponding discrete-time state-space realization is decomposed in two SISO subsystems. Assuming that input u_1 controls output y_1 and input u_2 controls output y_2, the coefficient α represents a measure of the interactions.

The maximum eigenvalues computed by the dynamic matrices obtained with the previous stated theorems are plotted in the three dimensional graphs of Figures 1(a), 2(a) and 3(a). The Z axis of these plots represents the maximum eigenvalue for the centralized MPC (dark gray surface) and the decentralized MPC with communication (light gray surface). The X and Y axis represent the logarithm of the weight γ and the prediction horizon p, respectively. The corresponding stable region (white surface) is represented in the upside part of these plots both for the centralized and the decentralized case. For each plant, the control performances of ND-MPC (black lines) are compared with the centralized MPC (gray lines) for a given combination of the parameters, as shown in Figures 1(b), 2(b) and 3(b).

In the proposed examples, which are a set within the performed numerical tests, the stability performances depend on the choice of the tuning parameters γ and p. In particular the stability of the closed-loop system is guaranteed for different combinations of the tuning parameters. A wider range of tuning parameters is available for weak interactions whereas a smaller stability region characterizes plants with strong interactions. At the same time, more weak are the interactions, more ND-MPC shows control performances similar to centralized MPC. Often, when MPC is stable, ND-MPC has the same maximum eigenvalues; in some cases ND-MPC seems to be even better.

There are, of course, situations in which the closed loop cannot be stabilized by the proposed decentralized strategy. For example, when the interactions become strong as in Figure 3, ND-MPC cannot stabilize the process (for this reason it is not plotted in Figure 3) whereas MPC provide an acceptable set of stable tuning parameters.
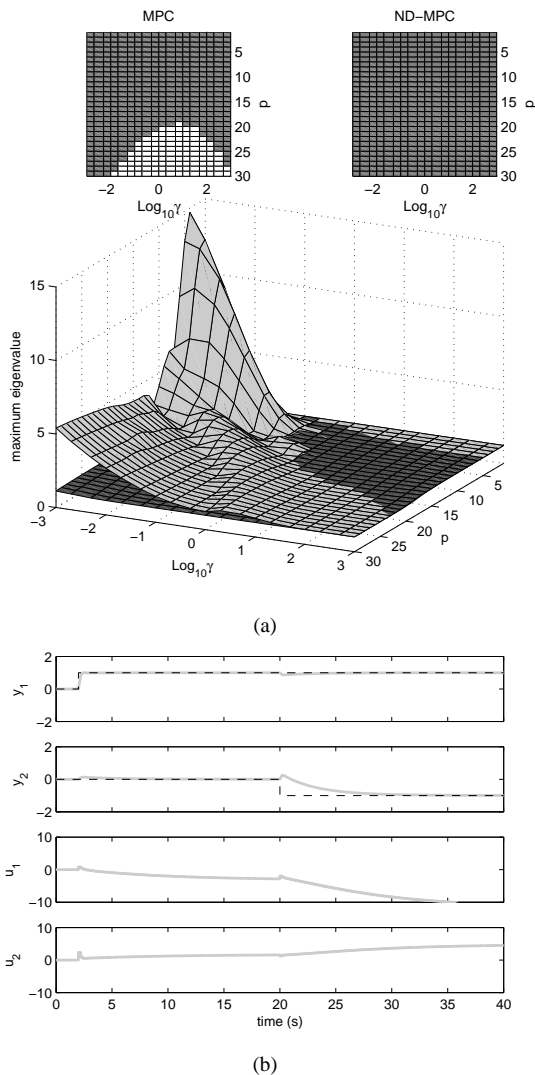
Fig. 3. Stability regions and maximum closed-loop eigenvalues for $\alpha = 2$ (a) and control performances with $\gamma = 0.01$ and $p = 30$ (b).

6. CONCLUSIONS AND FUTURE WORKS

In this paper a stability analysis tool for tuning ND-MPC architectures has been presented. The provided examples show that the proposed ND-MPC can often stabilize the process with a proper choice of the tuning parameters. In these regions the maximum eigenvalues approach that ones of the centralized case. However, in some cases, ND-MPC is not able to stabilize the process and other strategies must be considered.

In this work the unconstrained MPC for linear process with accessible state has been considered. Future works are needed for extending these conclusions in the more general case of constrained MPC. Techniques for obtaining explicit solutions in the constrained case have already been investigated in literature (Tøndel *et al.*, 2003). Therefore, these techniques can potentially be used for extending the proposed approach to the constrained ND-MPC. Furthermore also robustness results (Bemporad and Morari, 1999) with respect to model uncertainties and disturbances

are needed. For instance, a study about the effect that perturbations on the model matrices produce on the closed-loop eigenvalues can be performed.

7. REFERENCES

Bemporad, A. and M. Morari (1999). Robust model predictive control: A survey. In: *Robustness in Identification and Control, Lecture Notes in Control and Information Sciences* (A. Garulli, A. Tesi and A. Vicino, Eds.). Vol. 245. pp. 207–226. Springer. London.

Camponogara, E. (2000). Controlling networks with collaborative nets. PhD thesis. Carnegie Mellon.

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

Cheng, X. and B. H. Krogh (2001). Stability-constrained model predictive control. *IEEE Trans. on Aut. Cont.* **46**, 1816–1820.

El-Farra, N. H., A.Gani and P. D. Christofides (2004). Fault-tolerant control of multi-unit process systems using communication networks. In: *7th IFAC Symposium on Dynamics and Control of Proc. Syst.*. Prague, CZ.

Gudi, Ravindra D., James B. Rawlings, Aswin Venkat and N. Jabbar (2004). Identification for decentralized mpc. In: *DYCOPS*. Boston, MA.

Jia, D. (2003). Distributed coordination in multiagent control systems through model predictive control. PhD thesis. Carnegie Mellon.

Jia, D. and B. H. Krogh (2001). Distributed model predictive control. In: *Proceedings of the A.C.C.*. Vol. 4. pp. 2767–2772.

Katebi, M. R. and M. A. Johnson (1997). Predictive control design for large-scale systems.. *Automatica* **33**, 421–425.

Longhi, S., R. Trillini and M. Vaccarini (2005). Application of a networked decentralized MPC to syngas process in oil industry. In: *16th IFAC World Congress*. Prague, CZ.

Motee, Nader and Bijan Sayyar-Rodsari (2003). Optimal partitioning in distributed model predictive control. In: *Proceedings of the American Control Conference*. Denver, Colorado. pp. 1866–1870.

Tøndel, P., T. A. Johansen and A. Bemporad (2003). An algorithm for multi-parametric quadratic programming and explicit mpc solutions. *Automatica* **39**(3), 489–497.

Vaccarini, M., S. Longhi and M.R. Katebi (2006). State space stability analysis of unconstrained decentralized model predictive control systems. In: *Proceedings of the American Control Conference*. Minneapolis, Minnesota. To be published.

Venkat, A. N., J. B. Rawlings and S. J. Wright (2005). Stability and optimality of distributed model predictive control. CDC-ECC Joint Conference.

Šiljak, D. D. (1996). Decentralized control and computations: status and prospects. *Annual Reviews in Control* **20**, 131–141.