2018 26th Mediterranean Conference on Control and Automation (MED)
Zadar, Croatia, June 19-22, 2018

Distributed kinematic control and trajectory scaling for multi-manipulator systems in presence of human operators

Martina Lippi, Alessandro Marino IEEE Senior Member

Abstract— **The objective of the paper is to devise a general framework for handling the human safety in a multi-robot work-cell controlled within a decentralized framework. The paper is motivated by the increasing demand coming from new production paradigms for strict cooperation between humans and robots and for flexibility and robustness provided by decentralized control frameworks. The cell foresees several robots with different assigned roles. In particular, it is supposed that there are *worker* agents, that are in charge of performing the cooperative manipulation task, and *watcher* robots, that are in charge of supervising the cell with particular attention to the human safety. The latter is guaranteed by properly modifying the workers' task trajectory according to a state transition strategy that tries to preserve the task path as much as possible. The overall solution is tested via simulations in order to show the effectiveness of results.**

Index Terms— **Distributed control; Human-robot collaboration.**

I. Introduction

The strict cooperation or the simple coexistence of humans and robots in the same area is a necessary feature of next generation production environments and of robotics in general.The reason behind is that this feature allows the creation of extremely flexible work-cells which can be easily reconfigured and that could benefit of the outperforming reasoning capabilities of humans. However, as outlined in [1], the human-robot cooperation requires the definition of suitable robot control strategies able to operate in such shared workspace with no human injury. In [2], two main approaches to guarantee the human safety are identified, namely, the power and force limiting and the speed and position monitoring. The former aims at limiting the human injury in the case of impact with a robot and is exploited, for example, in [3] to design a variable compliance at actuation level, and in [4], that introduces different collision detection and reaction techniques based on the reduction of contact forces. On the contrary, the latter aims at avoiding any possible collision by keeping a certain safety distance between robots and operators. To the purpose, it is necessary to develop real-time motion planning capabilities that enable the distance constraint to be satisfied in accordance with the dynamic behaviour of human operators. While a force limiting strategy is a key feature for physical human-robot interaction and collaborative tasks, an avoidance strategy may be more appropriate in the case of space sharing without explicit human-robot collaboration.

The most common avoidance approach is based on the theory of artificial potential field [5], according to which the robot motion is guided by attractive forces towards the desired configuration and repulsive ones that move it away from obstacles. The reactive collision avoidance strategy presented in [6] exploits this principle and reacts if virtual or physical contact occurs with the robot.

The work in [7] defines a quantity, named danger field, that quantifies how dangerous the robot state is to the objects or people in the environment. This information is then used in order to make the motion of the robot safer to the environment by modifying its path. Since in industrial scenarios the deviation from the nominal path may not be acceptable or not desirable, a possible solution can be to scale only the speed of the nominal trajectory. This approach is explored in [8] where human safety is assessed by a metrics based on robot braking time and is handled as a hard constraint in an optimization problem aimed at maximizing the scaling factor while meeting kinematic and safety constraints.

As highlighted by the aforementioned works, the research is extremely active regarding human avoidance by single robots, nevertheless the case of multi-robot systems has not been adequately tackled. In this scenario, the problem arising is how to combine the multi-robot cooperative task with the requirement of human safety, since the application of local avoidance strategies alone may not be suitable or sufficient to the purpose, e.g. when the robots are tightly connected as in the case of a transportation task. The problem is even more challenging in the case the multi-robot system is controlled within a decentralized architecture which endows the robotic cell with high flexibility and reconfigurability.

Motivated by these considerations, this study presents a general decentralized multi-robot architecture which allows to achieve collaborative tasks while respecting human safety constraints. The architecture foresees different kind of robots, namely the *workers* that are in charge of the manipulation tasks and the *watchers* that monitor the scene concerning the human presence. The information coming from watchers is adopted by workers to properly modify the task trajectory trying to preserve the task path as far as possible, in order to ensure the safety of the human. The architecture is completely decentralized in the sense that not central unit is adopted to implement the devised strategy.

The paper is organized as follows. Section II introduces the mathematical background and the problem setting. In Section III, by leveraging on results in [9] a distributed architecture for handling the human safety is introduced. This architecture is then exploited in Section IV to design a suitable human-robot collision avoidance strategy. Finally, numerical simulations and conclusions are presented in Sections V and VI, respectively.

II. Mathematical Background

By following the taxonomy developed in [10], the multi-arm work-cell configuration considered is composed by N *worker* robots which execute the main work the cell is aimed to, and one or more *watcher* robots whose aim is to supervise the cell to account for the safety of the cell itself and, in particular, of the human operators that might enter the cell, as it will be described in the following.

M. Lippi and A. Marino are with University of Salerno, Via Giovanni Paolo II, 132, 84084, Salerno (SA), Italy email: {mlippi,almarino}@unisa.it
*Authors are in alphabetical order.

The following assumption is made concerning robot dynamics.

Assumption 1. *Each robot is equipped with an inner motion control loop which guarantees tracking of a reference joint trajectory.*

This assumption is realistic for all commercial platforms and makes the devised approach suitable also for off-the shelf robotic platforms, for which the low level control layer is not generally made accessible.

Based on the above assumption, the second order kinematic relationship is assumed as robot model

$$\ddot{\boldsymbol{x}}_i = \boldsymbol{J}_i(\boldsymbol{q}_i)\ddot{\boldsymbol{q}}_i + \dot{\boldsymbol{J}}(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i = \boldsymbol{J}_i(\boldsymbol{q}_i)\boldsymbol{y}_i + \dot{\boldsymbol{J}}(\boldsymbol{q}_i)\dot{\boldsymbol{q}}_i \qquad (1)$$

where $\boldsymbol{q}_i \in \mathbb{R}^{n_i}$ ($\dot{\boldsymbol{q}}_i$, $\ddot{\boldsymbol{q}}_i$) is the joint position (velocity, acceleration) vector, $\boldsymbol{x}_i \in \mathbb{R}^p$ is the end-effector configuration of the ith manipulator with respect to the world frame, $\boldsymbol{J}_i(\boldsymbol{q}_i) \in \mathbb{R}^{p\times n_i}$ is the manipulator Jacobian matrix, and $\boldsymbol{y}_i = \ddot{\boldsymbol{q}}_i$ is the input of the assumed virtual model.

