2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Distributed Winner-Take-All Teleoperation of A Multi-Robot System

Yuan Yang[1], Daniela Constantinescu[1] and Yang Shi[1]

***Abstract*— In a distributed multi-master-multi-slave teleoperation system, the human users may compete against each other for the control of the team of slave robots. To win the competition, one operator would send the largest command to the slave group. For the sake of team cohesion, the slave group should follow the command of the winning operator and ignore the commands of the other users. To enable (i) the slave team to identify the winning operator, and (ii) each slave to determine whether to admit or discard the command it receives from its operator, this paper proposes a dynamic decision-making protocol that distinguishes the decision variable of the slave commanded by the winner from the decision variables of all other slave robots. The protocol only requires the slaves to exchange and evaluate their decision variables locally. Lyapunov stability analysis proves the theoretical convergence of the proposed decision-making algorithm. An experimental distributed winner-take-all teleoperation in a 3-masters-11-slaves teleoperation testbed validates its practical efficacy.**

I. Introduction

Multi-robot systems teleoperated by human operators can avail of the human intelligence when making critical decisions and when having to respond fast to unexpected situations [1]. In particular, the multi-master-multi-slave framework can enhance the dexterity, manipulation capability and loading capacity of teleoperation systems [2], [3], and can improve the efficiency of haptic training [4]. Yet, the interaction between cooperative robot teams and human operators may become unstable [5]. Being sufficient for stabilizing robot teams in interaction with passive users, the passivity of the robot team has become a prevalent design objective in teleoperation control [6], [7].

Passivity-based control techniques have tackled slave collisions with other slaves and the remote environment, limited robot communication ranges, and master-slave kinematic dissimilarities [8]. An energy tank-based strategy has created and broken slave interconnections to enable a teleoperated robot group to passively avoid obstacles [9]. A potential function of the estimated algebraic connectivity has offered a decentralized controller for global connectivity preservation of a teleoperated robot team [10]. A hierarchical control with energy tanks with energy flows assigned by port-Hamiltonian analysis has overcome kinematic dissimilarity [11].

Recent research has optimized the interaction of multi-robot systems with external terminators subject to system passivity. In [12], a quadratic optimization constrained by proper scaling of inter-robot connections has regulated passive interaction forces between a robot group and external entities. In [13], an energy tank-based optimization has adapted the admittance of the robot team subject to a condition on the lower bound of energy storage that guarantees passivity. In [14], a generalized two-layer approach has let all masters and slaves share an energy tank to minimize the conservativeness of maintaining a multi-master-multi-slave teleoperator passive. In [15], an energy transaction protocol has distributed energy budgets passively in a multi-robot team.

The aforementioned works share a lower-level control objective: to render a teleoperated multi-robot system passive. Yet, multi-user teleoperation introduces an additional higher-level decision-making problem rarely covered [16]: a multi-slave team must determine how to respond to simultaneous commands from multiple users [17]. Relevant work has mainly tracked the average of the multiple exogenous inputs (user commands in multi-user teleoperation). In [18], a filter-based algorithm with state-dependent gains has enabled dynamic average tracking control of nonlinear second-order multi-agent systems. In [19], a robust approach has guaranteed arbitrarily small steady-state error of dynamic average consensus for directed networks. In [20], an integral-based dynamic consensus technique has robustly filtered the average of only nonzero sensor measurements.

This paper introduces a distributed decision-making mechanism that enables a teleoperated robot group to identify, and carry out the commands of only, the winner among its multiple operators. Inspired by Oja's rule for principal component analysis [21], the proposed strategy endows each slave with a decision variable whose evolution depends on (i) the input from its local operator and (ii) the decision variables of its neighbouring slaves. Compared to [22], the distributed decision-making layer can handle the time-varying operator commands arising in teleoperation by exploiting robust dynamic average estimations. Lyapunov energy analysis proves that the decision variables of all slaves in communication with the winner converge to a positive limit set and all other decision variables asymptotically approach zero when the winner is unique. In other words, the proposed decision-making strategy enables a multi-robot team teleoperated by multiple users to distinguish the winner among all its operators by evaluating only local decision variables. Winner-take-all teleoperation follows by permitting only the slaves with decision variables above a suitable threshold to apply their received operator commands: the slave team is teleoperated exclusively by the winner. Experimental winner-take-all teleoperation in a 3-masters-11-slaves testbed validates the performance of the distributed decision-making algorithm.

[1] The authors are with the Department of Mechanical Engineering, University of Victoria, Victoria, BC V8W 2Y2 Canada (e-mail: yangyuan@uvic.ca; danielac@uvic.ca; yshi@uvic.ca).

978-1-7281-7395-5/20/$31.00 ©2020 IEEE

II. Problem Formulation

Let a multi-user teleoperation system include $N_m > 1$ masters and $N_s \geq N_m$ slaves, all gravity compensated. Without loss of generality, let the slave team be connected and each slave receive information from at most one master.

The task-space dynamics of the master robots are:

$$\mathbf{M}_{mi}(\mathbf{y}_{mi})\ddot{\mathbf{y}}_{mi} + \mathbf{C}_{mi}(\mathbf{y}_{mi}, \dot{\mathbf{y}}_{mi})\dot{\mathbf{y}}_{mi} = \mathbf{f}_{hi} - \mathbf{u}_{mi}, \tag{1}$$

where $i = 1, \cdots, N_m$ indexes all human users and their master devices. For each master i: $\mathbf{y}_{mi}$, $\dot{\mathbf{y}}_{mi}$ and $\ddot{\mathbf{y}}_{mi}$ are its position, velocity and acceleration; $\mathbf{M}_{mi}(\mathbf{y}_{mi})$, $\mathbf{C}_{mi}(\mathbf{y}_{mi}, \dot{\mathbf{y}}_{mi})$ are its matrices of inertia and of Coriolis and centrifugal effects; $-\mathbf{u}_{mi}$ and $\mathbf{f}_{hi}$ are the forces applied by its own controller and by its user, respectively. The control forces:

$$\mathbf{u}_{mi} = K_{mp}\mathbf{y}_{mi} + K_{md}\dot{\mathbf{y}}_{mi}, \tag{2}$$

with K_{mp} and K_{md} well-tuned positive gains, (i) provide force feedback to the operators and (ii) transmit the operator commands to the N_{ls} active slaves, where $N_m \leq N_{ls} \leq N_s$.

