2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

On the Link Between Multi-Coloring Problems for Graphs and Distributed Supervision of Interconnected Systems

Alessandro Casavola, Emanuele Garone and Francesco Tedesco

***Abstract*— In this paper the link between Graph colorability theory and the design of a distributed Command Governor (CG) strategy is investigated. The first link between distributed supervision of systems with constraints and colorability properties was highlighted in a previous work where the idea of Turn-based CG was introduced for dynamically decoupled systems subject to coupling constraints. In this paper we will extend such a scheme to the case of dynamically coupled systems and we will show that the problem of determining the the minimal number of turns along with maximization of the agents appearance in turns is equivalent to a particular graph coloring problem. Such a problem is presented in a formal way and its complexity properties deeply discussed. A final example is presented to illustrate the effectiveness of the proposed strategy.**

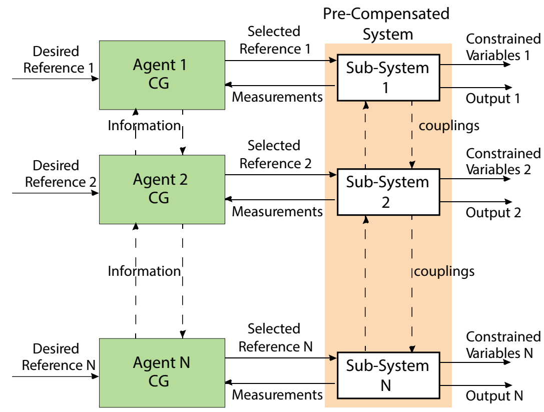

Fig. 1. Multi-agent architectures

I. Introduction

This paper focuses on the design of a distributed supervision strategy based on multi-agent Command Governor (CG) ideas for networked interconnected systems in situations where the use of a centralized coordination unit is impracticable because requiring unrealistic or unavailable communication and/or computation infrastructures.

The considered distributed context is depicted in Figure 1. There the supervisory task is allocated among many agents, which are assumed to be able to share information. Each agent is in charge of supervising and coordinating one specific subsystem by modifying the desired local reference whenever the joint application of all nominal references would produce constraint violations and hence loss of coordination.

Several works dealt with the above distributed CG problem. Among them it is worth mentioning [1], [2], [3]. Particular attention is devoted in this paper to the Turn-Based distributed CG approach presented in [5] that dramatically improves the performance and the scalability property of the Sequential distributed CG scheme introduced in [2]. The exploited idea is that agents not jointly involved in coupling constraints can simultaneously update their control actions without violating constraints. In that paper some hints were provided to show that the use of graph colorability theory is instrumental to systematically group agents into particular subsets (Turns) that allow this kind of parallelization.

Those preliminary results were presented for systems having no dynamic interactions. Here, the proposed approach has been generalized to the case where dynamic interactions are possible. Furthermore in this paper, we will discuss in depth the link between the grouping policy and the graph vertex coloring problem applied to the graph describing the couplings induced by the constraints among couples of agents. As well known, the minimal graph vertex coloring problem ([6]) aims at assigning a color to each vertex of a graph such that no two adjacent vertices may have the same color. The objective is to make this by using the fewest possible number of colors.

This paper treats systematically the connections with the graph colorability theory and formulates the optimal turn-definition problem as a new vertex multi-coloring problem [7]. Such a formulation is crucial to achieve more performing turn configurations of the network where agents may belong to multiple, not conflicting, turns. To the best of our knowledge, the specific problem introduced in this paper and its computational complexity properties have not been studied before. Moreover this paper proposes a heuristic to compute a sub-optimal multi-coloring round-robin policy.

In the final example the benefit of using a multi-coloring approach to group the agents will be shown in the case of coordination of a robots' formation.

II. Notations and Preliminaries

$\mathbf{R}$, $\mathbf{R}_+$ and $\mathbb{Z}_+$ denote the real, non-negative real and non-negative integer numbers respectively. The Euclidean norm of a vector $x \in \mathbf{R}^n$ is denoted by $\|x\| = \sqrt{x_1^2 + \ldots + x_n^2}$ whereas $\|x\|_\Psi^2, \Psi = \Psi^T > 0$, denotes the quadratic form $x^T \Psi x$. A generic ball in an Euclidean n-space $\mathbf{R}^n$ is defined as $\mathcal{B}_\delta := \{x \in \mathbf{R}^n : \|x\| \leq \delta\}$.

Definition 2.1: ***(Pontryagin Set Difference)*** - For given sets $\mathcal{A}, \mathcal{E} \subset \mathbf{R}^n$, the set determined as $\mathcal{A} \sim \mathcal{E} := \{a : a + e \in$

Francesco Tedesco and Alessandro Casavola are with DIMES, Università della Calabria, Via Pietro Bucci, Cubo 42-C, Rende (CS), 87036, ITALY, {ftedesco,a.casavola}@dimes.unical.it Emanuele Garone is with Université Libre de Bruxelles, 50 Av. F.D. Roosvelt, B-1050 Brussels, Belgium {egarone}@ulb.ac.be This research has been partially funded by the Belgian Fond Nationale de la Recherche Scientifique (FNRS) through the research project Optimization-free Control of Nonlinear Systems subject to Constraints, Ref. F452617F

$\mathcal{A}, \forall e \in \mathcal{E}\}$ is the *Pontryagin Set Difference* of $\mathcal{A}$ with respect to $\mathcal{E}$.

