PROOF COVER SHEET

Author(s): Giuseppe Franzè, Walter Lucia and Francesco Tedesco

Article title: A distributed model predictive control scheme for leader–follower multi-agent systems

Article no: 1282178

Enclosures: 1) Query sheet
2) Article proofs

Dear Author,

1. Please check these proofs carefully. It is the responsibility of the corresponding author to check these and approve or amend them. A second proof is not normally provided. Taylor & Francis cannot be held responsible for uncorrected errors, even if introduced during the production process. Once your corrections have been added to the article, it will be considered ready for publication.

Please limit changes at this stage to the correction of errors. You should not make trivial changes, improve prose style, add new material, or delete existing material at this stage. You may be charged if your corrections are excessive (we would not expect corrections to exceed 30 changes).

For detailed guidance on how to check your proofs, please paste this address into a new browser window: http://journalauthors.tandf.co.uk/production/checkingproofs.asp

Your PDF proof file has been enabled so that you can comment on the proof directly using Adobe Acrobat. If you wish to do this, please save the file to your hard disk first. For further information on marking corrections using Acrobat, please paste this address into a new browser window: http://journalauthors.tandf.co.uk/production/acrobat.asp

2. Please review the table of contributors below and confirm that the first and last names are structured correctly and that the authors are listed in the correct order of contribution. This check is to ensure that your name will appear correctly online and when the article is indexed.

Sequence	Prefix	Given name(s)	Surname	Suffix
1		Giuseppe	Franzè	
2		Walter	Lucia	
3		Francesco	Tedesco	

Queries are marked in the margins of the proofs, and you can also click the hyperlinks below.
Content changes made during copy-editing are shown as tracked changes. Inserted text is in red font and revisions have a blue indicator. Changes can also be viewed using the list comments function. To correct the proofs, you should insert or delete text following the instructions below, but **do not add comments to the existing tracked changes**.

AUTHOR QUERIES

General points:

1. **Permissions**: You have warranted that you have secured the necessary written permission from the appropriate copyright owner for the reproduction of any text, illustration, or other material in your article. Please see http://journalauthors.tandf.co.uk/permissions/usingThirdPartyMaterial.asp.
2. **Third-party content**: If there is third-party content in your article, please check that the rightsholder details for re-use are shown correctly.
3. **Affiliation**: The corresponding author is responsible for ensuring that address and email details are correct for all the co-authors. Affiliations given in the article should be the affiliation at the time the research was conducted. Please see http://journalauthors.tandf.co.uk/preparation/writing.asp.
4. **Funding**: Was your research for this article funded by a funding agency? If so, please insert 'This work was supported by <insert the name of the funding agency in full>', followed by the grant number in square brackets '[grant number xxxx]'.
5. **Supplemental data and underlying research materials**: Do you wish to include the location of the underlying research materials (e.g. data, samples or models) for your article? If so, please insert this sentence before the reference section: 'The underlying research materials for this article can be accessed at <full link>/ description of location [author to complete]'. If your article includes supplemental data, the link will also be provided in this paragraph. See <http://journalauthors.tandf.co.uk/preparation/multimedia.asp> for further explanation of supplemental data and underlying research materials.
6. The **CrossRef database** (www.**crossref**.org/) has been used to validate the references. Changes resulting from mismatches are tracked in red font.

Q1. AU: Please check the sentence "Then to ensure that ..." for clarity.
Q2. AU: You have not included an Acknowledgement section. Please either supply one with your corrections or confirm that you do not wish to include one.
Q3. AU: A disclosure statement has been inserted. Please correct if this is inaccurate.
Q4. AU: The CrossRef database (www.crossref.org/) has been used to validate the references. Mismatches between the original manuscript and CrossRef are tracked in red font. Please provide a revision if the change is incorrect. Do not comment on correct changes.
Q5. AU: The reference "Bemporad et al., 2002" is listed in the references list but is not cited in the text. Please either cite the reference or remove it from the references list.
Q6. AU: The reference "Domahid et al., 2012" is listed in the references list but is not cited in the text. Please either cite the reference or remove it from the references list. Also, conference location is missing in this reference.
Q7. AU: Please provide the page range, editor names, publisher name and location for "Franze & Tedesco, 2013" references list entry following journal style http://www.tandf.co.uk/journals/authors/style/reference/tf_APA.pdf.
Q8. AU: Please provide the editor names, publisher name and location for "Franze & Lucia, n.d" references list entry following journal style http://www.tandf.co.uk/journals/authors/style/reference/tf_APA.pdf.
Q9. AU: The reference "Hong et al., 2007" is listed in the references list but is not cited in the text. Please either cite the reference or remove it from the references list.
Q10. AU: Please provide the page range, editor names, publisher name and location for "Kuwata et al., 2005" references list entry following journal style http://www.tandf.co.uk/journals/authors/style/reference/tf_APA.pdf.
Q11. AU: Please provide the editor names, publisher name and location for "Liu et al., 2006" references list entry following journal style http://www.tandf.co.uk/journals/authors/style/reference/tf_APA.pdf.
Q12. AU: Please provide the publisher location for "Necoara et al., 2012" references list entry following journal style http://www.tandf.co.uk/journals/authors/style/reference/tf_APA.pdf.
Q13. AU: Please provide the editor names, publisher name and location for "Snzaier & Damborg, 1987" references list entry following journal style http://www.tandf.co.uk/journals/authors/style/reference/tf_APA.pdf.
Q14. AU: Please provide the editor names, publisher name and location for "Wesselowski & Fierro, 2003" references list entry following journal style http://www.tandf.co.uk/journals/authors/style/reference/tf_APA.pdf.

How to make corrections to your proofs using Adobe Acrobat/Reader

Taylor & Francis offers you a choice of options to help you make corrections to your proofs. Your PDF proof file has been enabled so that you can edit the proof directly using Adobe Acrobat/Reader. This is the simplest and best way for you to ensure that your corrections will be incorporated. If you wish to do this, please follow these instructions:

1. Save the file to your hard disk.
2. Check which version of Adobe Acrobat/Reader you have on your computer. You can do this by clicking on the "Help" tab, and then "About".

If Adobe Reader is not installed, you can get the latest version free from http://get.adobe.com/reader/.

3. If you have Adobe Acrobat/Reader 10 or a later version, click on the "Comment" link at the right-hand side to view the Comments pane.
4. You can then select any text and mark it up for deletion or replacement, or insert new text as needed. Please note that these will clearly be displayed in the Comments pane and secondary annotation is not needed to draw attention to your corrections. If you need to include new sections of text, it is also possible to add a comment to the proofs. To do this, use the Sticky Note tool in the task bar. Please also see our FAQs here: http://journalauthors.tandf.co.uk/production/index.asp.
5. Make sure that you save the file when you close the document before uploading it to CATS using the "Upload File" button on the online correction form. If you have more than one file, please zip them together and then upload the zip file.

If you prefer, you can make your corrections using the CATS online correction form.

Troubleshooting

Acrobat help: http://helpx.adobe.com/acrobat.html
Reader help: http://helpx.adobe.com/reader.html

Please note that full user guides for earlier versions of these programs are available from the Adobe Help pages by clicking on the link "Previous versions" under the "Help and tutorials" heading from the relevant link above. Commenting functionality is available from Adobe Reader 8.0 onwards and from Adobe Acrobat 7.0 onwards.

Firefox users: Firefox's inbuilt PDF Viewer is set to the default; please see the following for instructions on how to use this and download the PDF to your hard drive: http://support.mozilla.org/en-US/kb/view-pdf-files-firefox-without-downloading-them#w_using-a-pdf-reader-plugin

INTERNATIONAL JOURNAL OF CONTROL, 2017
VOL. 0, NO. 0, 1–14
http://dx.doi.org/10.1080/00207179.2017.1282178

Taylor & Francis
Taylor & Francis Group

A distributed model predictive control scheme for leader–follower multi-agent systems

Giuseppe Franzè[a], Walter Lucia[b] and Francesco Tedesco[a]

[a]Dipartimento di Ingegneria Informatica, Modellistica, Elettronica e Sistemistica (DIMES), Università della Calabria, Rende, Italy; [b]Concordia Institute for Information Systems Engineering (CIISE), Concordia University, Montreal, Canada

ABSTRACT

In this paper, we present a novel receding horizon control scheme for solving the formation problem of leader–follower configurations. The algorithm is based on set-theoretic ideas and is tuned for agents described by linear time-invariant systems subject to input and state constraints. The novelty of the proposed framework relies on the capability to jointly use sequences of one-step controllable sets and polyhedral piecewise state-space partitions in order to online apply the 'better' control action in a distributed receding horizon fashion. Moreover, we prove that the design of both robust positively invariant sets and one-step-ahead controllable regions is achieved in a distributed sense. Simulations and numerical comparisons with respect to centralised and local-based strategies are finally performed on a group of mobile robots to demonstrate the effectiveness of the proposed control strategy.

ARTICLE HISTORY
Received 24 June 2016
Accepted 7 January 2017

KEYWORDS
Model predictive control; leader–follower formation; constrained control; set-theoretic control

1. Introduction

In last years, control and coordination of multi-agent network systems have emerged as topics of major interest (Fax & Murray, 2004; Jadbabaie, Lin, & Morse, 2003; Ji, Lin, & Yu, 2015; Lin, Broucke, & Francis, 2004; Liu, Chu, Wang, & Xie, 2006; Mu, Chu, & Wang, 2005; Shi, Wang, & Chu, 2006; Tanner, Pappas, & Kumar, 2004; Wang & Slotine, 2006). This is partly due to broad applications of multi-agent systems in cooperative control of unmanned air vehicles, scheduling of automated highway systems, formation control of satellite clusters and congestion control in communication networks. Along this direction, most of the studies have been greatly inspired by the ubiquitous cooperative behaviour of biological swarms, such as ant colonies, bird flocks and fish schools, where collective motions may emerge from groups of simple individuals through limited interactions. Essentially, a cooperative multi-agent system consists of a set of autonomous agents that interact with one another in a shared environment in order to reach a common goal or to optimise a global performance measure.