For the purpose of the overall description of the cell, let us introduce the collective vector $\boldsymbol{x} = \left[\boldsymbol{x}_1^{\mathrm{T}}, \quad \boldsymbol{x}_2^{\mathrm{T}}, \quad \ldots, \quad \boldsymbol{x}_N^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{Np}$.

In what follows, $\boldsymbol{O}_m$ and $\boldsymbol{I}_m$ denote the null and identity matrices in $\mathbb{R}^{m\times m}$, respectively, and $\boldsymbol{0}_m$ denote the column vector in $\mathbb{R}^m$ with all zero elements. The following assumption on the boundedness of the Jacobian $\boldsymbol{J}_i$ is made.

Assumption 2. *There exists a positive scalar constant α such that $\|\boldsymbol{J}_i(\boldsymbol{q}_i)\| \leq \alpha < \infty$, $\forall\, \boldsymbol{q}_i$ ($i = 1, 2, \ldots, N$).*

A. Information exchange

The information flow between robots is described by a communication graph $\mathcal{G}(\mathcal{E}, \mathcal{V})$ characterized by the set $\mathcal{V}$ of the indexes labeling the N vertices (nodes), the set of edges (arcs) $\mathcal{E} = \mathcal{V} \times \mathcal{V}$ connecting the nodes, and the $(N \times N)$ Adjacency matrix whose element a_{ij} is different from zero if node j can send information to node i. Therefore, $\mathcal{N}_i = \{j \in \mathcal{V} : (j, i) \in \mathcal{E}\}$ is the set of nodes that send information to ith node. We assume the graph is connected and *undirected* [11]. The Laplacian matrix associated to the graph $\mathcal{G}$ is defined as

$$\boldsymbol{L} = \{l_{ij}\} : \quad l_{ii} = \sum_{j=1}^{N} a_{ij}, \quad l_{ij} = -a_{ij},\ i \neq j$$

and zero is always a right eigenvalue with the $(N \times 1)$ vector of all ones, $\boldsymbol{1}_N \in \mathbb{R}^N$, as the corresponding right eigenvector, i.e., $\boldsymbol{L}\boldsymbol{1}_N = \boldsymbol{0}_N$. Hence, $\mathrm{rank}(\boldsymbol{L}) \leq N - 1$ where the equality holds when the graph is connected.

B. Problem setting

It is assumed that the task assigned to robots is described by a task function $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{x}) \in \mathbb{R}^m$:

$$\boldsymbol{\sigma} = \boldsymbol{J}_\sigma \boldsymbol{x}, \quad \dot{\boldsymbol{\sigma}} = \boldsymbol{J}_\sigma \dot{\boldsymbol{x}}, \quad \ddot{\boldsymbol{\sigma}} = \boldsymbol{J}_\sigma \ddot{\boldsymbol{x}} \qquad (2)$$

where $\boldsymbol{J}_\sigma \in \mathbb{R}^{m\times Np}$ is the task Jacobian matrix.

As possible and flexible choice, the task is here expressed by means of a proper set of absolute-relative variables [10]. In detail, the absolute variables define the position $\boldsymbol{p}$ and orientation $\boldsymbol{o}$ of the centroid of the end-effector configurations:

$$\boldsymbol{\sigma}_1 = \begin{bmatrix}\boldsymbol{p}\\ \boldsymbol{o}\end{bmatrix} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i = \boldsymbol{J}_{\sigma_1}\boldsymbol{x} \qquad (3)$$

with $\boldsymbol{J}_{\sigma_1} = \frac{1}{N}\boldsymbol{1}_N^{\mathrm{T}} \otimes \boldsymbol{I}_p \in \mathbb{R}^{p\times Np}$.

Regarding the relative motion, it can be described as below

$$\boldsymbol{\sigma}_2 = [(\boldsymbol{x}_N - \boldsymbol{x}_{N-1})^{\mathrm{T}} \ldots (\boldsymbol{x}_2 - \boldsymbol{x}_1)^{\mathrm{T}}]^{\mathrm{T}} = \boldsymbol{J}_{\sigma_2}\boldsymbol{x} \qquad (4)$$

with

$$\boldsymbol{J}_{\sigma_2} = \begin{bmatrix} -\boldsymbol{I}_p & \boldsymbol{I}_p & \boldsymbol{O}_p & \ldots & \boldsymbol{O}_p \\ \boldsymbol{O}_p & -\boldsymbol{I}_p & \boldsymbol{I}_p & \ldots & \boldsymbol{O}_p \\ \vdots & & \ddots & & \vdots \\ \boldsymbol{O}_p & \ldots & \boldsymbol{O}_p & -\boldsymbol{I}_p & \boldsymbol{I}_p \end{bmatrix} \in \mathbb{R}^{(N-1)p\times Np} \qquad (5)$$

Thus, a possible choice for the task function in (2) is

$$\boldsymbol{\sigma} = \begin{bmatrix}\boldsymbol{\sigma}_1\\ \boldsymbol{\sigma}_2\end{bmatrix} = \begin{bmatrix}\boldsymbol{J}_{\sigma_1}\\ \boldsymbol{J}_{\sigma_2}\end{bmatrix}\boldsymbol{x} = \boldsymbol{J}_\sigma \boldsymbol{x} \qquad (6)$$

with $\boldsymbol{J}_\sigma \in \mathbb{R}^{Np\times Np}$ and $m = Np$.

The main objective is to compute the input $\boldsymbol{y}_i$ in (1) in order to have $\boldsymbol{\sigma}$ tracking the reference $\boldsymbol{\sigma}_d(t)$ without a central control unit. In particular, the following assumption holds.

Assumption 3. *The reference value of the task function, $\boldsymbol{\sigma}_d(t)$, is known only by one of the worker robots (namely, the leader robot) while the other worker robots (namely, the follower robots) need to estimate this reference.*

Moreover, human operators can enter the cell during the robots' operation. In this situation, the safety of the humans is the highest priority task. To this aim, the watcher robots are in charge of identifying human operators, detecting potential risk situations, so that workers task can be possibly modified and made safe for humans. In this paper, we are interested neither in the planning of the watchers' motion nor in vision detection algorithms for human movement recognition. Instead, we aim at giving a flexible and decentralized architecture for multi-robot manipulation in presence of human operators and mainly focus on the workers' motion control strategy. Thus, the following assumption is made.