To synchronize the slaves: (i) connect each slave to an own second-order virtual proxy; and (ii) couple the proxies of adjacent slaves. Then, the slave group dynamics become:

$$\begin{aligned} \hat{m}_{si}\ddot{\hat{\mathbf{y}}}_{si} &= \sum_{j\sim i} \hat{K}_{sp}(\hat{\mathbf{y}}_{sj} - \hat{\mathbf{y}}_{si}) - \hat{K}_{sd}\dot{\hat{\mathbf{y}}}_{si} + \mathbf{f}_{si}, \\ \mathbf{M}_{si}(\mathbf{y}_{si})\ddot{\mathbf{y}}_{si} + \mathbf{C}_{si}(\mathbf{y}_{si}, \dot{\mathbf{y}}_{si})\dot{\mathbf{y}}_{si} &= K_{sp}\tilde{\mathbf{y}}_{si} - K_{sd}\dot{\mathbf{y}}_{si}, \end{aligned} \tag{3}$$

where the index $i = 1, \cdots, N_{ls}$ indicates the active slaves, i.e., slaves that receive user commands from masters, $i = N_{ls} + 1, \cdots, N_s$ indicates asleep slaves, i.e., slaves that communicate only to other slaves, and $j \sim i$ indicates the set of neighbours of the slave i, i.e., slaves j that exchange information with slave i. For every slave i: $\hat{m}_{si}$, $\hat{\mathbf{y}}_{si}$, $\dot{\hat{\mathbf{y}}}_{si}$ and $\ddot{\hat{\mathbf{y}}}_{si}$ are the mass, position, velocity and acceleration of its virtual proxy; $\tilde{\mathbf{y}}_{si} = \hat{\mathbf{y}}_{si} - \mathbf{y}_{si}$ is the displacement of its proxy from it; and $\mathbf{M}_{si}(\mathbf{y}_{si})$, $\mathbf{C}_{si}(\mathbf{y}_{si}, \dot{\mathbf{y}}_{si})$, $\mathbf{y}_{si}$, $\dot{\mathbf{y}}_{si}$ and $\ddot{\mathbf{y}}_{si}$ are the same quantities as in (1) but for slave robots; and $\mathbf{f}_{si}$ is determined by the proposed decision-making strategy. As proven in [5, Chapter 4.2.4], the intrinsically passive control using second-order proxies with positive constant gains $\hat{K}_{sp}$, $\hat{K}_{sd}$, K_{sp} and K_{sd} can ensure passive teleoperation.

The proposed decision-making strategy associates a decision variable x_i with each slave and determines $\mathbf{f}_{si}$ in (3) as follows: $\mathbf{f}_{si} = \mathbf{0}$ for each asleep slave $i = N_{ls} + 1, \cdots, N_s$, because asleep slaves communicate with no master; $\mathbf{f}_{si} = \mathbf{u}_{mj}$ for the active slave i commanded by the winner operator j, i.e., the active slave with decision variable x_i above a threshold; and $\mathbf{f}_{si} = \mathbf{0}$ for all active slaves i who communicate with the remaining users, i.e., active slaves with decision variables x_i below the threshold. Identifying the winner among all users then lies in evolving the decision variables x_i appropriately. The decision-making mechanism evolving x_i is the primary concern of this paper.

In the absence of a decision-making strategy, all users can teleoperate the slave team from their different locations. Their commands $\mathbf{u}_{mi}$ can stand for the degree of their demands to steer the slave group: user i may increase their command $\mathbf{u}_{mi}$ when they intend to gain more control of the slave network. In Fig. 1, for example, users $i = 1, 2, 3$ all send commands to the slave team and may compete to dominate the teleoperation of the multi-slave group. Suppose all operator commands are applied to the slave network, i.e., all switches are closed. If user 1 sought to control the network regardless of opposition from users 2 and 3, user 1 would keep increasing their command $\mathbf{u}_{m1}$ to conquer $\mathbf{u}_{m2}$ and $\mathbf{u}_{m3}$, while users 2 and 3 would seek to resits and thus slow the movement of the slave team and threaten its coordination.

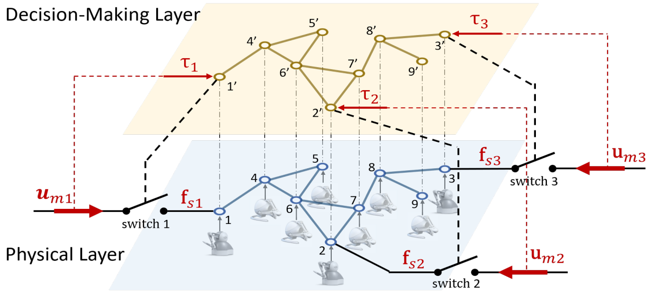

Fig. 1. A slave robot team, with 3 active slaves (Geomagic Touch robots) and 6 asleep slaves (Novint Falcon robots), teleoperated by 3 users.

To guarantee the agility and cohesion of the slave group, this paper proposes winner-take-all teleoperation, which permits only the winner user to teleoperate the slave network. Winner-take-all teleoperation requires a decision-making mechanism that enables all slaves to recognize the winner and ignore the commands of all other operators. Hence, this paper develops a distributed decision-making protocol that permits each slave to detect whether it receives commands from the winner by evaluating its own decision variable based only on information from its neighbouring slaves.

