

Online Noise-Estimation-based Neighbor Selection for Multi-Manipulator Systems^{*}

Henghua Shen^{*}, Ya-Jun Pan^{*} and Georgeta Bauer^{*}

^{*} *Mechanical Engineering Department, Dalhousie University, Halifax, Canada, B3H 4R2, (e-mail: henghua.shen@dal.ca, yajun.pan@dal.ca, georgeta.bauer@dal.ca).*

Abstract: In this paper, a novel online neighbor selection policy is proposed in the control of nonlinear networked multi-manipulator systems where manipulators' joints' signals are subject to varying noise levels. By addressing the issue in many conventional control methods of multi-agent systems (MASs) where all available neighbor signals are used without evaluating the quality of the information, efforts of this paper seek to improve the overall tracking performance by actively selecting neighbor feedback signals in the robust non-singular terminal sliding mode (NTSM) control. A fast neighbor selection scheme is presented by incorporating an online noise covariance estimation into a nonlinear continuous-discrete unscented Kalman filter (CD-UKF). A selection index vector is recursively updated by the estimated noise covariance matrix for the control design. Simulation results of a group of six degrees of freedom (with three actuated joints) Phantom Omni models demonstrate the effectiveness of the online neighbor selection approach and compare it to previous work which does not actively select neighbor candidates.

Keywords: Neighbor Selection, Online Noise Estimation, Unscented Kalman Filter (UKF), Non-singular Terminal Sliding-Mode (NTSM) Control, Manipulator, Nonlinear Systems.

1. INTRODUCTION

In the majority of the previous approaches, all the transmitted data was immediately used in the control design of MASs. However, in many practical MASs, agents and sensors are heterogeneous and/or experience different degrees of wear and tear, and therefore suffer from different levels of noise and inaccuracy. Unnecessarily using noisy propagated signals in the control design can severely degrade an otherwise well-performing agent's performance. Therefore, it is necessary to set up a neighbor selection strategy to reduce the influence of neighbors with noisy signals. This idea has been applied to the leader selection problem in a leader-following structure (Franchi et al. (2011)). However, actively selecting the neighbor information has been given less attention in networked MASs for the purpose of improving performance.

Dynamic neighbor selection involves the changing network connectivity that also has been considered in the matter of switching topology in a network of MASs. Currently, a great deal of research is being devoted to the topic of switching topology in response to the dynamic external constraints such as the limited field of view or unexpected obstacle occlusions. Such topology switching strategies are referred to as *Passive Switching Topology* (PST). Many control methods aim at demonstrating the control stability under all possible topologies or different connections using PST (Olfati-Saber and Murray (2004); Xiao and Wang

(2008)). Xie and Wang (2005) proposed *Active/Strategic Switching Topology* (AST) by intentionally selecting two neighbors' states in the control design of each agent. Other research on AST can also be found in (Xie and Wang (2005); Yang et al. (2014); Hoang et al. (2018)), which help decrease the communication complexity/cost drastically, but they are more applicable for MASs with a large number of agents (e.g. swarm system (Krishnanand and Ghose (2009))). Control approaches with AST for networked MASs with small number of agents, such as multi-manipulator systems, have not been thoroughly studied. Nevertheless, intentionally selecting suitable neighbor/s in the control design for small-volume systems is of great importance and motivation for the design of the neighbor selection policy in this paper.

The literature that is the most relevant to the proposed paper includes: Sarkka (2007); Hoang et al. (2018); Mao and Zhang (2018); Khoo et al. (2009); Shen et al. (2017) and Kontoroupi and Smyth (2015). Compared to these works, the main contributions of this paper include:

(1) Compared to Sarkka (2007) where a CD-UKF is introduced with the noise mean and covariance given as a priori, this paper considers an online noise covariance estimation which is incorporated into a nonlinear CD-UKF to recursively approximate the true noise covariance. Furthermore, compared to Kontoroupi and Smyth (2015), the noise estimation method is extended to also be used as a neighbor selection index for a fast noise-level-based neighbor selection scheme in the control design.

(2) Compared to Hoang et al. (2018) and Mao and Zhang (2018) where the active neighbor selection aims at reduc-

^{*} This work was supported by NSERC (Natural Sciences and Engineering Research Council of Canada), Canada, NSRIGS (Nova Scotia Research and Innovation Graduate Scholarship), Canada, and CSC (China Scholarship Council), China.

ing the communication complexity and computation load for MASs with a large number of agents, the active neighbor selection approach in this paper focuses on reducing the noise propagation to improve the performance of the nonlinear small-volume MASs.