Assumption 4. *At any instant, information of operators possibly in the nearby of the work-cell is detected by the watchers and is made available to the leader robot (worker).*

Finally, it is worth mentioning that in the likely and general case of multiple watchers they can be arranged in a decentralized architecture such as to cooperatively and distributively monitor and track a given area [12].

Summarizing, the overall cell configuration and the information flow between the agents are depicted in Figure 1.

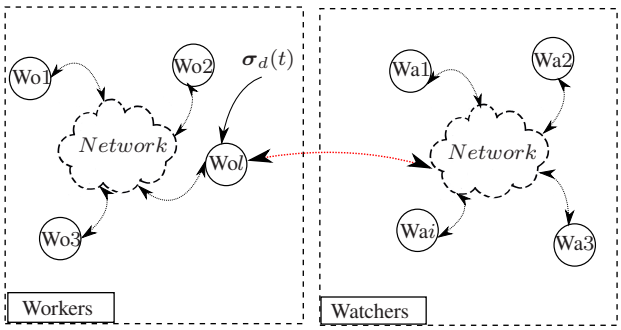


Fig. 1. The decentralized architecture underlying the work-cell. Watchers (Wa) are in charge of detecting and track humans and of communicating with the workers' (Wo) leader who also knows the task trajectory $\boldsymbol{\sigma}_d(t)$.

III. The distributed control strategy

In this section, the solution to the problem described in Section II-B is detailed. In particular, Section III-A briefly

introduce a strategy that allows follower robots to estimate leader reference, while Section III-B describes the decentralized workers' control scheme.

A. Distributed estimation of the task reference $\boldsymbol{\sigma}_d(t)$

Because of Assumption 3 and 4, the workers reference information is only known by the leader. Therefore let us consider the task reference $\boldsymbol{\sigma}_d(t)$ estimation.

By leveraging the approach in [13] and introducing the quantity $\boldsymbol{\zeta}_d = \dot{\boldsymbol{\sigma}}_d + k_\sigma \boldsymbol{\sigma}_d$ ($\dot{\boldsymbol{\zeta}}_d = \ddot{\boldsymbol{\sigma}}_d + k_\sigma \dot{\boldsymbol{\sigma}}_d$), being $k_\sigma > 0$ a positive scalar gain, it is possible for robot i to estimate $\boldsymbol{\zeta}_d$ and $\dot{\boldsymbol{\zeta}}_d$ with the following update rule

$$\begin{bmatrix} {}^i\dot{\hat{\boldsymbol{\zeta}}} \\ {}^i\ddot{\hat{\boldsymbol{\zeta}}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{O}_m & \boldsymbol{I}_m \\ \boldsymbol{O}_m & \boldsymbol{O}_m \end{bmatrix} \begin{bmatrix} {}^i\hat{\boldsymbol{\zeta}} \\ {}^i\dot{\hat{\boldsymbol{\zeta}}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{O}_m \\ \boldsymbol{I}_m \end{bmatrix} \boldsymbol{u}_{\zeta,i} \tag{7}$$

where the symbol ${}^i\hat{\boldsymbol{\zeta}}$ is adopted to denote the estimation of $\boldsymbol{\zeta}_d$ made by worker i and with $\boldsymbol{u}_{\zeta,i} \in \mathbb{R}^m$ chosen as

$$\boldsymbol{u}_{\zeta,i} = k_1 \text{sign}\left[\dot{\boldsymbol{\alpha}}_i + k_2 \text{sig}(\boldsymbol{\alpha}_i)^{0.5}\right] \tag{8}$$

$$\boldsymbol{\alpha}_i = \sum_{j \in \mathcal{N}_i} ({}^j\hat{\boldsymbol{\zeta}} - {}^i\hat{\boldsymbol{\zeta}}) + b_i(\boldsymbol{\zeta}_d - {}^i\hat{\boldsymbol{\zeta}}) \tag{9}$$

where k_1 and k_2 are positive scalar gains, $\text{sign}(\cdot)$ is the component-wise signum function, while $\text{sig}(\cdot)$ is such as

$$\text{sig}(\boldsymbol{x})^{0.5} = \left[\text{sign}(x_1)|x_1|^{0.5} \ \text{sign}(x_2)|x_2|^{0.5} \dots \ \text{sign}(x_n)|x_n|^{0.5}\right]^{\mathrm{T}}$$

for any vector $\boldsymbol{x} \in \mathbb{R}^n$. Concerning b_i in (8), it is equal to 1 only for the leader robot and is 0 otherwise.

By following the same reasoning as in [13], it is possible to prove that ${}^i\hat{\boldsymbol{\zeta}}$ (${}^i\dot{\hat{\boldsymbol{\zeta}}}$) converges to $\boldsymbol{\zeta}_d$ ($\dot{\boldsymbol{\zeta}}_d$) in finite time $T_s > 0$ provided that the communication graph as in Section II-A is connected and that gains k_1 and k_2 are properly chosen. The Reader might conveniently refer to [13] for the proof.

B. The worker control strategy

By following a similar approach as in [9], a layered architecture is adopted for the control of the worker model: the first layer aims at estimating the overall cell configuration $\boldsymbol{x}$, while the second layer is in charge of generating $\boldsymbol{y}_i$ in (1) according to this estimation and to the cell task.

1) The distributed observer: In order to compute the local control input $\boldsymbol{y}_i$, it is required to distributively estimate $\boldsymbol{x}$.

Let ${}^i\hat{\boldsymbol{x}} = \left[{}^i\hat{\boldsymbol{x}}_1^{\mathrm{T}}, {}^i\hat{\boldsymbol{x}}_2^{\mathrm{T}}, \ \dots, \ {}^i\hat{\boldsymbol{x}}_N^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{Np}$ be the estimate of $\boldsymbol{x}$ made by robot i. This estimate is obtained according to the following update law

$${}^i\dot{\hat{\boldsymbol{x}}} = k_o \left(\sum_{j \in \mathcal{N}_i} \left({}^j\hat{\boldsymbol{x}} - {}^i\hat{\boldsymbol{x}}\right) + \boldsymbol{\Pi}_i \left(\boldsymbol{x} - {}^i\hat{\boldsymbol{x}}\right) \right) + {}^i\hat{\boldsymbol{u}} \tag{10}$$

with

$$\boldsymbol{\Pi}_i = \text{diag}\{\boldsymbol{O}_p \ \cdots \ \underbrace{\boldsymbol{I}_p}_{i \text{ th node}} \ \cdots \ \boldsymbol{O}_p\} \in \mathbb{R}^{Np \times Np}$$