Of interest here is the class of distributed model predictive control (DMPC)-based approaches that have been successfully proposed in the last decade (see Christofides, Scattolini, de la Peña, & Liue, 2013) for an extensive discussion and a detailed literature review. In Dunbar (2007), a DMPC scheme for coupled nonlinear systems subject to decoupled constraints was designed. In Magni and Scattolini (2006), a decentralised MPC was proposed for nonlinear systems with no information exchange amongst local controllers and the stability of the decentralised control system was ensured by a set of contractive constraints. In Stewart, Venkat, Rawlings, Wright, and Pannocchia (2010), a cooperative DMPC scheme was developed for linear systems with guaranteed stability of the closed-loop system and convergence of the cost to its optimal value. In Maestre, de la Peña, and Camacho (2011), a game-theory-based DMPC scheme for constrained linear systems was proposed, while in Liu, Chen, de la Peña, and Christofides (2010), a sequential and iterative DMPC architecture was designed for nonlinear systems via Lyapunov-based control techniques. A control scheme for distributed sensing using a leader–follower multi-agent architecture has been presented in Galbusera, Ferrari-Trecate, and R.Scattolini (2013). A relevant feature of this approach relies on the capability to combine hybrid control and MPC techniques with the aim to plan sensing manoeuvres under the satisfaction of input constraints. Moreover, recent contributions (see Necoara, Dumitrache, & Suykens, 2012; Shi et al., 2006; Stewart et al., 2010; Tanner et al., 2004; Wang & Slotine, 2006; Wesselowski & Fierro, 2003) have shown that specific MPC strategies can represent an attractive alternative for control problems where distributed and fast solutions are mandatory. Along these lines, those MPC strategies

CONTACT Walter Lucia walter.lucia@concordia.ca

that move offline most of computations pertaining to the online derivation of the command input are particularly appealing, e.g. the multi-parametric quadratic program (mp-QP) approach of Bemporad, Morari, Dua, and Pistikopoulos (2002) and the ellipsoidal strategies in Angeli, Casavola, Franzè, and Mosca (2008).

In this paper, we will develop a novel distributed discrete-time receding horizon strategy for solving the formation problem of leader–follower networks for agents subject to prescribed state and input constraints. This problem is here attacked by exploiting the basic set-theoretic approach proposed in Angeli et al. (2008) and successfully applied in different contests (see e.g. Franzè & Lucia, 2015, Franzè & Lucia, 2016). Preliminary results of Franzè and Tedesco (2013) and Franzè, Lucia, Tedesco, and Scordamaglia (2014) are here extended to guarantee constraint fulfillment and large domain of attraction. Moreover, a distributed strategy that preserves all the properties pertaining to the basic scheme is achieved by developing a hybrid scheme that combines in a unique framework of the dual-mode MPC strategy of Angeli et al. (2008) and an adequate robust extension of the explicit MPC algorithm of Bemporad et al. (2002). As it will be clarified and proved in the subsequent developments, the latter is required in order to ensure the feasibility retention property and to enlarge as much as possible the domain of attraction.

The main ingredients of the proposed model-based predictive strategy can be summarised as follows:

- Compute pairs of stabilising state-feedback laws and robust positively invariant ellipsoidal sets (for all the subsystems/agents) under the requirement that such regions give rise to a Cartesian product structure.
- Enlarge the sets of initial states that, according to the leader–follower configuration, can be steered to the target in a finite number of steps.
- Compute polyhedral state-space partitions and the corresponding piecewise linear controllers of the domains of attraction pertaining to all the augmented subsystems (current agent + predecessor).
- At each sample time, an online distributed receding horizon strategy is obtained by deriving the smallest region complying with the leader–follower configuration. The control moves are locally computed by minimising a decoupled performance index such that the one-step-ahead state prediction belongs to its successor set.

A key feature of this scheme is its intrinsic capability to move offline most of the required computations so rendering the online needed computational resources significantly modest. Indeed, feasible command inputs can be computed by solving a simple quadratic programming (QP) problem under linear constraints. Differently from the existing competitor strategies, the proposed scheme allows a low degree of communication interactions during the input computation phase because few data exchanges are required amongst the involved agents. Finally, an illustrative example is presented in order to show the benefits of the proposed DMPC strategy. In particular, numerical comparisons with respect to centralised and decentralised benchmark schemes are provided in terms of achievable domains of attraction and control performance to clearly put in light the advantages of the derived control scheme.

Notations and preliminaries

Definition 1.1: Given a set $S \subseteq \mathbb{R}^n$, $In[S] \subseteq S$ denotes its inner ellipsoidal approximation.

Definition 1.2: Given a set $S \subseteq X \times Y \subseteq \mathbb{R}^n \times \mathbb{R}^m$, the projection of the set S onto X is defined as $\mathrm{Proj}_X(S) := \{x \in X \mid \exists y \in Y \text{ s.t.} (x, y) \in S\}$.

Let us consider the following discrete-time plant description:

$$x_p(t+1) = \Phi x_p(t) + G u(t) \tag{1}$$

where $t \in \mathbb{Z}_+ := \{0, 1, \ldots\}$, $x_p(t) \in \mathbb{R}^n$ denotes the plant state and $u(t) \in \mathbb{R}^m$ the control input. Moreover, the system (1) is subject to the following set-membership state and input constraints:

$$u(t) \in \mathcal{U}, \ \forall t \geq 0, \mathcal{U} := \{u \in \mathbb{R}^m \mid u^T u \leq \bar{u}\}, \tag{2}$$

$$x_p(t) \in \mathcal{X}, \ \forall t \geq 0, \mathcal{X} := \left\{x_p \in \mathbb{R}^n \mid x_p^T x_p \leq \bar{x}_p\right\}, \tag{3}$$

with $\bar{u} > 0$, $\bar{x}_p > 0$, and $\mathcal{U}$, $\mathcal{X}$ compact subsets of $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively.

Definition 1.3: A set $\Xi \subseteq \mathbb{R}^n$ is robustly positively invariant for (1) if once the state $x_p(t)$ enters that set at any given time t_0, it remains inside for all future instants, i.e. $x_p(t_0) \in \Xi \rightarrow x_p(t) \in \Xi, \forall t \geq t_0$.

Moreover, it is possible to compute the sets of states i-step controllable to Ξ via the following recursion (see Blanchini & Miani, 2008):

$$\begin{aligned} &\Xi_0 := \Xi \\ &\Xi_i := \{x_p \in \mathcal{X} : \exists\, u \in \mathcal{U} : \Phi x_p + G u \in \Xi_{i-1}\}, \end{aligned} \tag{4}$$

where Ξ_i is the set of states that can be steered into Ξ_{i-1} using a single move with a causal control. By induction, we have that Ξ_i is the set of states that can be steered into Ξ in at most i control moves.

With

$$x_p^+ = \Phi x_p + Gu,$$

we denote any element of the state space $\mathbb{R}^n$ achieved by applying an admissible input $u \in \mathcal{U}$ along the one-step-ahead state evolution law.

2. Problem formulation

In this paper, we focus on large-scale systems of leader–follower type. As it is well known, leader–follower system architectures imply that each subsystem, from the second one onwards, is influenced by the previous agent. We consider leader–follower linear time-invariant (LTI) systems, for which the dynamics of the first subsystem is

$$x^1(t+1) = A^1 x^1(t) + B^1 u^1(t), \tag{5}$$

while the remaining agents are described by the following dynamical equations:

$$\begin{aligned} x^i(t+1) = {} & A^i x^i(t) + B^i u^i(t) + A^{i,i-1} x^{i-1}(t) \\ & + B^{i,i-1} u^{i-1}(t), \quad i = 2, \dots, l, \end{aligned} \tag{6}$$

where $x^i \in \mathbb{R}^{n_i}$ and $u^i \in \mathbb{R}^{m_i}$ are the state and input vectors of the ith subsystem, $A^i \in \mathbb{R}^{n_i \times n_i}$ and $B^i \in \mathbb{R}^{n_i \times m_i}$ the state and input matrices for the ith subsystem, while $A^{i,i-1} \in \mathbb{R}^{n_i \times n_{i-1}}$ and $B^{i,i-1} \in \mathbb{R}^{n_i \times m_{i-1}}$ are the coupling matrices which define the influence of the $(i-1)$th subsystem upon the ith one. Moreover, the following input and state constraints are prescribed for each ith subsystem:

$$u^i(t) \in \mathcal{U}^i, \ \forall t \geq 0, \mathcal{U}^i := \{u^i \in \mathbb{R}^{m_i} \,|\, u^{i^T} u^i \leq \bar{u}^i\}, \tag{7}$$

$$x^i(t) \in \mathcal{X}^i, \ \forall t \geq 0, \mathcal{X}^i := \{x^i \in \mathbb{R}^{n_i} \,|\, x^{i^T} x^i \leq \bar{x}^i\}, \tag{8}$$

with $\bar{u}^i > 0$, $\bar{x}^i > 0$, and $\mathcal{U}^i$, $\mathcal{X}^i$ compact subsets of $\mathbb{R}^m$ and $\mathbb{R}^n$, respectively.

Assumption 2.1: *Without loss of generality, we shall consider leader–follower networks whose agents have the same state-space and input dimensions, n and m, respectively, i.e.* $n_1 = n_2 = \cdots = n_l = n$ *and* $m_1 = m_2 = \cdots = m_l = m$

Assumption 2.2: (***Communication facilities***): *At each time instant t, each ith follower subsystem knows the current state and input values of the* $(i-1)$*th subsystem.*

Then, the problem we want to solve can be stated as follows.

Distributed control problem for leader–follower networks (DC-LF-N): *Given the leader and followers plant models (5) and (6), determine a distributed state-feedback control policy*

$$\begin{aligned} &u^1(t) = g(x^1(t)), \\ &u^i(t) = g(x^i(t), x^{i-1}(t), u^{i-1}(t)), \quad i = 2, \dots, l, \end{aligned} \tag{9}$$

compatible with (7) and (8), such that starting from an admissible initial condition $x(0) = [x^{1^T}(0), x^{2^T}(0), \dots x^{l^T}(0)]^T$, *the state trajectory of each ith agent is asymptotically driven to* 0_n

In the sequel, the problem will be addressed by means of the dual-mode receding horizon control approach proposed in Angeli et al. (2008) which prescribes the computation of (1) a robust stabilising state-feedback control law and the corresponding ellipsoidal region; (2) a sequence of one-step-state-ahead controllable sets.

In order to exploit the ideas of Angeli et al. (2008) for dealing with the *DC-LF-N* problem, the following key question must be analysed:

How can one define terminal ellipsoidal regions and sequences of one-step controllable sets for each subsystem within a distributed framework such that there exist feasible commands compatible with the dynamics (5) and (6)?

The next section will provide a solution to this relevant issue.

3. Robustly positively invariant regions and one-step controllable set sequences

The aim of this section is to characterise terminal pairs, i.e. stabilising controllers (K^i) and ellipsoidal regions (Ξ_0^i), and one-step controllable sets according to the *DC-LF-N* problem prescriptions.

3.1 Terminal sets

Due to the distributed nature of the proposed framework, it is necessary to find a terminal constraint set which preserves the following Cartesian structure:

$$\Xi_0 := \prod_{i=1}^{l} \Xi_0^i \tag{10}$$

where $\Xi_0 \subset \mathbb{R}^{nl}$ is in principle the terminal set of the centralised system achievable by using (5) and (6).