(3) Compared to Khoo et al. (2009) and Shen et al. (2017) where joint accelerations are required in the NTSM method for nonlinear MASs, this paper considers the state estimation with a nonlinear CD-UKF. As a result, the acceleration feedback signals are not required to be measurable. In addition, an active neighbor selection approach is integrated into the NTSM control design for nonlinear systems that are subject to unknown measurement and process noises.

2. PROBLEM STATEMENT

2.1 Graph Theory

In the leader-following MASs, a graph $G = (\nu, \varepsilon, A)$ is used to describe the connection and information flow among the followers, where ν represents the nodes, ε denotes the edges, and the adjacency matrix $A = [a_{ij}]$ describes the connections between agents, which is defined such that $a_{ij} = 1$ if $(i, j) \in \varepsilon$, and $a_{ij} = 0$, if $(i, j) \notin \varepsilon$. The connection between the i^{th} follower and the leader is represented by $b_i = 1$, if the i^{th} follower is connected to the leader, and $b_i = 0$ otherwise. The following assumptions specify the communication network.

Assumption 1. Assumed that the manipulators are communicating via Ethernet Local Area Network (LAN) such that the transmission delay is very small (usually several milliseconds or less) and neglected.

Assumption 2. During the operation, the follower graph is connected and at least one follower agent can receive information from the leader.

2.2 System Description

An m -DOF manipulator's dynamics can be represented by the Euler-Lagrange equations in joint space as follows.

$$M_i(\mathbf{q}_i)\ddot{\mathbf{q}}_i + C_i(\mathbf{q}_i, \dot{\mathbf{q}}_i)\dot{\mathbf{q}}_i + \mathbf{g}_i(\mathbf{q}_i) + \mathbf{f}_i = \boldsymbol{\tau}_i - \boldsymbol{\tau}_i^e, \quad (1)$$

where $\mathbf{q}_i \in \mathbb{R}^m$, $\dot{\mathbf{q}}_i \in \mathbb{R}^m$, and $\ddot{\mathbf{q}}_i \in \mathbb{R}^m$ are the angular position, velocity and acceleration for the i^{th} manipulator. Definitions of the following matrices are given as: $M_i \in \mathbb{R}^{m \times m}$, is the symmetric and uniformly positive definite inertia matrix. $C_i \in \mathbb{R}^{m \times m}$ is the Coriolis and centrifugal loading matrix. $\mathbf{g}_i \in \mathbb{R}^m$ is the gravitational loading vector. $\boldsymbol{\tau}_i \in \mathbb{R}^m$ denotes the designed control input and $\boldsymbol{\tau}_i^e \in \mathbb{R}^m$ denotes the environment force exerted on the manipulator. Since only free motion tracking is considered in this paper, $\boldsymbol{\tau}_i^e = \mathbf{0}$. $i = 1, 2, \dots, n$ ($n \geq 2$).

In many engineering and physics applications, a continuous-time signal from a nonlinear system is observed discretely due to processing delays or the manner of operation. Therefore, it is more suitable to use the nonlinear CD-UKF (Sarkka (2007)). By redefining the states $\mathbf{X}_i(t) \in \mathbb{R}^L$ and $\dot{\mathbf{X}}_i(t) \in \mathbb{R}^L$ ($L = 2m$) as

$$\mathbf{X}_i(t) = \begin{bmatrix} \mathbf{q}_i(t) \\ \dot{\mathbf{q}}_i(t) \end{bmatrix}, \quad \dot{\mathbf{X}}_i(t) = \begin{bmatrix} \dot{\mathbf{q}}_i(t) \\ \ddot{\mathbf{q}}_i(t) \end{bmatrix}, \quad (2)$$

the nonlinear state-transition function is given in a continuous-time form as

$$\begin{aligned} \dot{\mathbf{X}}_i(t_{k+1}) &= F_m(\mathbf{X}(t_k), \boldsymbol{\tau}_i(t_k), t_k) + \mathbf{v}_i(t_{k+1}) \\ &= \left[\int_{t_k}^{t_{k+1}} F_v(\mathbf{X}(\tau), \boldsymbol{\tau}_i(\tau), \tau) d\tau \right. \\ &\quad \left. F_v(\mathbf{X}(t_k), \boldsymbol{\tau}_i(t_k), t_k) \right] + \mathbf{v}_i(t_{k+1}), \quad (3) \\ \ddot{\mathbf{X}}_i(t_{k+1}) &= F_v(\mathbf{X}_i(t_k), \boldsymbol{\tau}_i(t_k), t_k) \\ &= M_i^{-1}(t_k)(\boldsymbol{\tau}_i(t_k) - C_i(t_k)\dot{\mathbf{q}}_i(t_k) - \mathbf{g}_i(t_k)). \end{aligned}$$