k_o a positive scalar gain and ${}^i\hat{\boldsymbol{u}}$ defined as

$${}^i\hat{\boldsymbol{u}}({}^i\hat{\boldsymbol{x}}, t) = \boldsymbol{J}_\sigma^\dagger \left({}^i\hat{\boldsymbol{\zeta}} - k_\sigma {}^i\hat{\boldsymbol{\sigma}}({}^i\hat{\boldsymbol{x}}) \right) \tag{11}$$

where ${}^i\hat{\boldsymbol{\sigma}}({}^i\hat{\boldsymbol{x}}) = \boldsymbol{J}_\sigma {}^i\hat{\boldsymbol{x}}$ (${}^i\dot{\hat{\boldsymbol{\sigma}}} = \boldsymbol{J}_\sigma {}^i\dot{\hat{\boldsymbol{x}}}$) is the estimate of the actual task function $\boldsymbol{\sigma}(\boldsymbol{x})$ as computed by robot i by resorting to the estimate ${}^i\hat{\boldsymbol{x}}$ from (10).

2) The local control input: Concerning the second layer, the virtual input $\boldsymbol{y}_i$ in (1) is selected as

$$\boldsymbol{y}_i = \ddot{\boldsymbol{q}}_{\sigma,i} + \lambda_\sigma \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} \tag{12}$$

where λ_σ is a positive gain to be selected and

$$\begin{cases} \dot{\boldsymbol{q}}_{\sigma,i} = \boldsymbol{J}_i^\dagger \boldsymbol{\Gamma}_i \boldsymbol{J}_\sigma^\dagger ({}^i\hat{\boldsymbol{\zeta}} - k_\sigma {}^i\hat{\boldsymbol{\sigma}}) \\ \ddot{\boldsymbol{q}}_{\sigma,i} = \boldsymbol{J}_i^\dagger \boldsymbol{\Gamma}_i \boldsymbol{J}_\sigma^\dagger ({}^i\dot{\hat{\boldsymbol{\zeta}}} - k_\sigma {}^i\dot{\hat{\boldsymbol{\sigma}}}) + \dot{\boldsymbol{J}}_i^\dagger \boldsymbol{J}_i \dot{\boldsymbol{q}}_{\sigma,i} \\ \dot{\tilde{\boldsymbol{q}}}_{\sigma,i} = \dot{\boldsymbol{q}}_{\sigma,i} - \dot{\boldsymbol{q}}_i, \end{cases} \tag{13}$$

and where

$$\boldsymbol{\Gamma}_i = \{\boldsymbol{O}_p \ \cdots \ \underbrace{\boldsymbol{I}_p}_{i \text{ th node}} \ \cdots \ \boldsymbol{O}_p\} \in \mathbb{R}^{p \times Np} \tag{14}$$

is a selection matrix.

Theorem 1. *Let us consider the model in* (1) *with the input in* (12)*, the observer* (10) *and the reference estimate* (7)*, then* ${}^i\tilde{\boldsymbol{x}} = \boldsymbol{x} - {}^i\hat{\boldsymbol{x}}$ *and* $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}_d - \boldsymbol{\sigma}$ *asymptotically converge to the origin provided that* k_o*,* k_σ *and* λ_σ *are chosen such as*

$$k_\sigma < \frac{k_o \lambda_L}{2N^2}, \ \lambda_\sigma > \frac{N\alpha^2}{4(k_o \lambda_L - 2N^2)} \tag{15}$$

with

$$\lambda_L = \lambda_{min}(\tilde{\boldsymbol{L}}^\star) \tag{16}$$

and $\tilde{\boldsymbol{L}}^\star = \boldsymbol{L} \otimes \boldsymbol{I}_{Np} + \boldsymbol{\Pi}^\star \in \mathbb{R}^{N^2p \times N^2p}$, $\boldsymbol{\Pi}^\star = \text{diag}\{\boldsymbol{\Pi}_1 \ \dots \ \boldsymbol{\Pi}_N\} \in \mathbb{R}^{N^2p \times N^2p}$.

Proof. The proof follows the same steps as in [9] with some minor differences related to the assumed virtual model $\boldsymbol{y}_i = \ddot{\boldsymbol{q}}_i$. Hence, for the sake of brevity, the proof is here omitted. □

IV. The human-robot avoidance strategy

In this section, we focus on the human-robot avoidance strategy whose aim is to guarantee the human safety while trying to preserve the robots main task as much as possible. In detail, the basic idea behind the safety strategy is to parameterize the nominal trajectory for $\boldsymbol{\sigma}$ in (6) as

$$\boldsymbol{\sigma}_d(t) = [\boldsymbol{\sigma}_{d,1}(t)^{\mathrm{T}}, \boldsymbol{\sigma}_{d,2}(t)^{\mathrm{T}}]^{\mathrm{T}} = \boldsymbol{\sigma}_d(s_n(t))$$

through a non-negative non-decreasing scalar function $s_n : t \in \mathbb{R} \to \mathbb{R}$ and to have the robots cooperative track

$$\boldsymbol{\sigma}_r(t) = [\boldsymbol{\sigma}_{r,1}(t)^{\mathrm{T}}, \boldsymbol{\sigma}_{r,2}(t)^{\mathrm{T}}]^{\mathrm{T}} = \boldsymbol{\sigma}_r(s_r(t))$$

where $s_r : t \in \mathbb{R} \to \mathbb{R}$ is a properly *scaled* version of $s_n(t)$ accounting for the human safety. This strategy allows the robots to preserve the task *path*; however, since it might result in a too much constraining strategy, the path deformation is allowed in the case no $s_r(t)$ able to guarantee the human safety is found. The strategy is detailed in the following.