The proposed protocol selects the user with the largest force command as the winner. To this end, it endows each slave i with a dynamic decision variable x_i and builds a decision-making layer whose inputs τ_i encode the magnitudes of the commands $\mathbf{u}_{mj}$ received from the operators via the following strictly increasing function:

$$\tau_i = \exp(\|\mathbf{u}_{mj}\|/5), \tag{4}$$

where $i = 1, \cdots, N_s$, $j = 1, \cdots, N_m$, and $\mathbf{u}_{mj}$ is given in (2) for $i = 1, \cdots, N_{ls}$ and $\mathbf{u}_{mj} = \mathbf{0}$ for $i = N_{ls} + 1, \cdots, N_s$. Note that $N_{ls} \geq N_m$ and thus one operator may send force commands to more than one slave. The experiment in Section IV shows an example where $N_m = 3$ and $N_{ls} = 4$ and τ_2 and τ_3 are encodings of a same user command $\mathbf{u}_{m2}$. Because every x_i evolves locally at its corresponding slave, the communication graph of the decision layer is isomorphic to that of the slave network and, thus, it is connected.

Each master sends its operator's command to different slaves. Hence, each slave robot needs to determine whether its master is operated by the winner user by verifying its decision variable. Let each decision variable x_i evolve according to τ_i and to the states of neighbouring vertices $j' \sim i'$. Let user k be the winner and send the largest force command $\mathbf{u}_{mk}$ among all users, $\|\mathbf{u}_{mk}\| > \|\mathbf{u}_{mi}\|, \forall i = 1, \cdots, N_m$ and $i \neq k$. Then, the decision-making layer drives the decision variables of all active slaves receiving the winner's command to a positive limit set and reduces the other decision variables to zero. Further, only slaves whose decision variables are above a certain threshold execute their received user commands. Hence, the winner can steer the slave network. For example, let user 1 be the winner in Fig. 1, i.e., $\|\mathbf{u}_{m1}\| > \max(\|\mathbf{u}_{m2}\|, \|\mathbf{u}_{m3}\|)$ and thus $\tau_1 > \tau_i$ for $i = 2, \cdots, 9$ by (4). Then x_1 converges to a positive limit set while the other x_i-s approach zero. In turn, the active slave 1 closes switch 1 and executes $\mathbf{f}_{s1} = \mathbf{u}_{m1}$ and the active slaves 2 and 3 open switches 2 and 3 and ignore their received user commands $\mathbf{u}_{m2}$ and $\mathbf{u}_{m3}$, i.e., $\mathbf{f}_{s2} = \mathbf{f}_{s3} = \mathbf{0}$. As a result, the slave network is driven only by the force command $\mathbf{u}_{m1}$ of the winning user 1.

III. Main Results

To identify the winner, let the evolution law of the decision variable x_i of each slave i be:

$$\dot{x}_i = \sigma \cdot (N_s \tau_i x_i - \eta_i x_i), \tag{5}$$

where the subscript $i = 1, \cdots, N_s$ indexes the vertex i' in the decision-making layer, τ_i is given in (4), $\sigma > 0$, and:

$$\eta_i = \alpha_i + N_s \tau_i x_i^2 \tag{6}$$

is a dynamically-updated gain with:

$$\dot{\alpha}_i = \sum_{j \sim i} w_{ij}(t) \cdot (\eta_j - \eta_i) + h_{ij}(t) \cdot \text{sign}(\eta_j - \eta_i) \tag{7}$$

with $w_{ij}(t) = w_{ji}(t)$ and $h_{ij}(t) = h_{ji}(t)$ to be designed.

Let M be the number of communication links in the slave group. Then, the incidence matrix $\mathbf{D}$ and the unweighted Laplacian matrix $\overline{\mathbf{L}}$ of the decision-making network are defined the same as in [23, Chapter 2]. Let $\mathbf{x} = [x_1, \cdots, x_{N_s}]^\mathsf{T}$, $\boldsymbol{\tau} = [\tau_1, \cdots, \tau_{N_s}]^\mathsf{T}$ and $\boldsymbol{\eta} = [\eta_1, \cdots, \eta_{N_s}]^\mathsf{T}$. Let ” $\circ$ ” be an operator that takes the product of two vectors componentwisely (their Haddamard product). The p-th power of a vector $\mathbf{v} = [v_1, \cdots, v_{N_s}]^\mathsf{T}$ is thus accordingly $\mathbf{v}^p = [v_1^p, \cdots, v_{N_s}^p]^\mathsf{T}$. Note that ” $\circ$ ” has higher precedence than ” $\cdot$ ” in algebraic operations. The dynamics of the overall decision-making layer then become:

$$\begin{aligned} \dot{\mathbf{x}} =& \sigma \cdot (N_s \boldsymbol{\tau} \circ \mathbf{x} - \boldsymbol{\eta} \circ \mathbf{x}), \\ \dot{\boldsymbol{\eta}} =& -\mathbf{L}(t) \cdot \boldsymbol{\eta} - \mathbf{D} \cdot \mathbf{H}(t) \cdot \text{sign}(\mathbf{D}^\mathsf{T} \cdot \boldsymbol{\eta}) + N_s \dot{\boldsymbol{\tau}} \circ \mathbf{x}^2 \\ & + 2\sigma N_s \boldsymbol{\tau} \circ \mathbf{x}^2 \circ (N_s \boldsymbol{\tau} - \boldsymbol{\eta}), \end{aligned} \tag{8}$$

where $\mathbf{L}(t) = \mathbf{D} \cdot \mathbf{W}(t) \cdot \mathbf{D}^\mathsf{T}$ with $\mathbf{W}(t) = \text{diag}\{w_{ij}(t)\}$ and $\mathbf{H}(t) = \text{diag}\{h_{ij}(t)\}$.