Definition 2.2: **Pareto Optimal solution (PO):** Consider the following multi-objective problem

$$\begin{array}{l} \min_g [f_1(g_1), f_2(g_2)..., f_N(g_N)] \\ \text{subject to } g = [g_1^T,...,g_i^T,...,g_N^T]^T \in \mathcal{S} \end{array} \quad (1)$$

The vector $g^{*p} \in \mathcal{S}$ is a PO solution if there exist no other $g \in \mathcal{S}$ such that: $f_i(g_i) \le f_i(g_i^{*p}), \quad i = 1,...,N$ and for which there exists a j such that $f_j(g_j) < f_j(g_j^{*p})$.

Definition 2.3: **Graph:** A graph is an ordered pair $\Gamma(\mathcal{A}, \mathcal{E})$, such that

- $\mathcal{A}$ is the set of nodes;
- $\mathcal{E}$ is a subset of pairs of Γ known as the set of edges connecting two nodes, i.e. $\mathcal{E} \subset \mathcal{A} \times \mathcal{A}$.

Definition 2.4: **Degree of a node** Given a graph $\Gamma(\mathcal{A}, \mathcal{E})$ $\Delta_i : \mathcal{A} \to \mathbb{Z}_+$ is the number of edges incident to a node.

Definition 2.5: (**Neighborhood of the *i*-th node**) The neighborhood $\mathcal{N}_i$ of the i-th node in $\Gamma(\mathcal{A}, \mathcal{E})$ consists of its adjacent nodes i.e.

$$\mathcal{N}_i = \{i\} \cup \{j \in \mathcal{A} : (i, j) \in \mathcal{E}\}. \quad (2)$$

A. Basic Centralized CG

Let us consider the following plant model:

$$\left\{ \begin{array}{rcl} x(t+1) & = & \Phi x(t) + Gg(t) \\ y(t) & = & H^y x(t) \\ c(t) & = & H^c x(t) + Lg(t) \end{array} \right. \quad (3)$$

where Φ, G, H^y, H^c, L have proper dimensions, $x(t) \in \mathbf{R}^n$ is the state including both plant and primal controller components, $g(t)$ the CG action, i.e. a suitably modified version of the reference signal $r(t) \in \mathbf{R}^m$, $y(t) \in \mathbf{R}^m$ the plant output which is required to track $r(t)$ and $c(t) \in \mathbf{R}^{n_c}$ the constrained output vector

$$c(t) \in \mathcal{C}, \quad \forall t \in \mathbb{Z}_+ \quad (4)$$

with $\mathcal{C}$ a specified convex and compact set.

Moreover it is assumed that: (**A1**) Φ is a Schur matrix.

The main idea of the classical centralized solution to the CG design problem (see [8]) is to choose at each time instant a set-point $g(t)$ approximating $r(t)$ such that:

1) for $g(t) \equiv g, \forall t$, the associated steady-state constrained output
$$c_g = H^c (I - \Phi)^{-1} Gg \quad (5)$$
satisfies the constraints with a margin $\delta > 0$, i.e. $c_g \in \mathcal{C} \sim \mathcal{B}_\delta$.
2) if the command $g(t) \equiv g$ is kept constant from t onward, constraints are never violated along the predictions of c, i.e. $c(x(t), k, g) \in \mathcal{C}, \forall k \ge 0$, where
$$c(k, x, g) := H^c \left(\Phi^k x + \sum_{\tau=0}^{k-1} \Phi^{k-\tau-1} Gg \right) + Lg \quad (6)$$

As proven in [10] and [8], the latter two conditions translates into confining all admissible g in a convex set defined as follows

$$\mathcal{V}(x) := \left\{ g \in \mathcal{W}_\delta : c(k, x, g) \in \mathcal{C}, \forall k \in \{0, 1, \dots, k_0\} \right\} \quad (7)$$

where k_0 is an integer that can be computed off-line on the basis of the system dynamics and

$$\mathcal{W}_\delta := \{g \in \mathbf{R}^m : \ c_g \in \mathcal{C} \sim \mathcal{B}_\delta\} \quad (8)$$

Then the Centralized CG problem can be summarized in finding the signal $g(t)$ by solving the following optimization problem

$$g(t) := \arg \min_{g \in \mathcal{V}(x(t))} \|g - r(t)\|_\Psi^2 \quad (9)$$

with matrix $\Psi = \Psi^T$ being a positive matrix. For details please refer to [8]. It has been proven that this scheme always ensures constraints satisfaction. Moreover, under constant references r, the applied command $g(t)$ converges in finite time to a constant reference $\hat{r}$, which is the best approximation of r compatible with the steady-state constraint (7).