Then, the simplest way to design terminal pairs complying with (10) is to initially compute the pair $(K^1, \ \Xi_0^1)$ pertaining to the leader because its dynamics (5) does not depend on the other subsystem behaviours. Hence, the remaining pairs $(K^i, \ \Xi_0^i)$, $i = 2, \dots, l$, are determined by resorting to a 'worst-case' approach.

Specifically, the leader terminal pair (K^1, Ξ_0^1) is obtained by solving a standard semi-definite programming (SDP) problem as outlined, for example, in Kothare, Balakrishnan, and Morari (1996). While the follower pairs (K^i, Ξ_0^i), $i = 2, \ldots, l$, require different arguments because of the coupled terms $A^{i,\,i-1}x^{i-1}$ and $B^{i,\,i-1}u^{i-1}$. Therefore, for each follower of the ith subsystem, $A^{i,\,i-1}x^{i-1}$ and $B^{i,\,i-1}u^{i-1}$ can be considered as unknown bounded disturbances in view of terminal set computed for the $(i-1)$th agent, i.e. $x^{i-1} \in \Xi_0^{i-1}$. As a consequence, the robust positively invariant ellipsoid Ξ_0^i can be obtained by resorting to SDP procedures based on P-difference arguments (see Kurzhanski & Valyi, 1997).

3.2 One-step controllable sets

Here we determine, for each ith agent, a family of one-step controllable sets Ξ_j^i to the target sets Ξ_0^i, $i = 1, \ldots, l$ by carefully taking care that the one-step state predictions are evaluated along interacting subsystem models.

Therefore, in order to derive one-step controllable regions which ensure at each time instant admissible solutions to the *DC-LF-N* problem, the following issues have to be taken into account:

(1) For each ith agent, the local pair $(x^i(t), u^i(t))$ must be compatible with the $(i+1)$th agent dynamics, i.e.

$$x^{i+1}(t+1) = A^{i+1}x^{i+1}(t) + B^{i+1}u^{i+1}(t) + A^{i+1,i}x^i(t) + B^{i+1,i}u^i(t).$$

(2) The ith sequence $\{\Xi_j^i\}$ should be built such that each element Ξ_j^i guarantees that the one-step state evolution

$$x^{i^+} = A^i x^i + B^i u^i + A^{i,i-1}x^{i-1} + B^{i,i-1}u^{i-1},\ x^i \in \Xi_j^i \tag{11}$$

is at least confined within Ξ_j^i.

Point (1) first prescribes that the controllable sequences have to be computed by resorting to the following extended space description $x_{\text{aug}}^i = [x^{i^T}, u^{i^T}, x^{i-1^T}, u^{i-1^T}]^T$ and, therefore, from now on they will be named as $\{\mathcal{T}_j^i\}$, $i = 1, \ldots, l$.

Then, it is necessary to proceed with the set construction in 'backwards': the l set sequences are achieved level-by-level and starting from the last element of the leader-follower architecture, i.e. the lth follower subsystem (6). In this way, it can be ensured that each one-step controllable region $\mathcal{T}_j^i$ is compatible with the set $\mathcal{T}_j^{i-1}$ pertaining to the $(i-1)$th subsystem whose dynamics is shared with the ith agent.

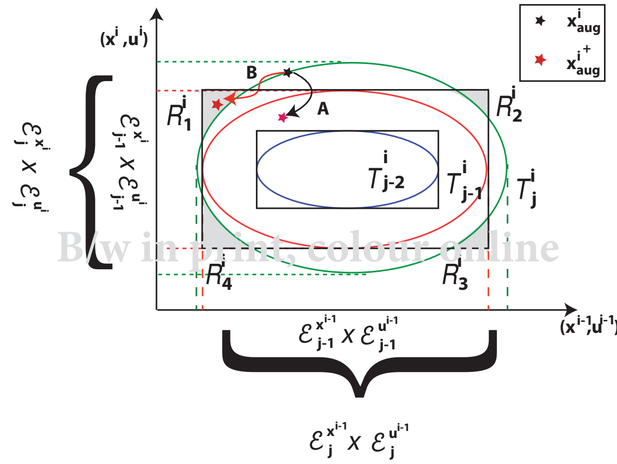

Figure 1. One-step controllable set construction.

One of the main consequences of the above arguments is that (11) becomes

$$x^{i^+} = A^i x^i + B^i u^i + A^{i,i-1}x^{i-1} + B^{i,i-1}u^{i-1},\ \ x^i \in \mathcal{E}_j^{x^i}, \tag{12}$$

with

$$\mathcal{E}_j^{x^i} := \text{Proj}_{x^i}\{\mathcal{T}_j^i\} \tag{13}$$

in place of Ξ_j^i.

For the sake of clarity, the illustrative example of Figure 1 is provided. There, the following admissible scenarios occur:

(A) The one-step evolution $x_{\text{aug}}^{i^+}$ belongs to $\mathcal{T}_j^i$.

(B) $x_{\text{aug}}^{i^+}$ remains confined into $\mathcal{E}_{j-1}^{x^i} \times \mathcal{E}_{j-1}^{u^i} \times \mathcal{E}_{j-1}^{x^{i-1}} \times \mathcal{E}_{j-1}^{u^{i-1}}$.

Since the one-step controllable sets are derived under (12), the set-membership $x^{i^+} \in \mathcal{E}_j^{x^i}$ does not necessarily translate into $x_{\text{aug}}^{i^+} \in \mathcal{T}_j^i$. This is depicted in Figure 1 when the scenario **B** is considered. The main consequence of this unfavourable situation is that the grey zones could lead to loss of feasibility: in fact by iterating this reasoning at a certain time instant, it is no longer ensured the existence of an admissible command input because the augmented state $x_{\text{aug}}^{i^+}$ is outside the domain of attraction $\mathcal{T}_N^i$. To overcome such a drawback, the idea would be to build the sequences $\{\mathcal{T}_j^i\}$, $i = 1, \ldots, l$, by imposing for each jth iteration of the following additional constraint (see the polyhedron boxing $\mathcal{T}_{j-2}^i$ and contained in $\mathcal{T}_{j-1}^i$ in Figure 1):

$$\mathcal{E}_{j-1}^{x^i} \times \mathcal{E}_{j-1}^{u^i} \times \mathcal{E}_{j-1}^{x^{i-1}} \times \mathcal{E}_{j-1}^{u^{i-1}} \subseteq \mathcal{T}_j^i, \qquad i = 1, \ldots, l, \tag{14}$$

where

$$\mathcal{E}_j^{u^i} := \mathrm{Proj}_{u^i}\{\mathcal{T}_j^i\}. \tag{15}$$

In principle, the construction of the l controllable sequences should take care of (14) at each step of recursions (4), but this leads to conservative results because (14) essentially is a *stopping criterion* during the computation of $\mathcal{T}_j^i$. A way to mitigate such a disadvantage is to proceed as follows:

(1) Compute the one-step controllable sequence $\{\mathcal{T}_j^i\}_{j=1}^{N_i}$ without taking into account (14) and with N_i the saturation level on the sequence growth (see Theorem 3.1).

(2) If

$$\mathcal{E}_{N_i-1}^{x^i} \times \mathcal{E}_{N_i-1}^{u^i} \times \mathcal{E}_{N_i-1}^{x^{i-1}} \times \mathcal{E}_{N_i-1}^{u^{i-1}} \nsubseteq \mathcal{T}_{N_i}^i, \tag{16}$$

then compute (e.g. the grey regions of Figure 1)

$$\bigcup_s \mathcal{R}_s^i := \left(\mathcal{E}_{N_i-1}^{x^i} \times \mathcal{E}_{N_i-1}^{u^i} \times \mathcal{E}_{N_i-1}^{x^{i-1}} \times \mathcal{E}_{N_i-1}^{u^{i-1}}\right) \setminus \mathcal{T}_{N_i}^i. \tag{17}$$

(3) Determine *local* controllers for the external regions $\mathcal{R}_s^i$ capable to achieve constraints satisfaction and to ensure that there exists a finite time $t^i < \infty$ such that

$$x_{\mathrm{aug}}^i(t + t^i) \in \mathcal{T}_{N_i}^i.$$

The idea behind the exploitation of points (1)–(3) relies on the fact that the sequences $\{\mathcal{T}_j^i\}_{j=1}^{N_i}$, $i = 1, \ldots, l$, can be build under the following nesting condition:

$$\mathcal{T}_{j-1}^i \subset \mathcal{T}_j^i \tag{18}$$

Therefore, even if at each step the requirement (14) does not hold true, this does not compromise the admissibility thanking to (18). The only hitch could rise at the saturation level N_i where it is necessary to ensure the existence of feasible command inputs also for (17). This can be ensured by first considering that along the ith one-step evolution,

$$\begin{aligned} x^i(t+1) = A^i x^i(t) + B^i u^i(t) + A^{i,i-1} x^{i-1}(t) \\ + B^{i,i-1} u^{i-1}(t) \end{aligned}$$

the term $A^{i,\,i-1}x^{i-1}(t) + B^{i,\,i-1}u^{i-1}(t)$ assumes the role of an exogenous disturbance with $x^{i-1} \in \mathcal{E}_{N_i-1}^{x^{i-1}}$ and $u^{i-1}(t) \in \mathcal{U}^{i-1}$. Then, a possible way to proceed is to resort the well-reputed approach developed in Bemporad et al. (2002) and extended to the robust case in, for example, Sakizlis et al. (2004) and Gao and Chong (2012). There, at each time instant t, the following receding horizon optimal control problem is formulated:

$$\min_{\mathbf{U}^i} \max_{\mathbf{D}^i} J(\mathbf{U}^i, x^i(t)), \tag{19}$$

such that

$$\begin{cases} x^i(t+k|t) \in \mathcal{X}^i,\ u^i(t+k|t) \in \mathcal{U}^i,\ k = 1, \ldots, N, \\ x^i(t|t) = x^i(t), \\ x^i(t+k|t) = A^i x^i(t+k|t) + B^i u^i(t+k|t) + \\ W^i d^i(t+k|t),\ k \geq 0, \\ u^i(t+k|t) = K_{LQR}^i x^i(t+k|t),\ k \geq N, \end{cases} \tag{20}$$

where

- $J(\mathbf{U}^i, x(t)) \triangleq x^{i^T}(t+N|t) P^i x^i(t+N|t) + \sum_{k=0}^{N-1} \times x^{i^T}(t+k|t) Q^i x^i(t+k|t) + u^{i^T}(t+k|t) R^i u^i(t+k|t)$;
- $\mathbf{U}^i \triangleq [u^{i^T}(t|t) \cdots u^{i^T}(t+N-1|t)]^T$;
- $W^i = [A^{i,\,i-1}\ B^{i,\,i-1}]$ and $d^i(t+k|t) := [x^{(i-1)^T}(t+k|t)\ u^{(i-1)^T}(t+k|t)]^T \in \mathcal{E}_{N_{i-1}}^{x^{i-1}} \times \mathcal{U}^{i-1}$;
- $\mathbf{D}^i \triangleq [d^{i^T}(t|t) \cdots d^{i^T}(t+N-1|t)]^T$;
- $x^i(t+k|t)$ denotes the predicted state vector at $t+k$ obtained by applying the input sequence $\{u^i(t|t), \ldots, u^i(t+k-1|t)\}$ to (6) starting from $x^i(t|t)$;
- $R^i = R^{i^T} > 0$, $Q^i = Q^{i^T} \geq 0$, $P^i = P^{i^T} \geq 0$;
- K_{LQR}^i is the feedback gain obtained jointly with the matrix P^i as the solution of the unconstrained infinite-horizon linear quadratic regulator problem with weights Q^i and R^i, see e.g. Snzaier and Damborg (1987);
- N the prediction horizon.