In (8), $\boldsymbol{\eta}$ is dynamically updated to estimate $\boldsymbol{\tau}^\mathsf{T} \cdot \mathbf{x}^2 \cdot \mathbf{1}$, where $\mathbf{1} = [1, \cdots, 1]^\mathsf{T}$. By (8), the estimating error:

$$\boldsymbol{\delta} = \boldsymbol{\eta} - \boldsymbol{\tau}^\mathsf{T} \cdot \mathbf{x}^2 \cdot \mathbf{1} \tag{9}$$

evolves according to:

$$\begin{aligned} \dot{\boldsymbol{\delta}} =& -\mathbf{L}(t) \cdot \boldsymbol{\delta} - \mathbf{D} \cdot \mathbf{H}(t) \cdot \text{sign}(\mathbf{D}^\mathsf{T} \cdot \boldsymbol{\delta}) + \overline{\mathbf{L}}_c \cdot (\dot{\boldsymbol{\tau}} \circ \mathbf{x}^2) \\ & + 2\sigma \overline{\mathbf{L}}_c \cdot \left[\boldsymbol{\tau} \circ \mathbf{x}^2 \circ (N_s \boldsymbol{\tau} - \boldsymbol{\eta})\right], \end{aligned} \tag{10}$$

where $\overline{\mathbf{L}}_c = N_s \mathbf{I} - \mathbf{1} \cdot \mathbf{1}^\mathsf{T}$ is the unweighted Laplacian of a complete graph of order N_s. Note that there exists a positive semi-definite $\mathbf{T}$ such that $\overline{\mathbf{L}}_c = \mathbf{D} \cdot \mathbf{T} \cdot \mathbf{D}^\mathsf{T}$.

The Lyapunov candidate constructed to study the convergence of the decision-making algorithm (5)-(7) is:

$$V = \frac{1}{4}(\mathbf{x}^\mathsf{T} \cdot \mathbf{x} - N_s)^2 + \frac{\epsilon}{2} \boldsymbol{\delta}^\mathsf{T} \cdot \boldsymbol{\delta}, \tag{11}$$

where $\epsilon > 0$ will be determined. The first term in (11) measures the deviation from 1 of the decision variables x_i, while the second term quantifies the impact of the error of estimation $\boldsymbol{\delta}$.

Define $\boldsymbol{\upsilon} = [\upsilon_1, \cdots, \upsilon_M]^\mathsf{T}$ with υ_i the maximum absolute value of all elements in the i-th row of $\mathbf{T} \cdot \mathbf{D}^\mathsf{T}$. Let k index the oriented link (i, j) defined by the selection of the incidence matrix $\mathbf{D}$ [23, Chapter 2]. Further, choose $h_{ij}(t)$ by:

$$h_{ij}(t) = N_s \sigma \tilde{\tau} \upsilon_k \cdot (\eta_i + \eta_j), \tag{12}$$

where $\tilde{\tau} = \overline{\tau} - 1$ with $\overline{\tau} > 1$ the upper bound of all encodings τ_i in (4), and select $w_{ij}(t)$ by:

$$\begin{aligned} w_{ij}(t) =& \Bigg[\upsilon_k^2 \cdot (\eta_i + \eta_j) + K_c K_2 M \cdot (\eta_i - \eta_j)^2 \\ & + \frac{K_c \upsilon_k^4}{K_2}\Bigg] \times \left(2M\sigma\epsilon\overline{\tau}^2 + \frac{M\sigma}{4N_s^2\epsilon} + \frac{M\sigma}{4N_s}\right) \\ & + \frac{4M^2\tilde{\tau}^2\upsilon_k^2}{N_s K_1} + \frac{N_s \sigma^2 K_1 \upsilon_k^2}{4} + \varrho K_c^2, \end{aligned} \tag{13}$$

where K_c is the maximum singular value of $\mathbf{D} \cdot \mathbf{T}/N_s$, and ϱ, K_1 and K_2 are positive constants. Note that, the adaptations of the control gains $h_{ij}(t)$ in (12) and w_{ij} in (13) require some quantities relevant to network topology. In practice, engineers can select, or flooding algorithms [24] can identify, the communication network when initializing the teleoperation system. The proposed design is thus distributed in the sense of implementation.

As in [25, pp.3-6], the derivative of V can then be upper-bounded by:

$$\dot{V} \leq -\kappa(t) \cdot V - \epsilon\varrho\boldsymbol{\delta}^\mathsf{T} \cdot \boldsymbol{\delta} + \kappa(t) \cdot \chi\left(\|\dot{\boldsymbol{\tau}}(t)\|^2\right), \tag{14}$$

where the dependence of $\kappa(t)$ and $\chi\left(\|\dot{\boldsymbol{\tau}}(t)\|^2\right)$ on time is highlighted for the ease of obtaining (15), and:

$$\kappa(t) = \sigma\boldsymbol{\tau}^\mathsf{T} \cdot \mathbf{x}^2 \quad \text{and} \quad \chi\left(\|\dot{\boldsymbol{\tau}}(t)\|^2\right) = \frac{N\epsilon\|\dot{\boldsymbol{\tau}}\|^2}{8\sigma^2}.$$

Time integration of (14) from $t_0 \geq 0$ to $t \geq t_0$ yields:

$$V(t) \leq \exp\left(-\int_{t_0}^{t} \kappa(\theta) d\theta\right) \cdot V(t_0) + \sup_{t_0 \leq \theta \leq t} \chi\left(\|\dot{\boldsymbol{\tau}}(\theta)\|^2\right). \tag{15}$$

The following lemma is key to proving the convergence of the decision-making algorithm in Theorem 1:

Lemma 1. *Let the dynamics* (5) *of the decision variables x_i start from $x_i(0) = 1$, with gains updated by (6)-(7) and $\alpha_i(0) = 0, \forall i = 1, \cdots, N_s$. Then, the dynamic modulations* (12) *and* (13)*, of $h_{ij}(t)$ and $w_{ij}(t)$, guarantee that:*

1. *The state $\mathbf{x}$ and the error of estimation $\boldsymbol{\delta}$ stay in the invariant set:*
$$\mathcal{I} = \left\{ \left(\|\mathbf{x}\|^2 - N_s\right)^2 + 2\epsilon\|\boldsymbol{\delta}\|^2 \le 2N_s^2\epsilon\overline{\tau}^2 + \frac{N_s\epsilon\overline{\dot{\tau}}}{2\sigma^2} \right\},$$
where $\overline{\dot{\tau}}$ is the upper bound of $\dot{\tau}_i$;
2. *If ϵ is selected by:*
$$\epsilon \le 2\sigma^2(N_s - \rho)^2 / \left(4N_s^2\sigma^2\overline{\tau}^2 + N_s\overline{\dot{\tau}}\right), \tag{16}$$
with $0 < \rho < N_s$, then $\mathbf{x}$ and $\boldsymbol{\delta}$ exponentially converge to the attractive set:
$$\mathcal{A} = \left\{ \left(\|\mathbf{x}\|^2 - N_s\right)^2 + 2\epsilon\|\boldsymbol{\delta}\|^2 \le \frac{N_s\epsilon\overline{\dot{\tau}}}{2\sigma^2} \right\}$$
and the rate of convergence is $\sigma\rho$.