III. System Description and Problem Formulation

Consider a set of N subsystems $\mathcal{A} = \{1, \dots, N\}$. Each subsystem is assumed to be a LTI closed-loop dynamical system regulated by a local controller that ensures asymptotic stability and good closed-loop properties when the constraints are not active (typically in small-signal regimes). Let the i-th closed-loop subsystem be described by the following discrete-time model

$$\left\{ \begin{array}{rcl} x_i(t+1) & = & \Phi_{ii} x_i(t) + G_i g_i(t) + \sum\limits_{j \in \mathcal{A} \setminus \{i\}} \Phi_{ij} x_j(t) \\ y_i(t) & = & H_i^y x_i(t) \\ c_i(t) & = & H_i^c x(t) + L_i g(t) \end{array} \right. \quad (10)$$

where: $t \in \mathbb{Z}_+$, $x_i \in \mathbf{R}^{n_i}$ is the state vector (which includes the controller states under dynamic regulation), $g_i \in \mathbf{R}^{m_i}$ the manipulable reference vector which, and $y_i \in \mathbf{R}^{m_i}$ is the output vector which is required to track a local desired reference r_i. $c_i \in \mathbf{R}^{n_i^c}$ represents the local constrained vector depending on the aggregate state vector $x = [x_1^T, \dots, x_N^T]^T \in \mathbf{R}^n$, with $n = \sum_{i=1}^N n_i$.

Let the aggregate constrained vector $c = [c_1^T, \dots, c_N^T]^T \in \mathbf{R}^{n^c}$, with $n^c = \sum_{i=1}^N n_i^c$. It is assumed that at each time instant $c(t)$ must be constrained as in (4) in the case where a $\mathcal{C} \subset \mathbf{R}^{n^c}$ is a convex and compact polytopic set. Note that $\mathcal{C}$ describes both local and coupling constraints.

The distributed CG design problem can be stated as follows

Problem *1: Given the systems* (10) *and constraints* (4)*, locally determine, at each time step t and for each agent $i \in \mathcal{A}$, a suitable reference signal $g_i(t)$ that is the best approximation of $r_i(t)$ such that its application do not produces constraints violation, i.e. $c(t) \in \mathcal{C}, \forall t \in \mathbb{Z}_+$.*

IV. Turn-Based Distributed CG

In this section the above stated problem will be solved by means of the distributed CG strategy presented in [5] where agents are grouped into turns on the basis of the constraints structure. In particular agents belonging to each turn are instructed to modify their reference according to a precise scheduling established offline. Such a strategy is here extended to the case of dynamically coupled systems.

A. Constraints associated to the i-th Agent

The first step towards the development of a distributed CG is to note that, under local dynamics (10), the aggregated system can be written as in (3) where $x(t) = [x_1^T(t), ..., x_N^T(t)]^T, g(t) = [g_1^T(t), ..., g_N^T(t)]^T$, $y(t) = [y_1^T(t), ..., y_N^T(t)]^T$, and $\Phi = [\Phi_{ij}]_{i,j=1,...,N}$, $G = diag(G_1, ..., G_N)$, $H^y = diag(H_1^y, ..., H_N^y)$, $H^c = [(H_1^c)^T, ..., (H_N^c)^T]^T$ and $L = [(L_1)^T, ..., (L_N)^T]^T$. Φ is assumed to satisfy property (**A1**). Such an assumption in this distributed context implies, according to Figure 1, that the local regulators of each subsystem (10) are capable to asymptotically stabilize the overall system (3).

In the case $\mathcal{C}$ is a polytope, $\mathcal{V}(x)$ is a polyhedron as well and can be characterized by a set of linear inequalities in $\mathbf{R}^m$ [11]

$$\mathcal{V}(x) := \{g \in \mathbf{R}^m : Ag + Qx - b \leq 0\} \tag{11}$$

A very important observation for the goals of this paper is that, once computed $\mathcal{V}(x)$ as in (11), the i-th agent may concur only to the violation of the constraints that directly depend on the command $g_i(t)$. Then each inequality in (11) involves just a limited number of agents. To put in evidence constraints involving i-th agent only, for each $i \in \mathcal{A}$, it is possible to define the block-matrices $A_i = [A_{i,1}|......|A_{i,N}]$, with $A_{i,j} \in \mathbf{R}^{z_i \times m_i}$, $Q_i = [Q_{i,1}|......|Q_{i,N}]$, with $Q_{i,j} \in \mathbf{R}^{z_i \times n_i}$ and vector $\tilde{b}_i \in \mathbf{R}^{z_i}$ collecting respectively all and only the, say z_i, rows a_j of A, rows q_j of Q and b_j of b in which the i-th agent is involved, i.e. such that the sub-vectors $a_j^i \neq 0_{m_i}$. As a consequence, the inequalities in $\mathcal{V}(x(t))$ associated to the i-th agent can be described as

$$A_i g(t) + Q_i x(t) \leq \tilde{b}_i \tag{12}$$

At this point, it is important to remark that, in some cases (e.g. for systems with no or weak dynamic interaction and coupling constraints), the matrix A can be quite sparse and only a subset of agents are involved in the constraints associated to the i-th agent.