In order to assess the level of safety in a human-robot cooperation scenario, we define a safety function f

$$f(d, \dot{d}, \ddot{d}) = \alpha_2(d, \dot{d})\ddot{d} + \alpha_1(d)\dot{d} + \alpha_0 d \tag{17}$$

where $\alpha_2 > 0$ and $\alpha_1, \alpha_0 \geq 0$ are scalar weights and $d = \|\boldsymbol{p}_r - \boldsymbol{p}_o\|$ is the distance between the reference position of manipulators centroid $\boldsymbol{p}_r$, defined in (3) and the operator's position $\boldsymbol{p}_o$ assumed to be available (see Assumption 4). The objective is to keep this safety value above a certain threshold f_{min} at all times, i.e., $0 \leq f_{min} \leq f < +\infty \, \forall t$. The ratio behind the expression of f in (17) is that it is a weighted sum of contributions linear in the distance and its derivatives and where weights can be, in general, configuration dependent. A possible choice for such coefficients is:

$$\begin{cases} \alpha_2(d, \dot{d}) = k_{\alpha,2} \left(1 + \exp(-\gamma_1 d - \gamma_2 \dot{d})\right) \\ \alpha_1(d) = k_{\alpha,1} \exp(-\gamma_3 d) \\ \alpha_0 = const \end{cases} \tag{18}$$

with γ_i $(i = 1, 2, 3)$ and $k_{\alpha,i}$ $(i = 1, 2)$ positive scalar constants. The ratio behind the choice of α_2, for example, is that the more the distance and its derivative are small the more important is the acceleration contribution.
Concerning the value of f_{min}, it represents the minimum allowed value of the safety measure and can be tuned via experimental trials and gathering the human feeling about the *experienced* level of safety. Another possibility is to dynamically estimate the human stress by measuring the cognitive workload from the analysis of the heart rate variability, defined as the variation over time of the interval between consecutive heart beats [14].

The strategy adopted to ensure the human safety is presented in Figure 2. Robots are asked to track the nominal trajectory as long as $f > f_{min}$; in the case the nominal trajectory would lead to $f \leq f_{min}$, then the nominal parameter $s_n(t)$ is scaled. Finally, if no scaling is found to meet the above constraint, the violation of the nominal path is allowed until the safety condition is recovered.

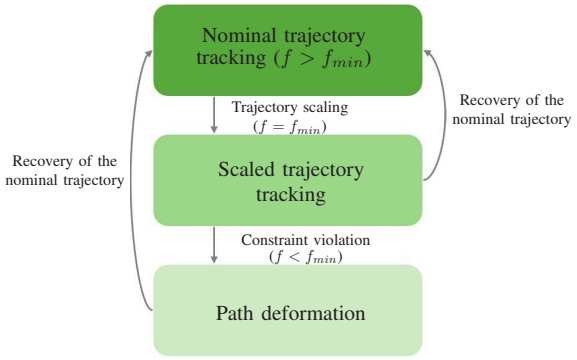


Fig. 2. High-level scheme describing possible states of human avoidance strategy; transition conditions are detailed in the following of Section IV.

A. Human-robot collision avoidance via trajectory scaling

As previously mentioned, a first strategy to ensure the human safety during the interaction is to scale the nominal trajectory parameter s_n. By leveraging a similar approach as in [15], designed for torque-limited path following of industrial robots, a scaled parameter $s_r(t)$ is introduced as:

$$\begin{cases} s_r(t) = s_n(t) + \Delta s(t) \\ \dot{s}_r(t) = \dot{s}_n(t) + \dot{\Delta} s(t) \\ \ddot{s}_r(t) = \ddot{s}_n(t) + \ddot{\Delta} s(t) \end{cases} \tag{19}$$

where $\Delta s(t)$ is adopted to properly scale the nominal path parameter and is such as $\Delta s(t) = \dot{\Delta} s(t) = \ddot{\Delta} s(t) = 0$ in nominal conditions. Moreover, it is required that $\forall t$

$$\dot{\Delta} s(t) \geq -\dot{s}_n(t) \tag{20}$$

$$s_n(t) + \Delta s(t) \leq s_n(t_f) \tag{21}$$

where t_f is the nominal final instant. The constraints (20) and (21) ensure that no reverse motion occurs along the path and that the end-point is not overcome, respectively.
The time derivative of the distance can be expressed as

$$\dot{d} = \frac{(\boldsymbol{p}_r - \boldsymbol{p}_o)^{\mathrm{T}}}{\|\boldsymbol{p}_r - \boldsymbol{p}_o\|}(\dot{\boldsymbol{p}}_r - \dot{\boldsymbol{p}}_o) = \gamma_1(s_r, \boldsymbol{p}_o)\dot{s}_r + \gamma_2(s_r, \boldsymbol{p}_o, \dot{\boldsymbol{p}}_o) \tag{22}$$

with

$$\begin{aligned} \gamma_1(s_r, \boldsymbol{p}_o) &= \frac{(\boldsymbol{p}_r - \boldsymbol{p}_o)^{\mathrm{T}}}{\|\boldsymbol{p}_r - \boldsymbol{p}_o\|}\frac{\partial \boldsymbol{p}_r(s_r)}{\partial s_r} \\ \gamma_2(s_r, \boldsymbol{p}_o, \dot{\boldsymbol{p}}_o) &= -\frac{(\boldsymbol{p}_r - \boldsymbol{p}_o)^{\mathrm{T}}}{\|\boldsymbol{p}_r - \boldsymbol{p}_o\|}\dot{\boldsymbol{p}}_o \end{aligned} \tag{23}$$

while the acceleration is

$$\ddot{d} = \gamma_1 \ddot{s}_r + \frac{\partial \gamma_1}{\partial s_r}\dot{s}_r^2 + \frac{\partial \gamma_1}{\partial \boldsymbol{p}_o}^{\mathrm{T}} \dot{\boldsymbol{p}}_o \dot{s}_r + \dot{\gamma}_2 \tag{24}$$

For the sake of notation compactness, the dependency of γ_1, γ_2, α_1 and α_2 on their parameters will be omitted.
By folding (24) in (17) and considering (19), one obtains

$$f = \bar{\alpha}_1(d, \dot{d})\ddot{\Delta} s + \bar{\alpha}_2(c, d, \dot{d}) \tag{25}$$

with

$$\bar{\alpha}_1 = \alpha_2 \gamma_1$$

$$\bar{\alpha}_2 = \alpha_2(\gamma_1 \ddot{s}_n + \frac{\partial \gamma_1}{\partial s_r}\dot{s}_r^2 + \frac{\partial \gamma_1}{\partial \boldsymbol{p}_o}^{\mathrm{T}} \dot{\boldsymbol{p}}_o \dot{s}_r + \dot{\gamma}_2) + \alpha_1(\gamma_1 \dot{s}_r + \gamma_2) + \alpha_0 d \tag{26}$$

which shows the dependency of f on the parameter $\ddot{\Delta} s$. In order to meet the constraint $f \geq f_{min}$, the following upper and lower bounds on $\ddot{\Delta} s$ can be found from (25):

$$\ddot{\Delta} s_{max} = \begin{cases} +\infty & \bar{\alpha}_1 \geq 0 \\ (f_{\min} - \bar{\alpha}_2)/\bar{\alpha}_1, & \bar{\alpha}_1 < 0 \end{cases} \tag{27}$$

and

$$\ddot{\Delta} s_{min} = \begin{cases} (f_{\min} - \bar{\alpha}_2)/\bar{\alpha}_1, & \bar{\alpha}_1 > 0 \\ -\infty & \bar{\alpha}_1 \leq 0 \end{cases} \tag{28}$$