The optimisation (19)–(20) can be recast as an *mp-QP*, whose solution consists of a set of affine control functions in terms of states and set of regions where these functions are valid. Specifically, one computes piecewise linear, continuous and explicit state-feedback controllers of the following form:

$$u(x^{i^*}) = F_k^i x^{i^*} + g_k^i, \ \text{ if } H_k^i x^{i^*} \leq b_k^i,\ k = 1, \ldots, N_p^i, \tag{21}$$

while the corresponding N_p^i validity regions are obtained as the polyhedral partition of a suitable subset $\mathcal{P}^i$ of the parameter set $\mathcal{X}^i$, with $\mathcal{P}^i$ the minimal inner polyhedral approximation of $\mathcal{X}^i$ such that

$$\bigcup_s \mathrm{Proj}_{x^i}\{\mathcal{R}_s^i\} \subseteq \mathcal{P}^i. \tag{22}$$

For more technical details on the *mp-QP* approach, the interested reader can refer to Bemporad et al. (2002) and Sakizlis et al. (2004).

Remark 3.1: Notice that the polyhedral partition arising from the requirement (22) can be made less conservative by exploiting the following arguments:

- Group the sets such that $\mathcal{R}_s^i \cap \mathcal{R}_r^i \neq \emptyset$, $\forall s \neq r$. Compute the union set $\mathcal{M}_u^i$ of such regions.
- Determine the closest (in the 2-norm sense) equilibria $\bar{x}^i_{\text{aug}^u} \in \mathcal{T}^i_{N_i}$ to each $\mathcal{M}^i_u$.
- Compute polyhedral partitions and related sequences of explicit state-feedback controllers by using as targets the equilibria $\bar{x}^i_{\text{aug}^u}$.

The following result summarises the above developments.

Theorem 3.1: *Let $\mathcal{T}_0^i \neq \emptyset$, $i = 1, \ldots, l$ be given robustly invariant ellipsoidal regions complying with (7)–(8). Let $x^1_{aug} = [x^{1^T}, u^{1^T}, x^{1^T}, u^{1^T}]^T \in \mathbb{R}^{2n+2m}$ and $x^i_{aug} = [x^{i^T}, u^{i^T}, x^{i-1^T}, u^{i-1^T}]^T \in \mathbb{R}^{2n+2m}$, $i = 2, \ldots, l$, the augmented state spaces describing the dynamics of subsystems (5) and (6), respectively. Then, the one-step controllable sets sequence $\{\mathcal{T}_j^i\}$, $i = 1, \ldots, l$, are obtained by means of the following recursions:*

$$\mathcal{T}_j^l = \{x^l_{aug} \in \mathcal{X}^l \times \mathcal{U}^l \times \mathcal{X}^{l-1} \times \mathcal{U}^{l-1} : A^l x^l + B^l u^l + A^{l,l-1}x^{l-1} + B^{l,l-1}u^l \in \mathcal{E}^{x^l}_{j-1}\} \tag{23}$$

$$\mathcal{T}_j^i = \{x^i_{aug} \in \mathcal{T}_j^{i+1} \cap (\mathcal{X}^i \times \mathcal{U}^i \times \mathcal{X}^{i-1} \times \mathcal{U}^{i-1}) : A^i x^i + B^i u^i + A^{i,i-1}x^{i-1} + B^{i,i-1}u^{i-1} \in \mathcal{E}^{x^i}_{j-1}\}, \quad i = l-1, \ldots, 2; \tag{24}$$

$$\mathcal{T}_j^1 = \{x^1_{aug} \in \mathcal{T}_j^2 \cap (\mathcal{X}^1 \times \mathcal{U}^1 \times \mathcal{X}^1 \times \mathcal{U}^1) : A^1 x^1 + B^1 u^1 \in \mathcal{E}^{x^1}_{j-1}\} \tag{25}$$

$$\mathcal{T}_{j-1}^i \subset \mathcal{T}_j^i, \ \forall j, \ \text{and } i = 1, \ldots, l. \tag{26}$$

Moreover, let

- *$\mathcal{P}^i$, $i = 1, \ldots, l$ be the polyhedra satisfying (22):*
-

$$x^i(t+1) = A^i x^i(t) + B^i u^i(t) + A^{i,i-1}x^{i-1}(t) + B^{i,i-1}u^{i-1}(t), \quad \forall x^{i-1}(t) \in \mathcal{E}^{x^{i-1}}_{N_{i-1}}, \forall u^{i-1}(t) \in \mathcal{U}^i, i = 1, \ldots, l, \tag{27}$$

  the state evolution law for each agent with $A^{i,\,i-1}x^{i-1}(t) + B^{i,\,i-1}u^{i-1}(t)$ bounded exogenous disturbances;
- *$\mathcal{S}_k^i := \{H_k^i x^i \leq b_k^i\} \subset \mathcal{P}^i$, $k = 1, \ldots, N_p^i$, $i = 1, \ldots, l$, the polyhedral regions defining the partition of $\mathcal{P}^i$, $i = 1, \ldots, l$;*
-

$$u(x^i) = F_k^i x^i + g_k^i, \forall x^i \in \{H_k^i x^i \leq b_k^i\}, \quad k = 1, \ldots, N_p^i, \ i = 1, \ldots, l, \tag{28}$$

  continuous piecewise affine LQ controllers with $F_k^i \in \mathbb{R}^{m\times n}$ and $g_k^i \in \mathbb{R}^{m\times 1}$;
-

$$k^i = \arg\max_j \{j \,|\, \mathcal{E}_j^{x^i} \times \mathcal{E}_j^{u^i} \times \mathcal{E}_j^{x^{i-1}} \times \mathcal{E}_j^{u^{i-1}} \subseteq \mathcal{T}^i_{N_i}\}, \quad i = 1, \ldots, l. \tag{29}$$

Then, each state $[x^{1^T} \, x^{2^T} \cdots x^{l^T}]^T \in \mathcal{E}_j^{x^1} \times \mathcal{E}_j^{x^2} \times \cdots \times \mathcal{E}_j^{x^l}$ will be confined into $\mathcal{E}_{k^1}^{x^1} \times \mathcal{E}_{k^2}^{x^2} \times \cdots \times \mathcal{E}_{k^l}^{x^l}$ in a finite number of steps.

Proof: Because each ith follower dynamics (6) depends on the current state pertaining to the $(i-1)$th subsystem until that the leader dynamics (5) is considered, we have to ensure that each admissible state belonging to $\mathcal{T}_j^i$ is also admissible for the construction of the one-step-ahead region $\mathcal{T}_j^{i+1}$ related to the $(i+1)$th subsystem. To comply with this idea the simplest, though not optimal, way is to start the construction of the sequences family $\{\mathcal{T}_j^i\}$, $i = 1, \ldots, l$, from the last subsystem, i.e. lth follower. Then, the $(l-1)$th subsystem of one-step controllable set (same level, i.e. j) is computed under the set-membership requirement that the quadruple $(x_{l-1}, u_{l-1}, x_{l-2}, u_{l-2})$ belongs to $\mathcal{T}_j^l$ (Equation (24)) which makes compatible the $(l-1)$th subsystem dynamics (6) with the lth one-step controllable sequence. By induction, the same arguments apply to the successive agents.

As the second part of the theorem is concerned, the following arguments can be exploited. If at the saturation level N_i the condition (16) holds true, the non-empty, non-convex region (17) comes out, there exists k_i such that (29) is valid and the piecewise partition of the polyhedron $\mathcal{P}^i$, covering $\bigcup_s \text{Proj}_{x^i}\{\mathcal{R}_s^i\}$, is determined by referring to the local dynamics (27). Then, the jointly action due to controllers (28) and to the contraction property (14) allows to ensure that whatever is the current overall state

$$[x^{1^T}(t)\, x^{2^T}(t) \cdots x^{l^T}(t)]^T \in \mathcal{E}_j^{x^1} \times \mathcal{E}_j^{x^2} \times \cdots \times \mathcal{E}_j^{x^l},$$

the state trajectory is driven by construction in a decentralised fashion and in a finite number of steps inside $\mathcal{E}_{k^1}^{x^1} \times \mathcal{E}_{k^2}^{x^2} \times \cdots \times \mathcal{E}_{k^l}^{x^l}$. ■

4. The distributed model predictive control algorithm

The results of the previous section are the key ingredients for developing a new distributed sequential control strategy whose main features are the simplicity in the design phase and low communication requirements. Specifically, the proposed scheme will rely on the properties of the leader–follower hierarchy in the sense that each ith agent makes a decision just before the successive $(i+1)$th agent.

To this end, two issues have to be carefully investigated: (1) local command input computation; (2) checking augmented set-membership state.

4.1 Local command input computation

The ith subsystem selects its local command u^i by resorting to the current state measurement $x^i(t)$ and to the information ($x^{i-1}(t)$ and $u^{i-1}(t)$) received from the $(i-1)$th predecessor agent. Then, the local input $u^i(t)$ is computed according to the following optimisation problem:

$$u^i(t) = \arg\min_{u^i} J_{j(t)}(x^i(t), u^i, x^{i-1}(t), u^{i-1}(t)) \tag{30}$$

subject to

$$\begin{aligned} &A^i x^i(t) + B^i u^i + A^{i,i-1} x^{i-1}(t) \\ &\quad + B^{i,i-1} u^{l-1}(t) \in \mathcal{E}^{x^i}_{j(t)-1},\ u^i \in \mathcal{U}^i \end{aligned} \tag{31}$$

Here, the running cost $J_{j(t)}(x^i(t), u^i, x^{i-1}(t), u^{i-1}(t))$ is chosen without loss of generality as follows:

$$\begin{aligned} &J_{j(t)}(x^i(t), u^i, x^{i-1}(t), u^{i-1}(t)) \triangleq \\ &\|A^i x^i(t) + B^i u^i + A^{i,i-1} x^{i-1}(t) + B^{i,i-1} u^{i-1}(t)\|^2_{(P^i_{j(t)-1})^{-1}} \end{aligned} \tag{32}$$

where $P^i_{j(t)-1}$ is the shaping matrix of the ellipsoidal region $\mathcal{E}^{x^i}_{j(t)} = \{x^i \in \mathbb{R}^{n_i} \mid (x^i)^T (P^i_{j(t)-1})^{-1} x^i \leq 1\}$.