Proof. 1. Let $t_0 = 0$. The definition of V (11) and (15) yield:

$$\left(\|\mathbf{x}\|^2 - N_s\right)^2 + 2\epsilon\|\boldsymbol{\delta}\|^2 \le 4V(0) + \frac{N_s\epsilon\overline{\dot{\tau}}}{2\sigma^2}. \tag{17}$$

The initializations $x_i(0) = 1$ and $\alpha_i(0) = 0$ guarantee that:

$$V(0) = \frac{\epsilon}{2} \cdot \boldsymbol{\tau}^{\mathsf{T}}(0) \cdot \overline{\mathbf{L}}_c \cdot \overline{\mathbf{L}}_c \cdot \boldsymbol{\tau}(0) \le \frac{N_s^2\epsilon\overline{\tau}^2}{2}, \tag{18}$$

because 0 and N_s are eigenvalues of $\overline{\mathbf{L}}_c$. Together, Equations (17)-(18) imply that $\mathbf{x}$ and $\boldsymbol{\delta}$ are bounded within $\mathcal{I}$.

2. The invariant set $\mathcal{I}$ indicates that:

$$\|\mathbf{x}\|^2 \ge N_s - \sqrt{2N_s^2\epsilon\overline{\tau}^2 + \frac{N_s\epsilon\overline{\dot{\tau}}}{2\sigma^2}}.$$

The selection of ϵ (16) then guarantees that $\|\mathbf{x}\|^2 \ge \rho$ and thus $\kappa(t) \ge \sigma\rho$ for all time. By (11) and (15), it follows that:

$$\begin{aligned}&\left(\|\mathbf{x}\|^2 - N_s\right)^2 + 2\epsilon\|\boldsymbol{\delta}\|^2 \\ \le& 4\exp(-\sigma\rho t) \cdot V(0) + 4\sup_{0\le\theta\le t} \chi\left(\|\dot{\boldsymbol{\tau}}(\theta)\|^2\right),\end{aligned} \tag{19}$$

which proves the exponential convergence to $\mathcal{A}$. ■

The following theorem summarizes the convergence of the proposed decision-making algorithm.

Theorem 1. *Let the distributed decision-making layer (5)-(7) wih time-varying encodings τ_i of user inputs be initialized by $x_i = 1$ and $\alpha_i = 0$, and let $\Delta_\tau > 0$ be a preset resolution of the decision-making algorithm. Further: (i) choose σ by:*

$$\sigma \ge \sqrt{N_s\overline{\dot{\tau}}} \Big/ (p\Delta_\tau), \tag{20}$$

where $p < 1$; (ii) select ϵ by (16)*; and (iii) update the parameters $h_{ij}(t)$ by* (12) *and $w_{ij}(t)$ by* (13) *for all $i \sim j$. Then, if there exists k such that $\tau_k(t) \ge \Delta_\tau + \tau_i(t)$ for all $t \ge t_0 \ge 0$ and for all $i = 1, \cdots, N_s$ with $i \neq k$, the decision variable $x_k(t)$ asymptotically converges to below the positive limit set:*

$$\mathcal{W}_k = \left\{ x_k \ge w = \max\left(N_s - p\Delta_\tau\sqrt{\epsilon/2}, \sqrt{\rho}\right) \right\} \tag{21}$$

while $x_i(t) \to 0$ asymptotically as $t \to +\infty$ for all $i \neq k$.

Proof. The selection (20) σ leads to:

$$\sup_{0\le\theta\le t} \chi\left(\|\dot{\boldsymbol{\tau}}(\theta)\|^2\right) \le \epsilon p^2\Delta_\tau^2/8.$$

From (19), the selection (16) of ϵ further ensures that:

$$2\|\boldsymbol{\delta}\| \le \sqrt{8V(0)/\epsilon} \cdot \exp(-\sigma\rho t/2) + p\Delta_\tau.$$

The variation of all decision variables is studied using the quotient function:

$$E = \sum_{i\neq k} \frac{x_i}{x_k}, \tag{22}$$

where i and k are defined in the theorem. By (5), x_k and all x_i-s are positive for all time as they start from 1. Further, because $\mathbf{x}$ is bounded by $\mathcal{I}$, E becomes zero if and only if $x_i = 0$ for all $i \neq k$. Then, the derivative of E along the dynamics of the decision-making layer is:

$$\begin{aligned}\dot{E} =& \sum_{i\neq k} \left[(\tau_i - \tau_k)\frac{x_i}{x_k} + (\delta_k - \delta_i)\frac{x_i}{x_k} \right] \\ \le& -(\Delta_\tau - 2\|\boldsymbol{\delta}\|) \cdot \sum_{i\neq k} \frac{x_i}{x_k} \\ \le& \left[-(1-p)\Delta_\tau + \sqrt{8V(0)/\epsilon} \cdot \exp(-\sigma\rho t/2) \right] \cdot E.\end{aligned} \tag{23}$$