It is worth emphasizing that the above constraints structure can be associated to an undirect graph $\Gamma(\mathcal{A}, \mathcal{E})$, whose set of nodes coincides with the set of agents $\mathcal{A}$ and where $\mathcal{E} \subset \mathcal{A} \times \mathcal{A}$ is the set of edges connecting agents (nodes) whose subsystem evolutions are jointly constrained with the i-th subsystem evolution as defined in (12), i.e. $j \in \mathcal{N}_i$ in $\Gamma(\mathcal{A}, \mathcal{E})$ if $A_{i,j} \neq 0$, where $A_{i,j} \in \mathbf{R}^{z_i \times m_i}$ is the j-th sub-matrix of A_i.

In this formulation, from the perspective of the i-th agent, constraints (4) become

$$\tilde{A}_i \tilde{g}_i(t) + \tilde{Q}_i \tilde{x}_i(t) \leq \tilde{b}_i \tag{13}$$

where $\tilde{g}_i(t) = S_{g,\mathcal{N}_i} g(t)$ and $\tilde{x}_i(t) = S_{g,\mathcal{N}_i} x(t)$ denote respectively the commands and the states associated to the neighbors of the i-th agent, with $\tilde{A}_i = A_i S_{g,\mathcal{N}_i}^T$ and $\tilde{Q}_i = Q_i S_{x,\mathcal{N}_i}^T$ the associated matrices. $S_{g,\mathcal{N}_i}$ and $S_{x,\mathcal{N}_i}$ are selection matrices that respectively extracts all rows of g and x related to the agents in $\mathcal{N}_i$.

Next, assume that at time t agent i receives from its neighbors the values of their states and of their previously applied commands, i.e. $\tilde{x}_i(t)$ and $\tilde{g}_i(t-1)$. If at time t all agents in $\mathcal{N}_i$ except the i-th were holding the commands applied at time $t-1$, then the i-th agent could select a local command g_i satisfying constraints (12) by fulfilling the following inequalities

$$\tilde{A}_i \tilde{g}_i \leq \tilde{b}_{i,\delta} - \tilde{Q}_i \tilde{x}_i(t) \tag{14}$$

where $\tilde{g}_i$ is set equal to $\tilde{g}_i(t-1)$ for all entries except g_i.

This idea can be extended to a larger number of agents using the notion of Turn:

Definition 4.1: (**Turn**) A turn $\mathcal{T} \subset \mathcal{A}$ is a *subset of non-neighboring nodes*, i.e. $\forall i, j \in \mathcal{T}$ such that $i \neq j$, $j \notin \mathcal{N}_i$ (none of them is a neighbor of the others).

The following proposition can be proved

Proposition 1: Let $\mathcal{T}_t \subset \mathcal{A}$ denote a turn selected at time t. Then, under the assumption that $Ag(0) + Qx(0) \leq b$, if at each time t all agents not in $\mathcal{T}_t$ keep applying their previously applied commands, i.e. $g_i(t) = g_i(t-1), \forall i \notin \mathcal{T}_t$ and the agents in $i \in \mathcal{T}_t$ update their commands g_i accordingly to (14), then the overall constraints (4) are never violated.

Proof: It directly follows from the structures of set $\mathcal{V}(x)$ and constraints (14). □

B. The Overall Algorithm

At this point, given a sequence of *turns* $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3 ...$, and assuming each agent knows to which turn it belongs to, the problem of locally determining at each time t the best $g_i(t)$ approximating $r_i(t)$ such that global constraints are satisfied can be solved by allowing only the agents in the current turn $\mathcal{T}_t$ to update their commands in accordance with constraints (14). This idea can be formalized as follows:

Algorithm 1: The Turn-Based CG (TB-CG)

(Agent i) repeat at each time t

if $i \in \mathcal{T}_t$ **then**
  receive $\tilde{x}_i(t), \tilde{g}_i(t-1)$ from neighbors;
  solve

$$g_i(t) = \arg\min_{g_i} ||g_i - r_i(t)||^2_{\Psi_i} \quad \text{subject to } (14) \tag{15}$$

else
  set $g_i(t) = g_i(t-1)$
  apply $g_i(t)$
  transmit $g_i(t)$ and $x_i(t)$ to the neighboring agents
end if

where $\Psi_i = \Psi_i^T > 0, \forall i \in \mathcal{A}$. Please note that the proposed algorithm satisfies the constraints using only local data. In fact, each agent in the turn $\mathcal{T}_t$ only needs to know the constraints in which it is involved and the commands, and the states of its neighbors.

C. Properties

The most relevant properties of the proposed Turn based-CG scheme are summarized in the following Theorem 1 ([5]).