Proposition 4.1: *The optimisation problem (30)–(31) can be solved by the following SDP problem:*

$$u^i(t) = \arg\min_{u^i} \gamma_i \tag{33}$$

subject to

$$\begin{bmatrix} \gamma_i & (x^{i^+})^T \\ * & P^i_{j(t)-1} \end{bmatrix} \leq 1, \begin{bmatrix} \bar{u} & (u^i)^T \\ * & I \end{bmatrix} \leq 1. \tag{34}$$

Proof: The optimisation problem (30)–(31) can be rewritten as follows:

$$u^i(t) = \arg\min_{u^i} \gamma_i \tag{35}$$

subject to

$$\begin{cases} (x^{i^+})^T (P^i_{j(t)-1})^{-1} (x^{i^+}) \leq \gamma_i, & \gamma_i \leq 1 \\ (u^i)^T (u^i) \leq \bar{u}. \end{cases} \tag{36}$$

By means of Schur complements, (33)–(34) come out. ■

Slight differences arise when the leader subsystem (5) is concerned. In fact, the leader does not make use of any information related to the other agents and determines the command $u^1(t)$ by only using the state measurements $x^1(t)$. As a consequence, the following optimisation problem results:

$$u^1(t) = \arg\min_{u^1} J_{j(t)}(x^1(t), u^1) \tag{37}$$

subject to

$$A^1 x^1(t) + B^1 u^1 \in \mathcal{E}^{x^1}_{j(t)-1},\ u^1 \in \mathcal{U}^1 \tag{38}$$

where

$$J_{j(t)}(x^1(t), u^1) \triangleq \|A^1 x^1 + B^1 u^1\|^2_{(P^1_{j(t)-1})^{-1}} \tag{39}$$

Proposition 4.2: *The optimisation problem (37)–(38) can be solved by the following SDP problem:*

$$u^1(t) = \arg\min_{u^1} \gamma_1 \tag{40}$$

subject to

$$\begin{bmatrix} \gamma_1 & (A^1 x^1(t) + B^1 u^1)^T \\ * & P^1_{j(t)-1} \end{bmatrix} \leq 1, \begin{bmatrix} \bar{u} & (u^1)^T \\ * & I \end{bmatrix} \leq 1. \tag{41}$$

Proof: By using the same arguments of Proposition 4.1. ■

4.2 Augmented set-membership state

One of the tricky aspects pointed out in the developments of Section 3 and, more specifically, in Theorem 3.1, is that the set-membership state must be checked on the augmented state-space $x^i_{\text{aug}} = [x^{i^T}, u^{i^T}, x^{i-1^T}, u^{i-1^T}]^T$ in order to ensure a correct control move selection.

To this aim, the main difficulty relies on the fact that during the online operations for each ith agent, only the triplet $(x^i(t), x^{i-1}(t), u^{i-1}(t))$ is available; therefore, the set-membership is subject to the solution of the optimisation (30)–(31) (respectively, (37)–(38) for the leader).

As a consequence in order to check the set-membership state, the local command computation $u^i(t)$ must be performed.

Then to ensure that at each time instant a feasible input $u^i(t)$ there always exists an admissible, though not optimal, way to solve the SDP (30)–(31) (resp. (37)–(38)) by considering as the projected set-membership region the saturation level region $\mathcal{E}^{x^i}_{N_i-1}$ (resp. $\mathcal{E}^{x^1}_{N_1-1}$) irrespective of which is the current $\mathcal{E}^{x^i}_{j(t)-1}$ (resp. $\mathcal{E}^{x^1}_{j(t)-1}$). This optimisation can lead to the following different scenarios:

M1: If a solution exists, then $x^i_{\text{aug}}(t) \in \mathcal{T}^i_{N_i-1}$ (resp. $\mathcal{T}^i_{N_1-1}$);

M2: else $x^i_{\text{aug}}(t) \in \bigcup_s \mathcal{R}^i_s$

Once this feasible command has been obtained (namely **M1**), the solution could be improved as follows. Let $\bar{j}$ be the minimum integer such that $x^1(t) \in \mathcal{E}^{x^1}_{\bar{j}}$, the optimisation (30)–(31) (resp. (37)–(38)) is first performed with $\mathcal{E}^{x^i}_{\bar{j}}$ (resp. $\mathcal{E}^{x^1}_{\bar{j}}$), then a bisection search amongst the initial and final ellipsoid extremes $\mathcal{E}^{x^i}_{\bar{j}+1}$ and $\mathcal{E}^{x^i}_{N_i-2}$ (resp. $\mathcal{E}^{x^1}_{\bar{j}+1}$ and $\mathcal{E}^{x^1}_{N_1-2}$) is adopted.

The following procedure summarises this reasoning:

Quasi-bisection optimisation search (QBOS)

AFTC-RHC-Algorithm

INPUT: ΔT_s, t_{curr}, i, N_i, $\bar{j}$
OUTPUT: u^{i^*}

1: **solve** (30)–(31) (resp. (37)–(38)) with $\mathcal{E}^{x^i}_{N_i-1}$ (resp. $\mathcal{E}^{x^1}_{N_1-1}$)
2: **store** u^{i^*} ▷ solution of Step 1;
3: $j_{\text{up}} \leftarrow N_1 - 2$
4: $j_{\text{down}} \leftarrow \bar{j} + 1$
5: $j_{\text{temp}} \leftarrow \lfloor \frac{j_{\text{up}}+j_{\text{down}}}{2} \rfloor$;
6: **if** (30)–(31) (resp. (37)–(38)) with $\mathcal{E}^{x^i}_{j_{\text{temp}}}$ (resp. $\mathcal{E}^{x^1}_{j_{\text{temp}}}$) has a solution $\bar{u}^i$, **then**
7: $u^{i^*} \leftarrow \bar{u}^i$;
8: $j_{\text{up}} \leftarrow j_{\text{temp}}$
9: **else**
10: $j_{\text{down}} \leftarrow j_{\text{temp}}$
11: **end if**
12: **if** $t_{\text{curr}} < \Delta T_s$ and $j_{\text{temp}} > 0$ **then**
13: **goto** Step 5.
14: **else**
15: **return** u^{i^*}
16: **end if**

4.3 The leader–follower sequential MPC algorithm

The proposed distributed MPC scheme is based on the fact that at each time instant, the following actions will be undertaken by leader and follower agents in a sequential fashion:

- First, the leader computes the command input $u^1(t)$ and transmits the pair $(x^1(t), u^1(t))$ to the agent 2;
- Then, each agent of the ith subsystem computes $u^i(t)$ and sends the information $(x^i(t), u^i(t))$ to the $(i+1)$th follower.

Moreover, because the whole sequential procedure needs to be completed within the given sampling interval Δt, in the sequel, such a time interval will be divided into equally spaced time slots $\Delta T_s := \frac{\Delta t}{l}$, each one allocated to the single agent.

Then, a computable distributed MPC scheme, hereafter denoted as *DMPC-LF*, consists of the following algorithm:

Distributed model predictive control leader–follower algorithm (DMPC-LF) – agent *i*th

---

INITIALISATION:

1: **compute** $\Xi^i_0 \subset \mathbb{R}^{n_i}$, $i = 1, \ldots, l$, ▷ the robust invariant ellipsoids
2: **compute** K^i, $i = 1, \ldots, l$, ▷ the corresponding stabilising state-feedback gainscomplying with the constraints (2) and (3);
3: **generate** the sequences $\{\mathcal{T}^i_j\}^{N_i}_{j=1}$, $i = 1, \ldots, l$, via recursions (23)–(26) and the corresponding projected regions $\{\mathcal{E}^{x^i}_j\}^{N_i}_{j=1}$, $i = 1, \ldots, l$, such that

$$x(0) \in \bigcup_j \left\{ \mathcal{E}^{x^1}_j \times \cdots \times \mathcal{E}^{x^l}_j \right\} \tag{42}$$

4: **compute** $\bigcup_s \mathrm{Proj}_{x^i}\{\mathcal{R}^i_s\}$, $i = 1, \ldots, l$, and the corresponding inner polyhedral approximations $\mathcal{P}^i$, $i = 1, \ldots, l$;
5: **determine** the partitions $\mathcal{S}^i_k$, $k = 1, \ldots, N^i_p$, $i = 1, \ldots, l$, and the associated controllers (28);
6: **store** $\{\mathcal{T}^i_j\}^{N_i}_{j=1}$, $\{\mathcal{E}^{x^i}_j\}^{N_i}_{j=1}$, $\{\mathcal{S}^i_k\}^{N^i_p}_{k=1}$, $i = 1, \ldots, l$, and the control laws $u(x^i) = F^i_k x^i + g^i_k$, $k = 1, \ldots, N^i_p$, $i = 1, \ldots, l$;
7: $\Delta T_s \leftarrow \frac{\Delta t}{l}$ and $t_{curr} \leftarrow 0$ ▷ the slot time interval and the time counter, respectively;

Remark 4.1: Note that the main computational complexity source concerns with the computation of the family of one-step-ahead controllable sets by means of recursions (23)–(26). However, it is important to remark that such

TCON_A_1282178 TFJATS_StdSerif-USA4.cls January 23, 2017 20:49 Trim Info: 215mm × 280mm

On-line phase (Leader):

1: **activate** t_{curr} and **solve** (37)-(38) by considering $\mathcal{E}_{N_1}^{x^1}$
2: **if** $u^{1^*}(t) \neq \emptyset$ **then**
3: **find**

$$j(t) = \min\{j : [x^{1^T}(t)\, u^{(1^*)^T}(t)\, x^{1^T}(t)\, u^{(1^*)^T}(t)]^T \in \mathcal{T}_j^1\}$$

4: **if** $j(t) = 1$ **then**
5: $u^{1^*}(t) \leftarrow K^1 x^1(t)$
6: **else**
7: **call** the **QBOS** procedure with the input: $(\Delta T_s - \epsilon,\ t_{curr},\ 1, N_i,\ j(t))$
8: **end if**
9: **else**
10: **find**

$$\bar{k} : x^1(t) \in \mathcal{S}_{\bar{k}}^1$$

11: $u^{1^*}(t) \leftarrow F_{\bar{k}}^1 x^1(t) + g_{\bar{k}}^1$
12: **end if**
13: **apply** $u^{1^*}(t)$
14: **transmit** $x^1(t)$ and $u^{1^*}(t)$ to the agent 2
15: **reset** t_{curr}
16: $t \leftarrow t + 1$; **goto** 1;