These bounds are used within the following dynamic system

$$\begin{aligned} \ddot{\Delta} s &= -k_d \dot{\Delta} s - k_p \Delta s \\ \ddot{\Delta} s &= \mathrm{sat}(\ddot{\Delta} s, \ddot{\Delta} s_{min}, \ddot{\Delta} s_{max}) \end{aligned} \tag{29}$$

where $\Delta s(t_0) = \dot{\Delta} s(t_0) = 0$, being t_0 the initial time instant, k_d and k_p are positive constants and $\mathrm{sat}()$ is a saturation function that bounds $\ddot{\Delta} s$ in the range $[\ddot{\Delta} s_{min}, \ddot{\Delta} s_{max}]$. The ratio behind (29) is that in the first equation $\ddot{\Delta} s$ is computed in order to recover the nominal trajectory by dynamically bringing Δs to zero, while, in the second equation this value is saturated according to the computed bounds.

Remark 1. *The scaling procedure does not generally guarantee that the condition $f \geq f_{min}$ is met because of constraints in* (20) *and* (21)*, e.g. in the case the scaling leads to the condition $\dot{\Delta} s = -\dot{s}_n$ and $\ddot{\Delta} s_{max} < 0$, it is clear that further scaling would violate the constraint in* (20)*.*

Thus, a different avoidance strategy is proposed in the next section which allows $\boldsymbol{p}_r$ to violate the path constraint.

B. Impedance-based human-robot collision avoidance

In the case the dynamics in (29) leads to one of the constraints (20) and (21) being violated, the constraint on the nominal path is relaxed and the leader robot modifies $\boldsymbol{p}_r$ according to an impedance model.
Let be t_s the time instant in which the trajectory scaling leads to some constraint violation. From this moment on, the reference trajectory is modified as below

$$\begin{cases} \boldsymbol{p}_r(t) = \boldsymbol{p}_r(t_s^-) + \Delta \boldsymbol{p}_r(t) \\ \dot{\boldsymbol{p}}_r(t) = \Delta \dot{\boldsymbol{p}}_r(t) \\ \ddot{\boldsymbol{p}}_r(t) = \Delta \ddot{\boldsymbol{p}}_r(t) \end{cases} \tag{30}$$

where the displacement $\Delta \boldsymbol{p}_r(t)$ is computed according to the following dynamics

$$\boldsymbol{M}\ddot{\Delta \boldsymbol{p}}_r(t) + \boldsymbol{D}\dot{\Delta \boldsymbol{p}}_r(t) + \boldsymbol{K}\Delta \boldsymbol{p}_r(t) = \boldsymbol{f}_r(t) \tag{31}$$

with $\boldsymbol{M}$, $\boldsymbol{D}$, $\boldsymbol{K} \in \mathbb{R}^{3\times 3}$ positive definite matrices and $\boldsymbol{f}_r$ a virtual force to be properly designed. In order to guarantee the continuity of the reference trajectory $\boldsymbol{p}_r$ in the switching instant t_s, it is set $\Delta \boldsymbol{p}_r(t_s) = \boldsymbol{0}_3$ and $\dot{\Delta \boldsymbol{p}}_r(t_s) = \dot{\boldsymbol{p}}_r(t_s^-)$. The virtual force $\boldsymbol{f}_r$ is designed as

$$\boldsymbol{f}_r = \beta_1(d)\left(k_{r,1} + k_{r,2}\beta_2(\dot{d})\right)\nabla f \tag{32}$$

with $k_{r,i}\,(i = 1, 2)$ positive scalar gains, $\beta_i\,(i = 1, 2)$ activation functions depending on the distance and its derivative, respectively, while ∇f is the gradient of the safety function in (17) with respect to $\ddot{\boldsymbol{p}}_r$. The activation functions, shown in Figure 3, are such that $\beta_1(d)$ is unitary until $d < d(t_s) + \Delta d$, being $\Delta d \geq 0$, and then slowly decreases to 0, while $\beta_2(\dot{d})$ is unitary when $\dot{d} < 0$ and decreasing to the origin for $\dot{d} \geq 0$.

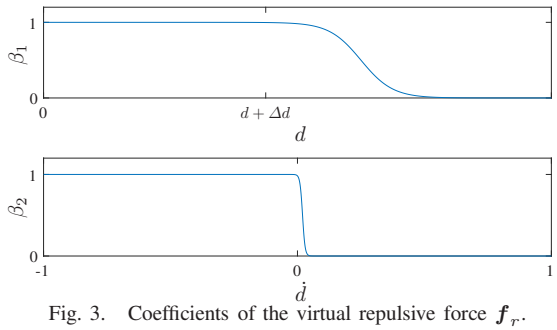


Fig. 3. Coefficients of the virtual repulsive force $\boldsymbol{f}_r$.

Concerning the return to the nominal trajectory tracking, the conditions to have this transition occur are

- $\boldsymbol{f}_r = \boldsymbol{0}_3$: repulsive forces are no longer required;
- $\Delta \boldsymbol{p}_r(t) = \dot{\Delta \boldsymbol{p}}_r(t) = \boldsymbol{0}_3$: the transient vanished.

V. Numerical Simulations

The effectiveness of the devised method has been validated with numerical simulations. The considered setup, shown in Figure 4, is composed by 3 workers ($N = 3$), each of which is a 8-DOFs ($n_i = 8$) Comau Smart Six (6-DOFs, $p = 6$) mounted on holonomic mobile bases (2-DOFs) able to move in the xy plane; 2 cameras representing the watcher agents are also provided in order to track possible operators in the cell. The objective of the multi-robot system is to perform a cooperative load transport task between 2 picking stations and a depositing one, ensuring the safety of an operator moving between two base stations inside the cell. A video of the simulation can be found at the following link[1].