Theorem 1: Consider systems (10) along with the distributed **TB-CG** (Algorithm 1) selection rule performed by agents in $\mathcal{A}$ distributed into turns $\mathcal{T}_i$ such that periodically, all agents of the network are selected, i.e. $\exists t' : \forall t > 0, \cup_{i=0}^{t'} \mathcal{T}_{t+i} = \mathcal{A}$. Assume that at time $t = 0$ an admissible solution $g(0)$ exists for problem (4). Then:

1) the overall system acted by the agents implementing the TB-CG policy never violates the constraints (4), $\forall t \in \mathbb{Z}_+$;
2) under the assumption that the set $\mathcal{W}_\delta$, defined in (8), is viable according to Definition given in [1], whenever $r(t) \equiv [\ r_1^T, \ldots, r_N^T\ ]^T, \forall t$, with r_i a constant set-point, the sequence of solutions $g(t) = [g_1^T(t), \ldots, g_N^T(t)]^T$ asymptotically converges to a Pareto-Optimal (PO) stationary (constant) solution of the following multi-objective optimization problem

$$\begin{array}{l} \min_g [\| g_1 - r_1 \|^2_{\Psi_1}, \ldots, \| g_i - r_i \|^2_{\Psi_2}, \ldots, \| g_N - r_N \|^2_{\Psi_N}] \\ \text{subject to } g = [g_1^T, \ldots, g_i^T, \ldots, g_N^T]^T \in \mathcal{W}_\delta \end{array} \tag{16}$$

The PO solution is given by r whenever $r \in \mathcal{W}_\delta$, or by any other solution $\hat{r} \in \mathcal{W}_\delta$ otherwise. □

V. Turns and Graph Multi-Colorability Theory

A crucial point of the proposed algorithm is the way the turn sets $\mathcal{T}_t$ are determined. As seen, the first general requirement to guarantee convergence is that, periodically, all agents of the network are selected, i.e. $\exists t' : \forall t > 0, \cup_{i=0}^{t'} \mathcal{T}_{t+i} = \mathcal{A}$.

In order to make the overall system behaving "fast" in response to fast changing reference signals, a further guideline is to choose a sequence of $\mathcal{T}_t$ that maximizes the frequency with which the agents are allowed to update their local commands.

From a practical implementation viewpoint the most reasonable choice is to resort to a periodic scheduling $\mathcal{T}_1, \ldots, \mathcal{T}_q$ of length q, with q as small as possible and such that $\cup_{i=1,..,q} \mathcal{T}_i = \mathcal{A}$. In this way, one can ensure that each agent can update its command with a frequency that is at least $1/q$. Furthermore, among the possible periodic definitions of turns that minimize q, it is convenient to chose the ones that maximize the sum of the frequencies with which each sensor is selected in the period. To formalize this idea, define the graph $\Gamma(\mathcal{A}, \mathcal{E})$, whose set of nodes coincides with $\mathcal{A}$ and where $\mathcal{E} \subset \mathcal{A} \times \mathcal{A}$ is the set of edges connecting neighbor agents, i.e. the edge (i, j) belongs to $\mathcal{E}$ if and only if $j \in \mathcal{N}_i$. The problem of determining a convenient set of turns can be defined as the following lexicographic problem

$$\begin{cases} q^* = \min\limits_{q \in \{1,\ldots,N\}, \mathcal{T}_1 \subseteq \mathcal{A}, \ldots, \mathcal{T}_q \subseteq \mathcal{A}} q \\ subject\ to \\ \bigcup_{t=1}^q \mathcal{T}_t = \mathcal{A} \\ (i, j) \notin \mathcal{E},\ \forall i, j \in \mathcal{T}_t, t = 1, \ldots, q. \end{cases} \tag{17}$$

$$\begin{cases} \max\limits_{\mathcal{T}_1 \subseteq \mathcal{A}, \ldots, \mathcal{T}_{q^*} \subseteq \mathcal{A}} \sum_{t=1}^{q^*} |\mathcal{T}_t| \\ subject\ to \\ \bigcup_{t=1}^{q^*} \mathcal{T}_t = \mathcal{A} \\ (i, j) \notin \mathcal{E},\ \forall i, j \in \mathcal{T}_t, t = 1, \ldots, q^*. \end{cases} \tag{18}$$

The first step of the lexicographic problem (17), it is equivalent to the well known minimal vertex coloring problem [6], i.e. to determine an assignment color to each graph vertex such that no two adjacent vertices have the same color and the number of colors is minimized. Accordingly, all the properties of the minimal vertex coloring apply.

The second step of the lexicographic problem (18) is used to exploit the fact that some agent can be selected more often than q, or in terms of graph colorability, that is possible to assign more than one color to some nodes.

A possible exact solution consists in first using existing state-of-the-art algorithms [6] to solve (17), and then solving (18) using backtracking. Clearly this approach is viable only at the design phase and for moderately small graphs. To deal with the other situations, where for instance online reconfiguration might be needed, a heuristic solution based on a greedy refining an (exact or approximated) minimal vertex coloring is proposed in [12]. In this paper we focus on some theoretical aspects not investigated in that work.