On-line phase (Follower):

1: **activate** t_{curr} and **receive**$(x^{i-1}(t), u^{i-1}(t))$ from the $(i-1)$-th predecessor
2: **solve** (33)-(34) by considering $\mathcal{E}_{N_i}^{x^i}$
3: **if** $u^{i^*}(t) \neq \emptyset$ **then**
4: **find**

$$j(t) = \min\{j : [x^{i^T}(t)\, u^{(i^*)^T}(t)\, x^{(i-1)^T}(t)\, u^{((i-1)^*)^T}(t)]^T \in \mathcal{T}_j^1\}$$

5: **if** $j(t) = 1$ **then**
6: $u^{i^*}(t) \leftarrow K^i x^i(t)$
7: **else**
8: **call** the **QBOS** procedure with the input: $(\Delta T_s - \epsilon,\ t_{curr},\ i, N_i,\ j(t))$
9: **end if**
10: **else**
11: **find**

$$\bar{k} : x^i(t) \in \mathcal{S}_{\bar{k}}^i$$

12: $u^{i^*}(t) \leftarrow F_{\bar{k}}^i x^i(t) + g_{\bar{k}}^i$
13: **end if**
14: **apply** $u^{i^*}(t)$
15: **transmit** $x^i(t)$ and $u^{i^*}(t)$ to the $(i+1)$-th follower
16: **reset** t_{curr}
17: $t \leftarrow t + 1$; **goto** 1;

a computation is offline performed and efficient ellipsoidal algorithms are available (see e.g. Kurzhanski & Valyi, 1997).

Then, the Steps 2–8 for the structure of the follower online phase (respectively, Steps 3–9 for the leader agent) comes out, thanking to one-step controllable region features. In fact, by construction, if $x_{\text{aug}}^i(t) \in \mathcal{T}_j^i$, then an admissible input always exists and, as a consequence, the optimisation (33)–(34) has a non-empty solution $u^{i^*}(t)$. The opposite case $(u^{i^*}(t) \equiv \emptyset)$ implies that $x^i(t)$ must belong to the polyhedral partition $\mathcal{S}_k^i \subset \mathcal{P}^i$, $k = 1, \ldots, N_p^i$. Moreover, the *QBOS* procedure is called with $\Delta T_s \leftarrow \Delta T_s - \epsilon$, where the quantity ϵ denotes the estimated CPU time required to apply $u^{(\cdot)^*}(t)$ and to transmit the local info $x^{(\cdot)}(t)$ and $u^{(\cdot)^*}(t)$.

Finally, it is important to underline that for each ith agent the solution of just one convex QP optimisation is required to compute an admissible, although non-optimal, command input within the time slot ΔT_s, i.e. (30)–(31) or (37)–(38). Therefore, until the time slot ΔT_s is not expired, such a solution can be in principle improved by invoking the *QBOS* procedure.

The next proposition shows that the *DMPC-LF* algorithm enjoys the feasibility retention and closed-loop stability.

Proposition 4.3: *Let the sequences of sets $\mathcal{E}_j^{x^i}$ be non-empty such that*

$$x(0) \in \bigcup_j \left\{ \mathcal{E}_j^{x^1} \times \cdots \times \mathcal{E}_j^{x^l} \right\} \tag{43}$$

Then, the DMPC-LF algorithm always satisfies the constraints and ensures asymptotic stability.

Proof: The proof directly follows by exploiting all the arguments of Section 3.2. Let us assume that at the time instant t, the optimisation problems in Steps 1 (leader agent) and 2 (follower agent) are, respectively, feasible. By construction, there exists an input vector u satisfying (7)–(8) such that the set-membership requirements in (38) and (31) hold true. In order to ensure the feasibility retention, it is enough to guarantee that at the next time instant $t + 1$, the existence of an admissible control move $u^i(t + 1), i = 1, \ldots, l$.

Note that when $x_{\text{aug}}^i(t+1) \in \mathcal{T}_{N_i}^i$, the existence of an admissible input trivially comes out. On the other hand, when the worst-case $x_{\text{aug}}^i(t+1) \notin \mathcal{T}_{N_i}^i$ occurs (see Figure 1 for the sake of clarity), the piecewise affine partitions by construction guarantee that the state trajectory can be driven in a finite number of steps into $\mathcal{T}_{N_i}^i$. Then, since each agent one-step state evolution is forced to belong to the successor projected region, i.e.

$$x^i(t+1) \in \mathcal{E}_{j-1}^{x^i}, \quad i = 1, \ldots, l,$$

this implies that there exists a finite future time instant $t + k$ such that the augmented state is driven to the terminal set, $x^i_{\text{aug}}(t+k) \in \mathcal{T}^i_0$, $i = 1, \dots, l$. As a consequence,

$$[x^{1^T}(t)\, x^{2^T}(t) \dots x^{l^T}(t)]^T \to \mathcal{E}_0^{x^1} \times \mathcal{E}_0^{x^2} \times \cdots \times \mathcal{E}_0^{x^l},$$
$$\text{as } t \to \infty.$$

■

Corollary 4.1: *Let the one-step controllable sequences $\{\mathcal{T}^i_j\}_{j=1}^{N_i}$, $i = 1, \dots, l$, be computed by recursions (23)–(26) satisfying at each jth level set of the requirement (14). Then, $\bigcup_s \mathcal{R}^i_s \equiv \emptyset, i=1,\dots,l,$ and the feasibility and asymptotic stability properties of the DMPC-LF algorithm are preserved.*

Proof: Note that in this case, Steps 2–12 of the *online phase* (Leader) and Steps 3–13 of the *online phase* (Follower), respectively, reduce as follows:

Online phase (Leader)

1: **find**

$$j(t) = \min\{j : [x^{1^T}(t)\, u^{(1^*)^T}(t)\, x^{1^T}(t)\, u^{(1^*)^T}(t)]^T \in \mathcal{T}^1_j\}$$

2: **if** $j(t) = 1$ **then**
3: $\quad u^{1^*}(t) = K^1 x^1(t)$
4: **else**
5: $\quad$ solve (40)–(41) by considering $\mathcal{E}^{x^1}_{j(t)-1}$
6: **end if**

Online phase (Followers)

1: **find**

$$j(t) = \min\{j : [x^{i^T}(t)\, u^{(i^*)^T}(t)\, x^{(i-1)^T}(t)\, u^{((i-1))^T}(t)]^T \in \mathcal{T}^{x^i}_j\}$$

2: **if** $j(t) = 1$ **then**
3: $\quad u^{i^*}(t) = K^i x^i(t)$
4: **else**
5: $\quad$ solve (33)–(34) by considering $\mathcal{E}^{x^i}_{j(t)-1}$
6: **end if**

Let us consider a single ith agent. At the current time instant t, let $x^i_{\text{aug}}(t) \in \mathcal{T}^i_j$ be the admissible augmented state, then one has that by construction at $t+1$ each component of $x^i_{\text{aug}}(t)$ is driven inside the corresponding projected region $\mathcal{E}^{x^i}_{j-1}$. Even if the augmented one-step state evolution $x^i_{\text{aug}}(t+1)$ remains confined in $\mathcal{T}^i_j$, in virtue of the nesting property (26), there will exist a finite sequence of command inputs such a that at a certain future time instant $t+k$ one has that $x^i_{\text{aug}}(t+k) \in \mathcal{T}^i_j$. Therefore, by iteratively applying such arguments, the augmented state trajectory is asymptotically driven to the terminal set $\mathcal{T}^i_0$. ■

4.4 Extensions and future research

Following the ideas above developed, there are some extensions that can be made to the proposed set-up. First, the inclusion of time-delay occurrences and packet dropouts arising in the communication medium between each pair of LF agent and controller unit can be exploited by combining the proposed approach with the framework in Franzè, Tedesco, and Famularo (2015).

Another extension could be the generalisation of the proposed control framework to groups of agents with more complex dynamics where model uncertainties and nonlinearities have to be formally taken into account. Such an extension is not straightforward because it also concerns with an assessment on the overall control performance of the proposed scheme.

Finally, a very interesting future investigation will regard the analysis of leader–follower networks subject to switching topologies, i.e. the leader plays the role of an external input to the other followers to steer the whole group, and the members update their states based on the information available from their neighbours and the leader. Such problem can be addressed by invoking concepts coming from the switched control system theory because any switching occurrence could lead to the loss of controllability of the given multi-agent team. It is evident that this will improve the flexibility property of the scheme when used in real applications.

5. Illustrative example

This section presents results of numerical simulations demonstrating the application of the *DMPC-LF* method and investigating its performance, including comparisons to the centralised counterpart (Angeli et al., 2008) and to the explicit extension scheme here referred as *MPC* and *PWA-MPC*, respectively.

We shall consider a formation of three robots (AG_1, AG_2, AG_3) where the robot AG_1 moves independently from AG_2 and AG_3. For each robot, we will use the point mobile robot model discussed in Kuwata (2005), whose the state consists of position and velocity components $x = [p_x\ p_y\ v_x\ v_y]^T$ and motions are governed by the following discrete-time LTI model:

$$x(t+1) = \Phi x(t) + G u(t)$$

where $u \in \mathbb{R}^2$ is the acceleration vector (m/sec^2),

$$\Phi = \begin{bmatrix} I_2 & \Delta t\, I_2 \\ 0_2 & I_2 \end{bmatrix}, \quad G = \begin{bmatrix} \frac{(\Delta t)^2 I_2}{2} \\ \Delta t I_2 \end{bmatrix},$$

TCON_A_1282178 TFJATS_StdSerif-USA4.cls January 23, 2017 20:49 Trim Info: 215mm × 280mm

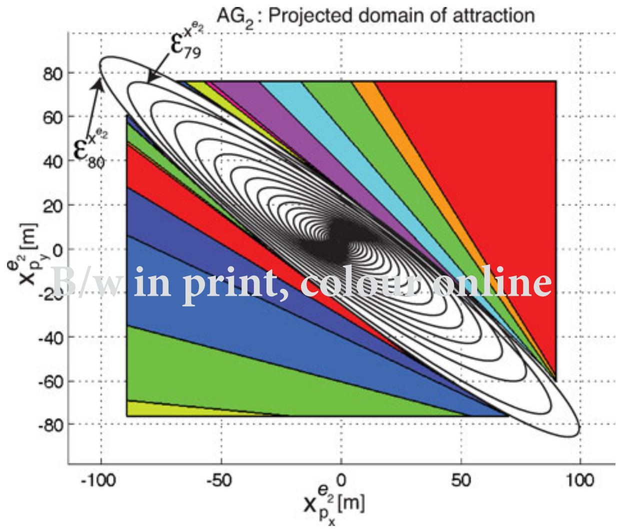

Figure 3. AG_2: Projected one-step controllable sets and piecewise partition.