By resorting to the task function in (6), such motions for cooperative transport can be easily described: $\boldsymbol{\sigma}_1$ is adopted for describing the position and orientation of the grasped object, while $\boldsymbol{\sigma}_2$ for describing the team formation. Moreover, robot 1 is considered the leader one therefore such desired trajectory (and its derivatives), as well as the human positions, are made available to it.

The communication graph is chosen as undirected and connected. Distributed control gains are chosen as $k_1 = 200$, $k_2 = 10$, $k_o = 20$, $k_\sigma = 10$, $\boldsymbol{\lambda}_\sigma = 10$ in (8), (10) and (12) while gains for avoidance strategy as $\gamma_1 = 0.1$, $\gamma_2 = 0.5$, $\gamma_3 = 0.5$, $k_{\alpha,1} = 2$, $k_{\alpha,2} = 0.5$, $\alpha_0 = 1$, $k_d = 4.5$, $k_p = 5$, $\boldsymbol{M} = \boldsymbol{I}_3$, $\boldsymbol{D} = 6.5\boldsymbol{I}_3$, $\boldsymbol{K} = 10\boldsymbol{I}_3$, $k_{r,1} = 6$, $k_{r,2} = 12$, $\Delta d = 1.5$ m in (18), (29) and (32); finally, the minimum safety value is set as $f_{min} = 3.5$.

[1] www.automatica.unisa.it/video/CoopHumanAvoidanceMED2018.mp4

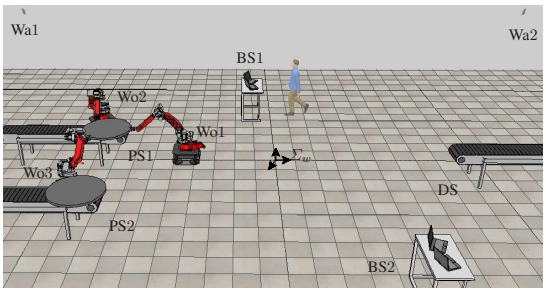
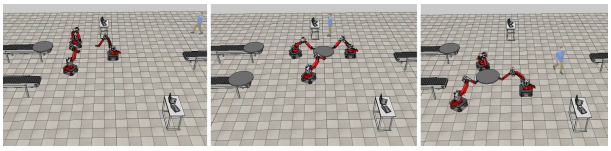


Fig. 4. Simulation setup composed by 3 workers (Woi), a human operator and 2 watchers (Wai); the picking and depositing stations for robots (PSi and DS, respectively) and the 2 base stations dedicated to the operator (BSi) are also shown, while Σ_w is the world reference frame.

Simulation snapshots are shown in Figure 5 while results are reported in Figures 6 - 10. In particular, human presence causes the alteration of nominal trajectory mainly in two phases: from $t = 8.5$ s up to $t = 15.5$ s, trajectory scaling occurs due to the passage of robots in the vicinity of the person (Figure 5.b) while, from $t = 36.4$ s up to $t = 53.4$ s, a deformation of the nominal path is made necessary since the operator crosses this nominal path (Figure 5.c); these phases will be referred as scaling (S) and impedance (I), respectively. The position of the operator has been assumed to coincide with that of the relative chest.



a) b) c)

Fig. 5. Simulation snapshots representing the initial system configuration (a), scaling phase (b) and impedance phase (c), respectively.

From Figure 6, it emerges that, during the scaling phase, the safety value is mainly saturated at the minimum value, while during the impedance phase a rapid increase is recorded due to the relaxation of path constraint. Figure 7 shows how scaling terms vary during the interaction, while Figure 8 reports the nominal and reference trajectories. In detail, in the scaling phase, a slowing down of the reference trajectory with respect to the nominal one can be observed; in the impedance phase, starting at time $t_s = 36.4s$, the reference trajectory is firstly modified from $\boldsymbol{p}_r(t_s)$ according to the impedance model in (31) and, in the end, it is returned to the same point $\boldsymbol{p}_r(t_s)$, in such a way to resume the nominal path tracking.

Finally, Figure 9 shows the evolution of the distance d and its derivatives during the interaction; while, for the sake of completeness, Figure 10 shows the asymptotic convergence to zero of the task tracking error $\|\tilde{\boldsymbol{\sigma}}\|$ and the observer error $\|\tilde{\boldsymbol{x}}^\star\|$, respectively, being $\tilde{\boldsymbol{x}}^\star = \begin{bmatrix} {}^1\tilde{\boldsymbol{x}}^{\mathrm{T}} & {}^2\tilde{\boldsymbol{x}}^{\mathrm{T}} & \dots & {}^N\tilde{\boldsymbol{x}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$. The initial error is due to the fact that ${}^i\tilde{\boldsymbol{x}}(t_0) \neq \boldsymbol{0}_{18}$, $\forall i$.

VI. Conclusions

This paper proposed a flexible architecture for distributed control of multi-robot systems accounting for human safety. In detail, such architecture consists of watcher robots, which

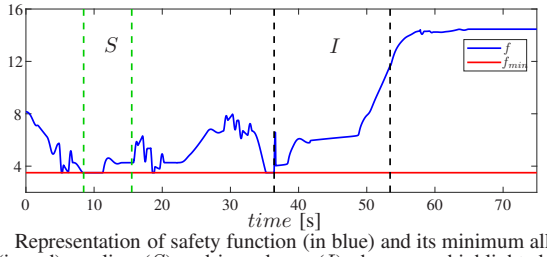


Fig. 6. Representation of safety function (in blue) and its minimum allowed value (in red); scaling (S) and impedance (I) phases are highlighted.

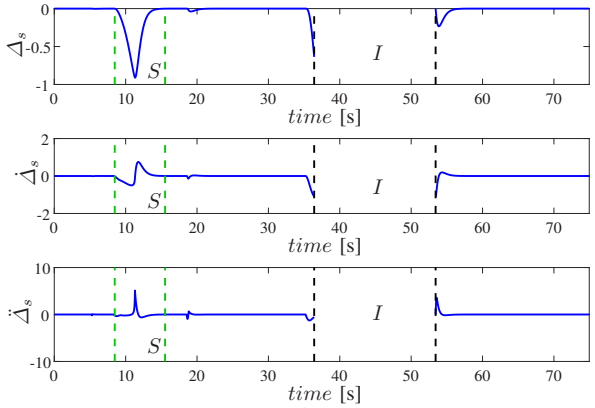


Fig. 7. Evolution of scaling terms Δ_s and its derivatives. Scaling (S) and impedance (I) phases are highlighted; in the impedance phase, no plots are provided since the nominal path is abandoned, thus it makes no sense to consider the relative parameter s_r.