A. Properties

In this section some insights about computational complexity of Problem (18) are given. In this respect, it is convenient to rewrite problems (17) and (18) in their decision form as graph coloring problems:

Problem *2:* (**Graph q-coloring problem**) Let $\Gamma(\mathcal{A}, \mathcal{E})$ be a graph and $\mathcal{P}$ a set of q colors, assign a color of $\mathcal{P}$ to each vertex in $\mathcal{A}$ such that each pairs of neighbor nodes do not share the same color.

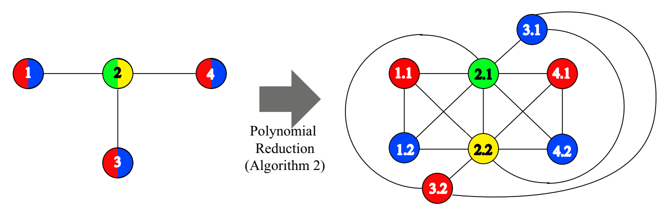

Fig. 2. Equivalence between Problem 4 with $p = 2$ and Problem 2 in the case where $|\mathcal{P}|$ is 4.

Problem *3:* (**Graph multi-coloring problem**) Let $\Gamma(\mathcal{A}, \mathcal{E})$ be a graph, $\mathcal{P}$ a set of colors and K a scalar greater than $|\mathcal{A}|$ respectively, assign one or more color of $\mathcal{P}$ to each vertex of Γ such that:

i) each pairs of neighbor nodes do not have common colors;
ii) the sum of nodes assigned to each color in $\mathcal{P}$ is greater than K

Clearly the Problem 3 with $K = |\mathcal{A}|$ reduces to Problem 2. Despite such relation, it is hard to establish if Problem 3 is NP-complete as Problem 2. In this respect we are able to prove such a result in the following specific case

Problem *4:* (**Graph constrained multi-coloring problem**) Let $\Gamma(\mathcal{A}, \mathcal{E})$ be a graph having n vertices, $\mathcal{P}$ a set of colors and a K a scalar greater than n respectively, assign p colors to each node of Γ such that each pairs of neighbors node do not have common colors.

The latter is a particular case of Problem 3 with $K = p|\mathcal{A}|$.

Proposition 2: Problem 4 is NP-complete.

Proof: As usual in computational complexity theory [13], to prove a Problem is NP-complete it is enough to prove that it belongs to the Nondeterministic Polynomial time complexity class NP and it can be reduced in polynomial time to a problem that is known to be NP-hard. In our case:

1) Problem 4 belongs to NP complexity class because it is possible to verify in polynomial time if a proposed map coloring is a solution. In fact it is enough to visit each node once and verify if it share some color with its neighbors.
2) Problem 4 can be reduced by means of a polynomial reduction [14] to the Problem 2 with the same set $\mathcal{P}$ that is known to be NP-hard in general. The reduction consists in generating a new graph $\Gamma'(\mathcal{A}',\mathcal{E}')$ by transforming each node in a p-clique. Please refer to Figure 2 for an intuitive idea.

Then it can be observed that if Problem 2 has a solution $\Gamma'(\mathcal{A}',\mathcal{E}')$, then Problem 4 has a solution $\Gamma(\mathcal{A},\mathcal{E})$ and *viceversa*. □

A second result we are able to prove about the computational complexity of Problem 3 is the following:

Proposition 3: If a q-coloring map of Γ complying with Problem 2 is known, then Problem 3 with $K = n+1$ can be solved in polynomial time.

Proof: In this case it is sufficient to find at least a node whom a second color can be assigned to. Then, in the worst case we have to visit each node of Γ. □

VI. Illustrative Examples

In order to show the effectiveness of the proposed method, a set of 21 decoupled particle masses representing vehicles has been considered, as depicted in Figure 3. The following equations describe the (i,j)-th mass dynamics

$$\begin{array}{rcl} m\ddot{x}_{i,j} &=& F^x_{i,j} \\ m\ddot{y}_{i,j} &=& F^y_{i,j} \end{array} \tag{19}$$

where $(x_{i,j},y_{i,j})$, $i \in \{1,2,..7\}$, $j \in \{1,2,3\}$ are the coordinates of the (i,j)-th mass position w.r.t a fixed Cartesian reference frame and $(F^x_{i,j},F^y_{i,j})$, $i \in \mathcal{A}$, the components along the same reference frame of the forces acting as inputs for the subsystems. The value $m = 1$ $[Kg]$ will be assumed in the simulations. For CG design purposes the models have been discretized with a sampling time of $T_c = 0.1$ $[s]$ and an optimal LQ state-feedback local controller is used as a precompensator for each mass.

Each mass is subject to the local constraints

$$\left|F^p_{i,j}(t)\right| \le 2\ [N] \quad p = x,y \tag{20}$$

representing input-saturation constraints on the forces $F^x_{i,j}$ and $F^y_{i,j}$, $i \in \{1,2,...,7\}$, $j \in \{1,2,3\}$, acting as inputs to the vehicles.