Further, time integration of $\dot{E}$ gives that, as $t \to +\infty$,

$$\begin{aligned}E(t) \le& \exp\Bigg[-(1-p)\Delta_\tau(t - t_0) + \sqrt{8V(0)/\epsilon} \\ & \times \int_{t_0}^{t} \exp(-\sigma\rho\theta/2)d\theta \Bigg] \cdot E(0) \\ \le& \exp\Bigg[-(1-p)\Delta_\tau t + 4\sqrt{2V(0)/\epsilon} \Big/ (\sigma\rho) \\ & \times \exp(-\sigma\rho t_0/2) \Bigg] \cdot E(0) \to 0,\end{aligned} \tag{24}$$

which, together with $E(t) \ge 0$, ensures that $E(t) \to 0$. By $\mathcal{I}$ in Lemma 1, $\mathbf{x}$ is bounded, and thus $E(t) \to 0$ indicates that $x_i(t) \to 0$ for every $i \neq k$. Because $\mathbf{x}$ asymptotically converges to $\mathcal{A}$, $x_i(t) \to 0$ for all $i \neq k$ further means that $x_k(t)$ converges to:

$$\mathcal{W}_{k1} = \left\{ \left(x_k^2 - N_s\right)^2 \le N_s\epsilon\overline{\dot{\tau}} / \left(2\sigma^2\right) \le \epsilon p^2\Delta_\tau^2/2 \right\}.$$

The proof of the second item of Lemma 1 also shows that (16) makes $\|\mathbf{x}\|^2 \ge \rho$. Because x_k and all x_i-s remain positive throughout time, it follows that:

$$x_k(t) + \sum_{i\neq k} x_i(t) \ge \|\mathbf{x}(t)\| \ge \sqrt{\rho}.$$

As a result, $x_i(t) \to 0$ for every $i \neq k$ indicates that $x_k(t)$ also approaches:

$$\mathcal{W}_{k2} = \left\{ x_k \ge \sqrt{\rho} > 0 \right\}.$$

Thus, $x_k(t)$ asymptotically converges to $\mathcal{W}_k \supseteq \mathcal{W}_{k1} \cap \mathcal{W}_{k2}$. ■

Compared to [22], the decision-making layer (5)-(7) with the dynamic gains (12) and (13) is robust against time-varying user inputs and, therefore, demands a distinct convergence analysis of the proposed network. As shown in Lemma 1, the design first explicitly upper bounds the state $\mathbf{x}$ and the estimation error $\boldsymbol{\delta}$, and then devises the selection of ϵ to force their exponential convergence to the attractive set $\mathcal{A}$. This convergence is critical for distinguishing the decision variable of the winner from those of other users via the quotient function (22) in the proof of Theorem 1.

The convergence of the proposed decision-making strategy when the winner commands multiple slaves can be proven similarly and is omitted due to space constraints. By Theorem 1, the decision variables of active slaves receiving commands from the winner converge to a positive limit set while the decision variables of all other slaves approach zero. By comparing its decision variable x_i to a threshold $x < w$, each slave i can decide if it receives commands from the winner: yes if $x_i \geq x$ and no otherwise. Future work will investigate the performance of the proposed design when several users apply equally large forces.

IV. Experimental Results

Fig. 2 shows the distributed 3-masters-11-slaves teleoperation testbed used to validate the performance of the proposed decision-making algorithm. It comprises: 3 black Novint Falcon robots, the master devices, operated by 3 volunteers; 4 Geomagic Touch robots, the active slaves, receiving user commands from the masters; and 7 white Novint Falcon robots, the asleep slaves, that communicate only with other slaves. Each robot is locally controlled via USB 2.0 by a C++ program on a dedicated Linux machine. All Linux machines access the Internet using a 16-port network switch (with bandwidth 32 Gbps), and run the Robot Operating System (ROS) to support distributed robot communications. In Fig. 2, the unidirectional dashed arrows depict the transmissions of user commands from masters to active slaves, and the bidirectional solid arrows indicate the information exchanges among the slaves. Robot sensing, communication and control run at 1 kHz. The additional computer at the bottom of Fig. 2 records the experimental data at 100 Hz and displays the movements of the end-effectors of all slave robots. A video of the experiment is available at `https://youtu.be/-XOGMFwi8_s`.

The experiment has two phases: (i) during the first 5 minutes, the users apply forces to the slave group sequentially, starting with user 3 and ending with user 1; (ii) during the next 3.5 minutes, they apply forces to the slave network simultaneously. Fig. 3-Fig. 5 plot the time histories of all operator forces. The shaded areas are time intervals when the respective user applies the largest force among all users.

Fig. 6 plots the encodings of the user commands to the decision-making layer. Note that τ_2 and τ_3 overlap because user 2 applies their force to two active slaves, 2 and 3. Initially, users 3, 2 and 1 apply forces sequentially in this order, and the encodings τ_4, τ_3 (τ_2), and τ_1 become larger than 1 during the time intervals T_1, T_2 and T_3, respectively.

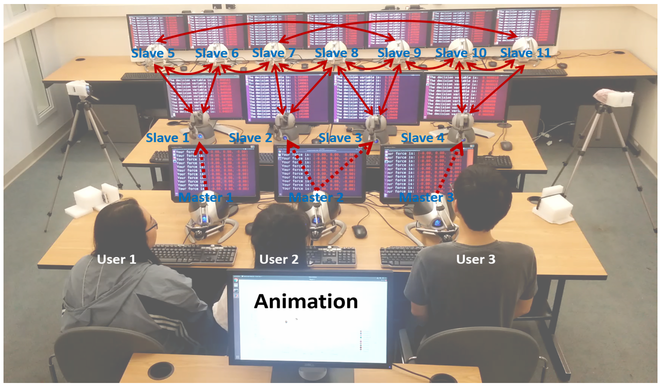

Fig. 2. The distributed 3-masters-11-slaves teleoperation testbed.

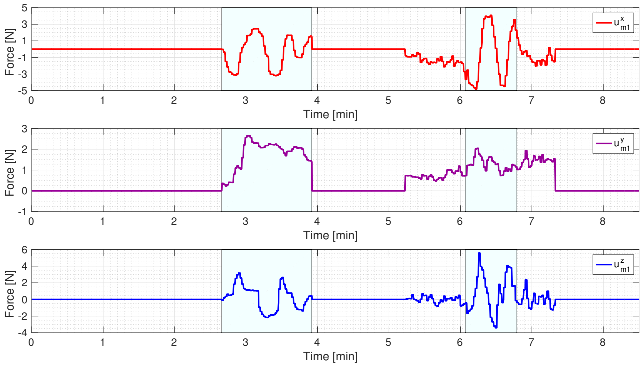

Fig. 3. The force command $\mathbf{u}_{m1}$ of user 1 along x-, y-, and z-axes.