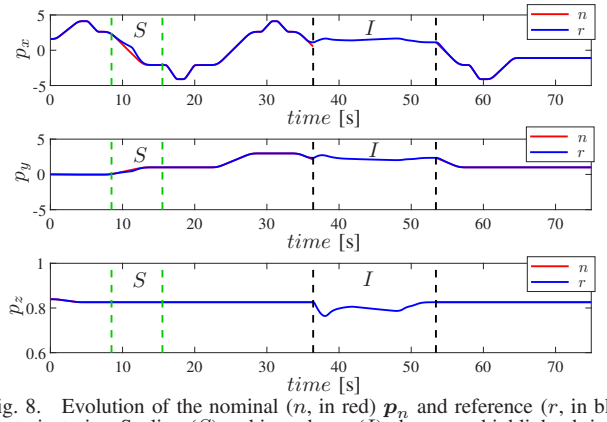


Fig. 8. Evolution of the nominal (n, in red) $\boldsymbol{p}_n$ and reference (r, in blue) $\boldsymbol{p}_r$ trajectories. Scaling (S) and impedance (I) phases are highlighted; in the impedance phase, nominal trajectory is not shown since it loses its meaning.

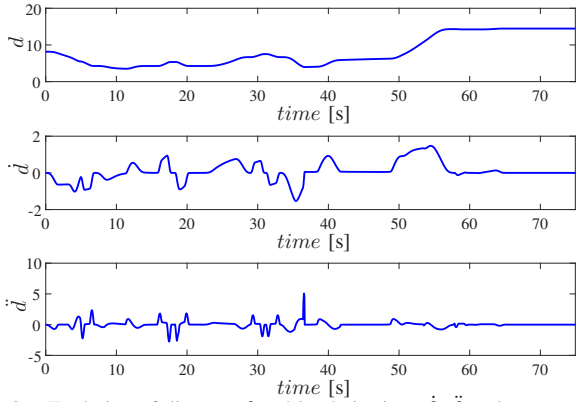


Fig. 9. Evolution of distance d and its derivatives $\dot{d}$, $\ddot{d}$ at the top, middle and bottom, respectively.

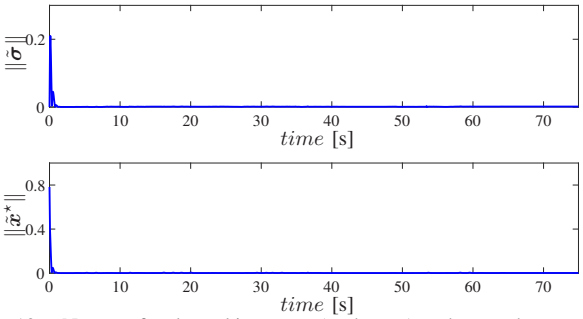


Fig. 10. Norms of task tracking error (at the top) and state observer error (at the bottom).

are responsible for monitoring human motions, and worker robots, which are responsible for performing the effective collaborative task. Knowledge of human motions is then exploited by a human-robot collision avoidance strategy in order to ensure the human safety. To the aim, the cooperative task performed by the workers can be on-line adapted either by modulating the velocity along the nominal path or by deforming it if required.

References

[1] S. Haddadin, A. Albu-Schffer, and G. Hirzinger, "Requirements for safe robots: Measurements, analysis and new insights," *The International Journal of Robotics Research*, vol. 28, no. 11-12, pp. 1507–1527, 2009.

[2] J. A. Marvel, "Performance metrics of speed and separation monitoring in shared workspaces," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 2, pp. 405–414, April 2013.

[3] A. Bicchi and G. Tonietti, "Fast and "soft-arm" tactics [robot arm design]," *IEEE Robotics Automation Magazine*, vol. 11, no. 2, pp. 22–33, June 2004.

[4] S. Haddadin, A. Albu-Schaffer, A. D. Luca, and G. Hirzinger, "Collision detection and reaction: A contribution to safe physical human-robot interaction," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2008, pp. 3356–3363.

[5] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, Mar 1985, pp. 500–505.

[6] S. Haddadin, H. Urbanek, S. Parusel, D. Burschka, J. Romann, A. Albu-Schffer, and G. Hirzinger, "Real-time reactive motion generation based on variable attractor dynamics and shaped velocities," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2010, pp. 3109–3116.

[7] P. R. B. Lacevic and A. M. Zanchettin, "Safety assessment and control of robotic manipulators using danger field," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1257–1270, Oct 2013.

[8] A. M. Zanchettin, N. M. Ceriani, P. Rocco, H. Ding, and B. Matthias, "Safety in human-robot collaborative manufacturing environments: Metrics and control," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 882–893, April 2016.

[9] A. Marino, "Distributed adaptive control of networked cooperative mobile manipulators," *IEEE Transactions on Control Systems Technology*, vol. PP, no. 99, pp. 1–15, 2017.

[10] F. Basile, F. Caccavale, P. Chiacchio, J. Coppola, and C. Curatella, "Task-oriented motion planning for multi-arm robotic systems," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 5, pp. 569–582, 2012.

[11] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton Univiversity Press, 2010.

[12] C. Soto, B. Song, and A. K. Roy-Chowdhury, "Distributed multi-target tracking in a self-configuring camera network," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2009, pp. 1486–1493.

[13] G. W. Y. Zhao, Z. Duan and G. Chen, "Distributed finite-time tracking for a multi-agent system under a leader with bounded unknown acceleration," *Systems & Control Letters*, vol. 81, pp. 8–13, 2015.

[14] U. R. Acharya, K. P. Joseph, N. Kannathal, C. M. Lim, and J. S. Suri, "Heart rate variability: a review," *Medical and Biological Engineering and Computing*, vol. 44, no. 12, pp. 1031–1051, Dec 2006.

[15] O. Dahl and L. Nielsen, "Torque-limited path following by on-line trajectory time scaling," *IEEE Transaction on Robotics and Automation*, vol. 6, pp. 554–561, 1990.