Moreover each couple of masses can be subject to proximity constraints involving either x position

$$0.125[m] \le |x_{i,j+1}(t) - x_{i,j}(t)| \le 0.375[m], \forall t \in \mathbb{Z}_+ \tag{21}$$

or y position

$$0.125[m] \le |y_{i+1,j}(t) - y_{i,j}(t)| \le 0.375[m], \forall t \in \mathbb{Z}_+ \tag{22}$$

or both x and y positions

$$\left\{ \begin{array}{l} 0.125[m] \le |x_{i+1,j+1}(t) - x_{i,j}(t)| \le 0.375[m] \\ 0.125[m] \le |y_{i+1,j+1}(t) - y_{i,j}(t)| \le 0.375[m], \\ 0.125[m] \le |x_{i-1,j+1}(t) - x_{i,j}(t)| \le 0.375[m] \\ 0.125[m] \le |y_{i-1,j+1}(t) - y_{i,j}(t)| \le 0.375[m], \\ \forall t \in \mathbb{Z}_+ \end{array} \right. \tag{23}$$

Along the presented simulations, each vehicle has been instructed to track a "circular" reference

$$\begin{array}{rcl} r^x_{i,j}(t) &=& \rho \cos\left(q_{i,j}\frac{2\pi}{75}t\right) + i\bar{r}, \quad t \in \mathbb{Z}_+ \\ r^y_{i,j}(t) &=& \rho \sin\left(q_{i,j}\frac{2\pi}{75}t\right) + j\bar{r}, \quad t \in \mathbb{Z}_+ \end{array} \tag{24}$$

with $i \in \{1,2,...,7\}$, $j \in \{1,2,3\}$, $\rho = 0.125[m]$ is the radius and $q_{i,j}$ denotes the spin direction evaluated in the following way

$$q_{i,j} = \left\{ \begin{array}{ll} 1, & \text{if} \quad i+j \quad \text{is an even number} \\ -1, & \text{if} \quad i+j \quad \text{is an odd number} \end{array} \right. \tag{25}$$

and scalar $\bar{r} = -0.0884[m]$.

We assume that each mass is subject to constraints of the type (21)-(23) according to the constraints coupling graph depicted in Fig. 3 where each vertex represents a mass, while the black dashed edges denote the existence of constraints between two masses. In this case the minimal vertex coloring problem (17) is solved by using only four colors (blu, red, green, yellow). As a consequence, the whole CG supervision action is spread among four groups of agents which adopt the Turn-Based policy introduced in Section 3. Since this constraint structure is quite sparser, it allows one to get (see Fig. 4) a multi-coloring map by performing reduction mentioned in the proof of Proposition 2.

Simulation results involve a comparison among three CG supervision strategies:

- the standard CG (Centralized CG) [8] where a unique CG device supervises the entire network by receiving/transmitting information to all subsystems;
- the Turn-Based distributed sequential CG presented in [5], hereafter referred to as Color-CG where the agents are grouped according to coloring map of Figure 3.
- the proposed Turn-Based distributed sequential CG with multi-coloring grouping policy summarized in Algorithm 2, hereafter referred to as m-Color-CG where the agents are grouped according to coloring map of Figure 4.

The numerical results appear in Fig. 5 and Table 1. In particular in Fig. 5 the cost $J(t) := \sum_{i=1}^{7}\sum_{j=1}^{3}\left(r_{i,j}(t) - g_{i,j}(t)\right)^2$ evaluated along the simulation is depicted. It is evident that the m-Color-CG gets better performance when compared to the Color-CG. Such an aspect is clearly related to the reduced average number of steps to the next decision reported in Table 1. In the same Table, by looking at the residual cost $\hat{J} := \frac{1}{T}\sum_{t=0}^{T} J(t)$, it is possible to note that the m-Color-CG features about the 20% of cost reduction, which is a significant improvement even if compared to the cost reduction of 25% obtained via the Centralized CG. This outcome is more relevant if we consider that is achieved by using the same amount of resources (see CPU Time and Average Info RX/TX in Table 1) used for the Color-CG scheme.