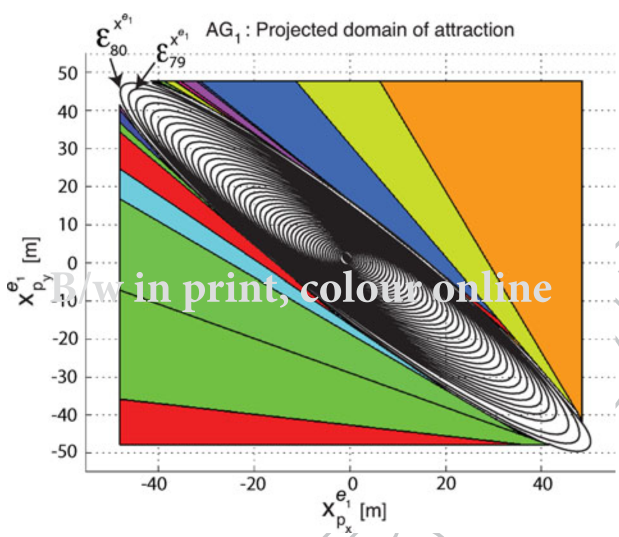

Figure 2. AG_1: Projected one-step controllable sets and piecewise partition.

with $\Delta t = 0.1$ sec, and subject to the saturation constraint

$$\|u(t)\|_2^2 \leq 1, \ \ \forall t \geq 0. \tag{44}$$

The aim of this simulation is to solve the *DC-LF-N* problem with $x^1(0) = [10, 30, 0, 0]^T$, $x^2(0) = [-65, 85, 0, 0]^T$, and $x^3(0) = [-110, 65, 0, 0]^T$, and where the goal is to reach the final position $x_f = [50, 5, 0, 0]^T$ under the satisfaction of the input constraints (44).

By defining $x^{e_1} := x^1$, $x^{e_2} := x^2 - x^1$, $x^{e_2} := x^3 - x^2$, the following error state-space description complying

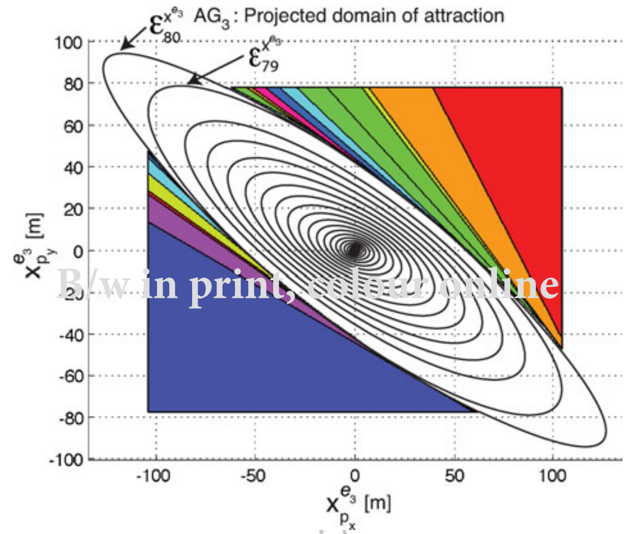

Figure 4. AG_3: Projected one-step controllable sets and piecewise partition.

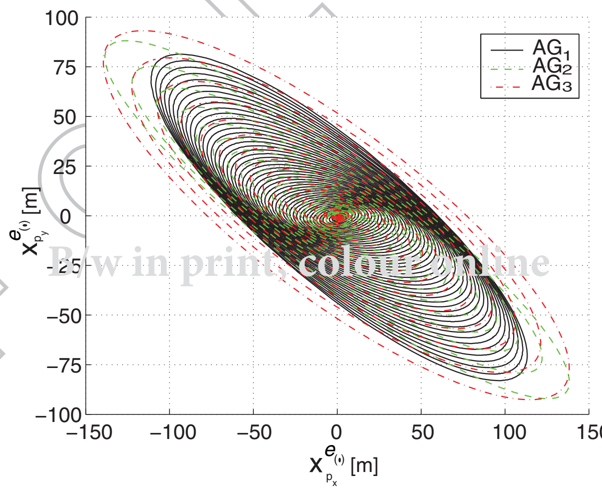

Figure 5. Projections of the one-step controllable sequence $\{\mathcal{E}_j^{MPC}\}_{j=0}^{80}$ for AG_1, AG_2 and AG_3.

with (5)–(6) is derived:

$$\begin{aligned} x^{e_1}(t+1) &= A^1 x^{e_1}(t) + B^1 u^1(t), \\ x^{e_2}(t+1) &= A^2 x^{e_2}(t) + B^2 u^2(t) + A^{2,1} x^{e_1}(t) + B^{2,1} u^1(t), \\ x^{e_3}(t+1) &= A^3 x^{e_3}(t) + B^3 u^3(t) + A^{3,2} x^{e_2}(t) + B^{3,2} u^2(t), \end{aligned} \tag{45}$$

where

$$A^1 = A^2 = A^3 := \Phi,\ A^{2,1} = A^{3,2} := \Phi - \Phi = 0_{n\times n},$$
$$B^1 = B^2 = B^3 = G,\ B^{2,1} = B^{3,2} = -G.$$

The above control problem can be recast as the regulation to the origin of (45) with $x^{e_1}(0) = [-40,\ 40,\ 0,\ 0]^T$, $x^{e_2}(0) = [-75,\ 40,\ 0,\ 0]^T$ and $x^{e_3}(0) = [-45,\ -20,\ 0,\ 0]^T$.

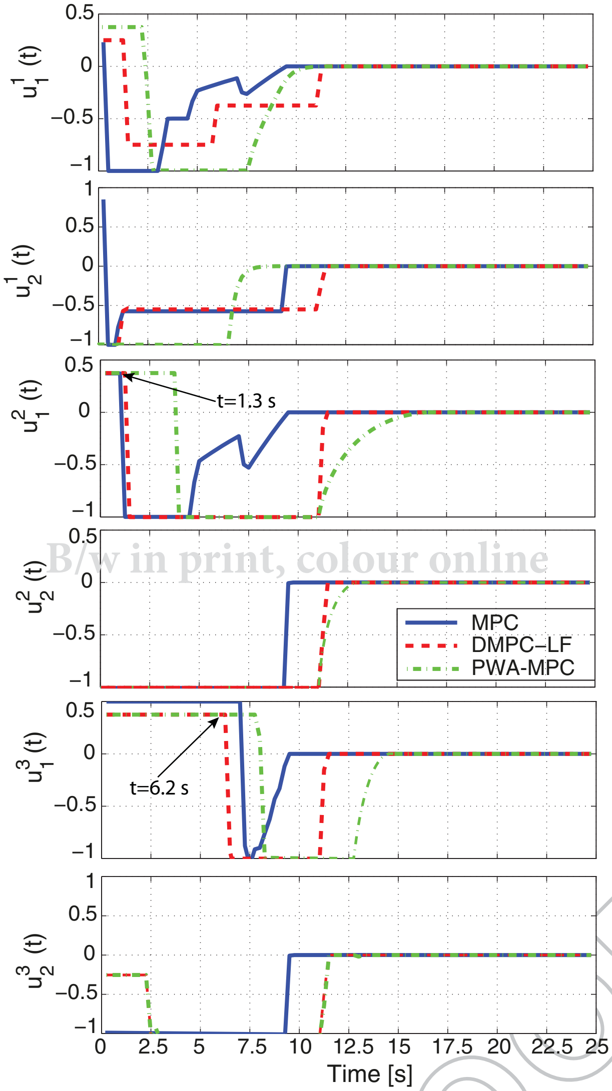

Figure 6. Applied command inputs.

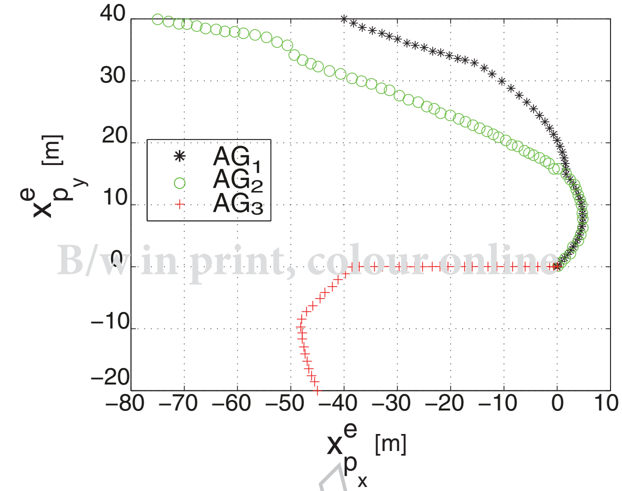

Figure 7. *DMPC-LF* algorithm: robot planar trajectories.

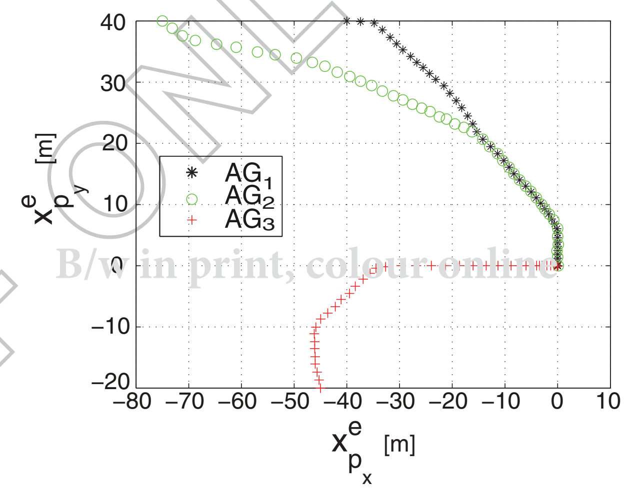

Figure 8. *MPC* algorithm: robot planar trajectories.

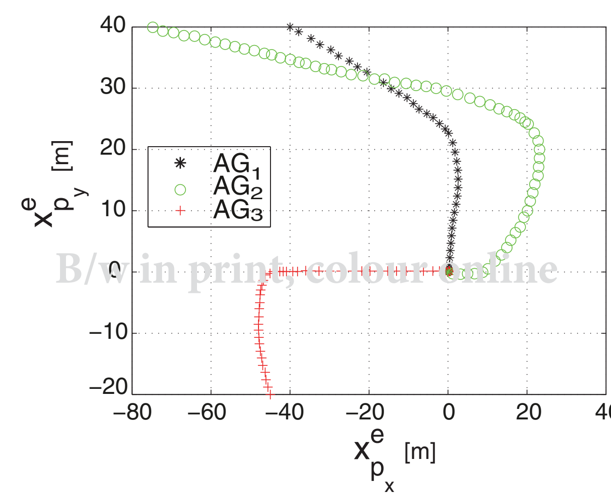

Figure 9. *PWA-MPC* algorithm: robot planar trajectories.