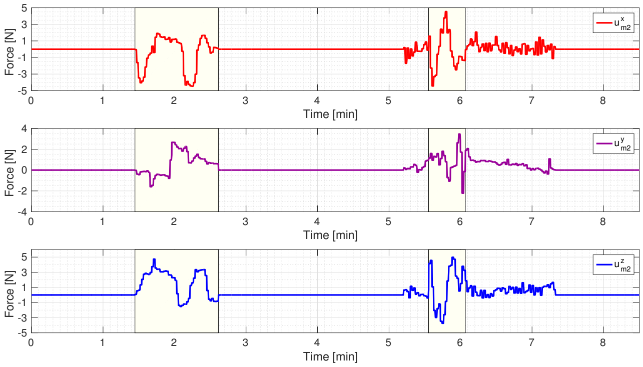

Fig. 4. The force command $\mathbf{u}_{m2}$ of user 2 along x-, y-, and z-axes.

Then, all users apply forces simultaneously, so all encodings become larger than 1 during all time intervals T_4 to T_7. However, only one user applies the largest force during each of these time intervals, see the command encodings in Fig. 6.

Because communication delays are up to 36 ms in the experiment, the damping $-10\alpha_i$ is injected in (7) to practically stabilize the decision-making algorithm, and stabilization by damping injection is currently under theoretical study. The time histories of the practically stabilized decision variables are depicted in Fig. 7 for the active slaves, and in Fig. 8 for the asleep slaves. Fig. 7 shows that the decision variables of active slaves grow larger than 2 when the active slaves

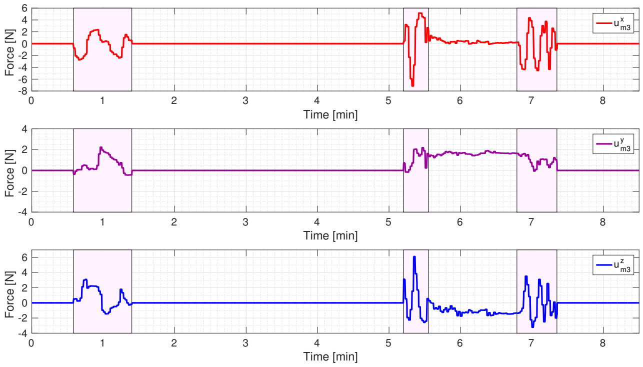

Fig. 5. The force command $\mathbf{u}_{m3}$ of user 3 along x-, y-, and z-axes.

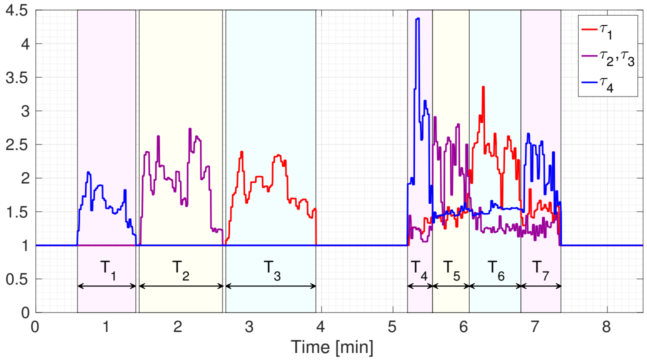

Fig. 6. The encodings of user commands to the decision-making layer.

receive force commands from the winner. Further, Fig. 7-Fig. 8 illustrate that: (i) the decision variables of all slaves approach 0 unless they receive force commands from the winner; and (ii) if the slave network receives no user command, i.e., $\tau_i = 1$ for all i, then all decision variables return to 1. Therefore, the threshold x for winner selection is set to 2. Lastly, all decision variables x_i stop decreasing when they reach 0.001 to avoid zero crossings in the discrete-time implementation of the decision-making algorithm.

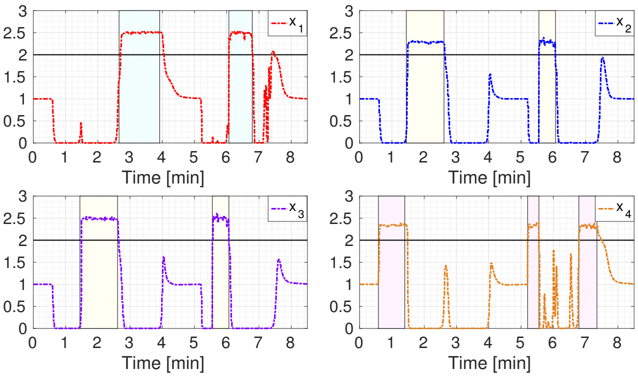

Fig. 7. The decision variables of the active slave robots.

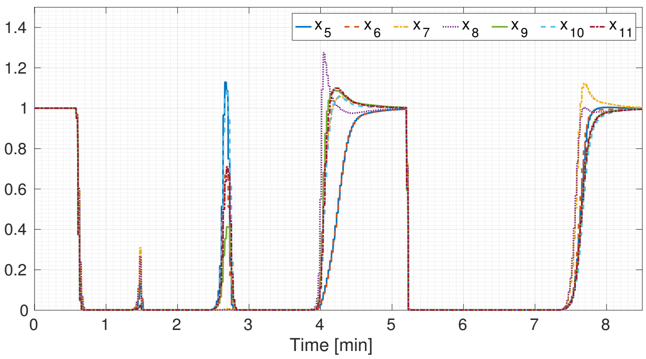

Fig. 8. The decision variables of the asleep slave robots.