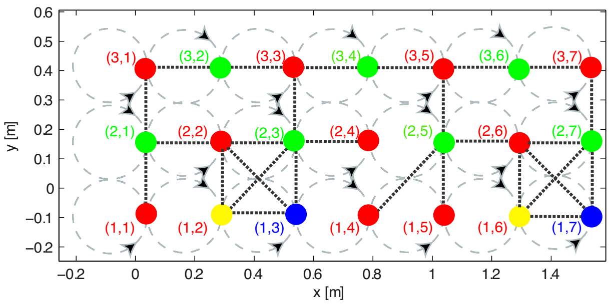

Fig. 3. Planar masses and coloring map

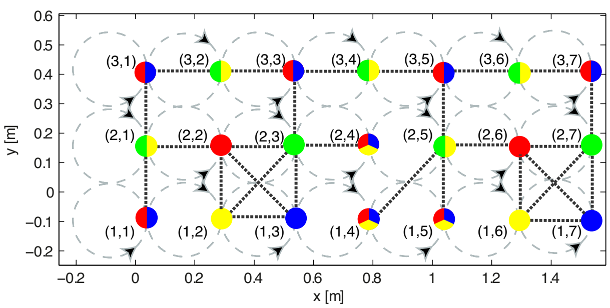

Fig. 4. Planar masses and multi-coloring map

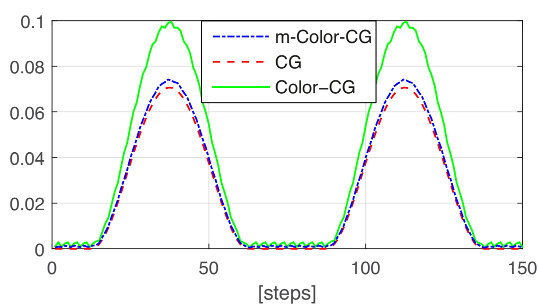

Fig. 5. Trend related to $J(t)$ during the simulation

TABLE I

SIMULATION RESULTS

	CG	Color-CG	m-Color-CG
AVERAGE STEPS TO NEXT DECISION	1	4	2.62
RESIDUAL COST $\hat{J}$	0.0502	0.067	0.0536
CPU TIME $[sec]$	0.0695	0.014	0.014
AVERAGE INFO RX/TX	4032	402	402

VII. CONCLUSIONS

In this work the novel Turn-Based sequential distributed CG scheme proposed in [5] has been extended to the case of dynamically coupled systems and its link with graph multi-coloring problems has been discussed in a formal way. Unlike the traditional sequential schemes of [2], where, according to a prescribed periodic order, only a single agent at a time was instructed to update its command while all others were committed to keep applying their current commands, here such a polling strategy is applied turn-wise, where a turn is a group of agents that can simultaneously update their commands without consequences on the fulfillment of the constraints.

Graph minimal vertex coloring problems have been shown to be instrumental for the determination of turns and for the implementation of the Turn-Based CG strategy, whose main properties concerning optimality, stability and feasibility have been discussed. Specifically a new link between distributed control and graph multi-coloring problems has been formally presented and used as a special tool to include agents into multiple not conflicting turns in order to increase the performance. Furthermore computational complexity of this novel class of problem has been investigated.

REFERENCES

[1] A. Casavola, E. Garone, F. Tedesco "A Distributed Multi-Agent Command Governor Strategy for the Coordination of Networked Interconnected Systems", *IEEE Transactions on Automatic Control*, 2014, 59 (8), pp. 2099 - 2112.

[2] F. Tedesco, A. Casavola, E. Garone, "Distributed Command Governor Strategies for Constrained Coordination of Multi-Agent Networked Systems", *Proc. of IEEE American Control Conference*, 2012, Montreal, Canada.

[3] U. Kalabić and I. Kolmanovsky. "Decentralized Constraint Enforcement Using Reference Governors". *Conference on Decision and Control*, Florence, Italy, pp. 6415-6421, 2013.

[4] A. Chinchuluun, P.M. Pardalos, A. Migdalas, L. Pitsoulis, *Pareto optimality, game theory and equilibria*. Springer, 2008

[5] A. Casavola, E. Garone, F. Tedesco"Scalability and performance improvement of distributed sequential command governor strategies via graph colorability theory", *Proceedings of 19th IFAC World Congress*, Cape Town, South Africa, August 24 - 29, 2014.

[6] T. Jensen, B. Toft, *Graph Coloring Problems*. Wiley-Interscience Ser. in Discrete Math. and Optim. John Wiley and Sons, Inc., N.Y. (1994).

[7] D. Marx, *Graph coloring with local and global constraints*, PhD thesis, Budapest University of Technology and Economics, 2004.

[8] A. Bemporad, A. Casavola and E. Mosca. "Nonlinear Control of Constrained Linear Systems via Predictive Reference Management", *IEEE Trans. Automat. Control*, 42, pp. 340-349, 1997.

[9] R. L. Brooks, "On colouring the nodes of a network", *Proc. Cambridge Philosophical Society, Math. Phys. Sci.*, 1941, 37, pp. 194197.

[10] E.G. Gilbert, I. Kolmanovsky and K. Tin Tan. "Discrete-time Reference Governors and the Nonlinear Control of Systems with State and Control Constraints". *Int. Journ. on Robust and Nonlinear Control*, 1995, 5, pp. 487-504,

[11] I. Kolmanovsky and E. G. Gilbert. "Maximal output admissible sets for discrete-time systems with disturbance inputs". *American Control Conference*, Seattle, pp. 1995-1999, 1995.

[12] A. Casavola, E. Garone, F. Tedesco, "A Distributed Command Governor based on Graph Colorability Theory", *International Journal of Robust and Nonlinear Control*, doi: 10.1002/rnc.4065

[13] S. Cook, "The Complexity of Theorem Proving Procedures", *Proceedings of the third annual ACM symposium on Theory of computing*, 1971, pp. 151158.

[14] R. M. Karp, "Reducibility Among Combinatorial Problems", *Complexity of Computer Computations*, Eds. R.E. Miller e J.W. Thatcher, New York, Plenum, 1972, pp. 85103.