All simulations have been performed using Yalmip language (Löfberg, 2004), all under MATLAB© 8.2 environment, running in an Intel© Core i5-3330 machine with 3.3 GHz and 8 GB RAM.

First, as it results from Figure 6, the prescribed constraints are always fulfilled. Figures 2–4 depict the projection onto the planar components $(x^e_{p_x}, x^e_{p_y})$ of the domain of attractions pertaining to the three robots and achieved by using Theorem 3.1. In particular, for each subsystem, a family of 80 ellipsoids has been computed by using recursions (23)–(26), while the solutions of the *mp-QP* problems arising from (19)–(20) with $N = 1$, $Q^i = I_n$ and $R^i = I_m$, $i = 1, 2, 3$, have partitioned the polyhedral sets $\mathcal{P}^i$, $i = 1, 2, 3$, into $N^1_p = 189$, $N^2_p = 259$, and $N^3_p = 283$ polyhedral regions, respectively.

Moreover, as the centralised algorithm *MPC* is concerned, the same number of ellipsoids, here referred to as $\{T^{\mathrm{MPC}}_j\}_{j=0}^{80}$, has been computed on the whole system (45),

Table 1. Discrimination of computational burdens.

Online CPU time DMPC-LF	MPC	PWA-MPC
0.0028	0.008	0.00012

see Figure 5, where the one-step controllable sequences of the three agents are shown with respect to its own planar components. First, it is interesting to put in light that the domain of attraction of the leader subsystem (see Figure 2) is encapsulated within the domains of the two followers. The latter is complying with the underline methodological results because this represents a necessary condition to impose the feasibility retention as prescribed in Theorem 3.1. Then, notice that only the initial condition of the leader agent AG_1 belongs to its one-step controllable sequence, i.e. $x^{e_1}(0) \in \mathcal{E}_{76}^{x_1^{e}}$, while the two followers are outside and belong to $\mathcal{S}_{87}^2$ and $\mathcal{S}_{115}^3$, respectively. This is complying with the command input dynamical evolutions of Figure 6. In fact, by referring to the AG_2 agent, it is clear that the *DMPC-LF* and *PWA-MPC* algorithms perform exactly the same until $t = 1.4$ sec in virtue of the set-membership to the polyhedral partition $\{\mathcal{S}_k^2\}_{k=1}^{259}$. Then, at that time instant $x^{e_2}(1.4) \in \mathcal{E}_{71}^{x_2^{e}}$, and the *DMPC-LF* scheme prescribes the switching to the one-step controllable sequence $\{\mathcal{E}_j^{x_1^{e}}\}_{j=0}^{80}$ with an evident performance improvement as also testified by taking a look to the robots trajectories on the real planar environment, see Figures 7–9. The same reasoning holds true for the AG_3 agent with $x^{e_3}(6.2) \in \mathcal{E}_{53}^{x_3^{e}}$.

Finally, a further comparison amongst *DMPC-LF*, *MPC* and *PWA-MPC* under a different perspective has been carried out in terms of CPU time required during the online phase. Results have been reported in Table 1 where it can be noticed that the CPU time required by the *DMPC-LF* is significantly lower with respect to its centralised competitor. Moreover, in this respect, it is worth remarking that if the number of agents grows up the online CPU time required by the *MPC* would accordingly increase while it would remain almost constant and independent by the number of agents in the *DMPC-LF* case.

6. Conclusions

In this paper, we have presented a novel DMPC strategy for solving the formation problem of leader–follower networks. Key features of the proposed method, which combine explicit robust-model-based control techniques with set-theoretic ideas in a unique framework, can be summarised as low communication facilities, low computational demands, domain of attractions and control performance comparable with the centralised counterpart. Moreover, feasibility retention, viz. constraints fulfilment, and closed-loop asymptotic stability have been formally proved. Finally, the benefits of the proposed *DMPC-LF* algorithm and its advantages with respect to a centralised *MPC* and a decentralised strategy *PWA-MPC* are illustrated by means of a point mobile robots team.

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Walter Lucia http://orcid.org/0000-0003-3776-8331
Francesco Tedesco http://orcid.org/0000-0001-5876-3711

References

Angeli, D., Casavola, A., Franzè, G., & Mosca, E. (2008). An ellipsoidal off-line MPC scheme for uncertain polytopic discrete-time systems. *Automatica, 44*, 3113–3119.

Bemporad, A., Morari, M., Dua, V., & Pistikopoulos, E.N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica, 38*, 3–20.

Blanchini, F., & Miani, S. (2008). *Set-theoretic methods in control*. Boston, MA: Birkäuser.

Christofides, P.D., Scattolini, R., de la Peña, D.M., & Liue, J. (2013). Distributed model predictive control: A tutorial review and future research directions. *Computers and Chemical Engineering, 51*, 21–41.

~~Domahidi, A., Zgraggen, A.U., Zeilinger, M.N., Morari, M., & Jones, C.N. (2012). Efficient interior point methods for multistage problems arising in receding horizon control. *IEEE ACC.*~~

Dunbar, W.B. (2007). Distributed receding horizon control of dynamically coupled nonlinear systems. *IEEE Transactions on Automatic Control, 52*, 1249–1263.

Fax, J.A., & Murray, R.M. (2004). Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control, 49*, 1465–1476.

Franzè, G., & Tedesco, F. (2013). A distributed receding Horizon control scheme for leader–follower architectures: A set-theoretic approach. In *4th IFAC Workshop on Distributed Estimation and Control in Networked Systems*, Koblenz.

Franzè, G., Lucia, W., Tedesco, F., & Scordamaglia, V. (2014). A distributed obstacle avoidance MPC strategy for leader-follower formations. *IFAC Proceedings Volumes, 47*(3), 2570–2575.

Franzè, G., & Lucia, W. (2015). A receding Horizon control strategy for autonomous vehicles in dynamic environments. *IEEE Transactions on Control Systems Technology, 24*(2), 695–702.

Franzè, G., & Lucia, W. A set-theoretic control architecture for constrained switching systems. In *American Control Conference (ACC)* (pp. 685–690).

Franzè, G., Tedesco, F., & Famularo, D. (2015). Model predictive control for constrained networked systems subject to data losses. *Automatica, 54*, 272–278.

Galbusera, L., Ferrari-Trecate, G., & R.Scattolini., (2013). A hybrid model predictive control scheme for containment and distributed sensing in multi-agent systems. *System & Control Letter, 62*, 413–419.

Gao, Y., & Chong, K.T. (2012). The explicit constrained min-max model predictive control of a discrete-time linear system with uncertain disturbances. *IEEE Transactions on Automatic Control, 57*, 2373–2378.

~~Hong, Y., Gao, L., Cheng, D., & Hu, J. (2007). Lyapunov-based approach to multi-agent systems with switching jointly connected interconnection. *IEEE Transactions on Automatic Control, 52*, 943–948.~~

Jadbabaie, A., Lin, J., & Morse, A.S. (2003). Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control, 48*, 988–1001.

Ji, Z., Lin, H., & Yu, H. (2015). Protocols design and uncontrollable topologies construction for multi-agent networks. *IEEE Transactions on Automatic Control, 60*(3), 781–786.

Kothare, M.V., Balakrishnan, V., & Morari, M. (1996). Robust constrained model predictive control using linear matrix inequalities. *Automatica, 32*, 1361–1379.

Kurzhanski, A., & Valyi, I. (1997). *Ellipsoidal calculus for estimation and control*. Berlin: Birkhauser.

Kuwata, Y., Schouwenaars, T., Richards, A., & How, J. (2005). Robust constrained receding horizon control for trajectory planning. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*.

Lin, Z., Broucke, M., & Francis, B. (2004). Local control strategies for groups of mobile autonomous agents. *IEEE Transactions on Automatic Control, 49*, 622–629.

Liu, J., Chen, X., de la Peña, D.M., & Christofides, P.D. (2010). Sequential and iterative architectures for distributed model predictive control of nonlinear process systems. *AIChE Journal, 56*, 2137–2149.

Liu, B., Chu, T., Wang, L., & Xie, G. (2006). Controllability of a class of multi-agent systems with a leader. In *IEEE ACC* (pp. 2844–2849).

Löfberg, J. (2004). YALMIP : A toolbox for modeling and optimization in MATLAB. In *CACSD Conference*, Taipei. Retrieved from http://users.isy.liu.se/johanl/yalmip

Maestre, J.M., de la Peña, D.M., & Camacho, E.F. (2011). Distributed model predictive control based on a cooperative game. *Optimal Control Applications and Methods, 32*, 153–176.

Magni, L., & Scattolini, R. (2006).Stabilizing decentralized model predictive control of nonlinear systems. *Automatica, 42*, 1231–1236.

Mu, S., Chu, T., & Wang, L. (2005). Coordinated collective motion in a motile particle group with a leader. *Physica A, 351*, 211–226.

Necoara, I., Dumitrache, I., & Suykens, J.A.K. (2012). Smoothing techniques based distributed model predictive control algorithms for networks. In R. Sipahi et al. (Ed.), *Time delay systems: Methods, applications and new trends* (Vol. 423). LNCS, Springer.

Sakizlis, V., Kakalis, N.M.P., Dua, V., Perkins, J.D., & Pistikopoulos, E.N. (2004). Design of robust model-based controllers via parametric programming. *Automatica, 40*, 189–201.

Snzaier, M., & Damborg, M.J. (1987). Suboptimal control of linear systems with state and control inequality constraints. In *26th IEEE CDC* (Vol. 1, pp. 761–762).

Shi, H., Wang, L., & Chu, T. (2006). Virtual leader approach to coordinated control of multiple mobile agents with asymmetric interactions. *Physica D, 213*, 51–65.

Stewart, B.T., Venkat, A.N., Rawlings, J.B., Wright, S.J., & Pannocchia, G., "Cooperative distributed model predictive control. *System & Control Letter, 59*, 460–469.

Tanner, H.G., Pappas, G.J., & Kumar, V. (2004). Leader-to-formation stability. *IEEE Transactions on Automatic Robotics, 20*, 443–455.

Wang, W., & Slotine, J.E. (2006). A theoretical study of different leader roles in networks. *IEEE Transactions on Automatic Control, 51*, 1156–1161.

Wesselowski, K., & Fierro, R. (2003). A dual-mode model predictive controller for robot formations. In *42nd IEEE CDC* (pp. 3615–3620).