V. Conclusions

This paper has presented a distributed decision-making strategy for winner-take-all teleoperation of multi-robot systems. The strategy equips every slave robot with a decision variable which it adjusts based on the variations (i) of the force command received from the local operator and (ii) of the decision variables of the neighbouring slaves. Lyapunov stability analysis shows that the strategy can drive the decision variables of the active slaves who receive force commands from the winner to a positive limit set and reduce other decision variables to zero. It thus enables the slave group to identify the winner in a decentralized manner. An experimental winner-take-all teleoperation in a distributed 3-masters-11-slaves teleoperation testbed has illustrated the efficacy of the proposed decision-making strategy. Upcoming research will make the algorithm robust to time-varying communication delays and integrate the decision-making layer into popular cooperative teleoperation schemes.

References

[1] Q. M. Ta, S. Lyu, and C. C. Cheah, "Human-guided optical manipulation of multiple microscopic objects," in *2018 IEEE International Conference on Robotics and Automation*. IEEE, 2018, pp. 1–5.

[2] M. Shahbazi, S. F. Atashzar, and R. V. Patel, "A systematic review of multilateral teleoperation systems," *IEEE Transactions on Haptics*, vol. 11, no. 3, pp. 338–356, July 2018.

[3] J. Ryu, Q. Ha-Van, and A. Jafari, "Multilateral teleoperation over communication time delay using the time-domain passivity approach," *IEEE Transactions on Control Systems Technology*, pp. 1–8, 2019.

[4] A. R. Licona, A. Lelevel, M. T. Pham, and D. Eberard, "A multi-trainee architecture for haptic hands-on training," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov 2019, pp. 7314–7320.

[5] C. Secchi, S. Stramigioli, and C. Fantuzzi, *Control of Interactive Robotic Interfaces: A Port-Hamiltonian Approach*. Springer Science & Business Media, 2007, vol. 29.

[6] M. Selvaggio, P. Robuffo Giordano, F. Ficuciellol, and B. Siciliano, "Passive task-prioritized shared-control teleoperation with haptic guidance," in *2019 IEEE International Conference on Robotics and Automation*. IEEE, 2019, pp. 430–436.

[7] E. Sartori, C. Tadiello, C. Secchi, and R. Muradore, "Tele-echography using a two-layer teleoperation algorithm with energy scaling," in *2019 IEEE International Conference on Robotics and Automation*. IEEE, 2019, pp. 1569–1575.

[8] D. Lee, A. Franchi, P. R. Giordano, H. I. Son, and H. H. Bülthoff, "Haptic teleoperation of multiple unmanned aerial vehicles over the internet," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 1341–1347.

[9] A. Franchi, C. Secchi, H. I. Son, H. H. Bülthoff, and P. R. Giordano, "Bilateral teleoperation of groups of mobile robots with time-varying topology," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1019–1033, 2012.

[10] P. R. Giordano, A. Franchi, C. Secchi, and H. H. Bülthoff, "A passivity-based decentralized strategy for generalized connectivity maintenance," *The International Journal of Robotics Research*, vol. 32, no. 3, pp. 299–323, 2013.

[11] A. Y. Mersha, S. Stramigioli, and R. Carloni, "On bilateral teleoperation of aerial robots," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 258–274, 2013.

[12] L. Sabattini, C. Secchi, and C. Fantuzzi, "Controlling the interaction of a multi-robot system with external entities," in *2018 IEEE International Conference on Robotics and Automation*. IEEE, 2018, pp. 7654–7659.

[13] C. Secchi and F. Ferraguti, "Energy optimization for a robust and flexible interaction control," in *2019 IEEE International Conference on Robotics and Automation*. IEEE, 2019, pp. 1919–1925.

[14] M. Minelli, F. Ferraguti, N. Piccinelli, R. Muradore, and C. Secchi, "An energy-shared two-layer approach for multi-master-multi-slave bilateral teleoperation systems," in *2019 IEEE International Conference on Robotics and Automation*. IEEE, 2019, pp. 423–429.

[15] S. S. Groothuis and S. Stramigioli, "Energy budget transaction protocol for distributed robotic systems," in *2019 IEEE International Conference on Robotics and Automation*. IEEE, 2019, pp. 1563–1568.

[16] M. Kwon, M. Li, A. Bucquet, and D. Sadigh, "Influencing leading and following in human-robot teams," in *Proceedings of Robotics: Science and Systems*, Freiburg im Breisgau, Germany, 2019, doi: 10.15607/RSS.2019.XV.075.

[17] Y. Sung, A. K. Budhiraja, R. K. Williams, and P. Tokekar, "Distributed simultaneous action and target assignment for multi-robot multi-target tracking," in *2018 IEEE International Conference on Robotics and Automation*. IEEE, 2018, pp. 1–9.

[18] S. Ghapani, S. Rahili, and W. Ren, "Distributed average tracking of physical second-order agents with heterogeneous unknown nonlinear dynamics without constraint on input signals," *IEEE Transactions on Automatic Control*, vol. 64, no. 3, pp. 1178–1184, 2019.

[19] J. George and R. Freeman, "Robust dynamic average consensus algorithms," *IEEE Transactions on Automatic Control*, 2019, doi: 10.1109/TAC.2019.2901819.

[20] J. D. Peterson, T. Yucelen, J. Sarangapani, and E. L. Pasiliao, "Active-passive dynamic consensus filters with reduced information exchange and time-varying agent roles," *IEEE Transactions on Control Systems Technology*, 2019, doi: 10.1109/TCST.2019.2896534.

[21] K. I. Diamantaras and S. Y. Kung, *Principal Component Neural Networks: Theory and Applications*. New York, NY, USA: John Wiley & Sons, Inc., 1996.

[22] S. Li, M. Zhou, X. Luo, and Z. You, "Distributed winner-take-all in dynamic networks," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 577–589, 2017.

[23] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[24] H. Lim and C. Kim, "Flooding in wireless ad hoc networks," *Computer Communications*, vol. 24, no. 3, pp. 353 – 363, 2001.

[25] Y. Yang, D. Constantinescu, and Y. Shi. Distributed winner-take-all teleoperation of a multi-robot system. Accessed 2019-09-13. [Online]. Available: http://hdl.handle.net/1828/11148