The measurement is modeled in a discrete-time manner as

$$\mathbf{Y}_{i,k+1} = H_v(\mathbf{X}_i(t_{k+1}), \mathbf{r}_{i,k+1}, t_{k+1}) = \mathbf{X}_i(t_{k+1}) + \mathbf{r}_{i,k+1}, \quad (4)$$

where $\mathbf{Y}_{i,k+1} \in \mathbb{R}^L$, F_m and H_m are process model and measurement model, respectively, and their noises are represented by $\mathbf{v}_i(t_{k+1}) \sim \mathcal{N}(0, V_i(t_{k+1}))$ and $\mathbf{r}_{i,k+1} \sim \mathcal{N}(0, R_{i,k+1})$. $\mathbf{v}_i(t_{k+1}) \in \mathbb{R}^L$ and $\mathbf{r}_{i,k+1} \in \mathbb{R}^L$. $V_i(t_{k+1})$ and $R_{i,k+1}$ are noise covariance matrices.

Since we are concerned with the design of the active neighbor selection approach, for simplicity the following assumptions are made.

Assumption 3. In the calculation of M_i , C_i , and \mathbf{g}_i , the system parameters, such as link length, joint mass, etc, are predetermined, although they can be estimated in real time by the CD-UKF introduced in this paper.

Assumption 4. The noises in the state transition and observation equations are additive zero-mean Gaussian processes. In addition, the true noise covariance matrices are heterogeneous between manipulators due to the different extents of wear and tear.

Assumption 5. The joint angular position \mathbf{q} and velocity $\dot{\mathbf{q}}$ can be measured, while the acceleration $\ddot{\mathbf{q}}$ is not measurable. Instead, the acceleration estimation from the CD-UKF is used when it is required in the controllers.

Control Objectives: For nonlinear multi-manipulator systems formulated by (3) and (4), that are subject to varying unknown noise levels, the objective is to design a NTSM controller incorporating a novel online neighbor selection policy and the CD-UKF-based state estimation to improve the overall tracking performance in the sense of $\|\mathbf{q}_i(t) - \mathbf{q}_0(t)\|_1 \leq \Delta_\varepsilon$, where the bound $\Delta_\varepsilon \geq 0$ is desired to be as small as possible while the tracking stability is well maintained.

3. CONTROL STRATEGIES

The unknown noise covariance estimation requires a recursive implementation to reach the true values from any given initialization. This is driven by the nonlinear CD-UKF that is introduced for the state estimation. Taking the i^{th} agent as an example, procedures of the CD-UKF are presented as follows.

3.1 Continuous-Discrete Unscented Kalman Filter

The UKF specifies the state distribution using a minimal set of carefully chosen sample points (called sigma points) to completely capture the true mean and covariance of the

state distribution for any nonlinearity. To facilitate the presentation of the CD-UKF, some weights are defined first as follows.

$$\begin{aligned} w_0^m &= \frac{\lambda}{L + \lambda}, \\ w_0^c &= \frac{\lambda}{L + \lambda} + (1 - \mu^2 + \varsigma), \\ w_p^m &= w_p^c = \frac{1}{2(L + \lambda)}, \quad p = 1, 2, \dots, 2L, \end{aligned}$$

where $\lambda = \mu^2(L + \varkappa) - L$. \varkappa as well as μ , ς are scalars to be designed (Julier et al. (1995)).

Rewriting the weights in matrix form as

$$W^m = [w_0^m \ w_1^m \ \dots \ w_{2L}^m]^T, \quad (5)$$

$$W^c = [w_0^c \ w_1^c \ \dots \ w_{2L}^c]^T, \quad (6)$$

$$\begin{aligned} W^{mc} &= (I - [W^m \ W^m \ \dots \ W^m]) \times \text{diag}(W^c) \\ &\quad \times (I - [W^m \ W^m \ \dots \ W^m])^T, \end{aligned} \quad (7)$$

where $I \in \mathbb{R}^{(2L+1) \times (2L+1)}$ is an identity matrix. $W^{mc} \in \mathbb{R}^{(2L+1) \times (2L+1)}$, $W^m \in \mathbb{R}^{2L+1}$ and $W^c \in \mathbb{R}^{2L+1}$.

Then, the CD-UKF that includes a prediction step and an update step can be formulated as follows (Sarkka (2007)).

Prediction step Prediction step solves the predicted probability density at time step t_{k+1} using the posterior probability density at the last time step (i.e., t_k) as the boundary condition. Assume that $\mathbf{X}_i(t_k)$ has a mean $\bar{\mathbf{X}}_i(t_k)$ and a covariance $P_i(t_k)$. The sigma points $\chi_i(t_k) \in \mathbb{R}^{L \times (2L+1)}$ are calculated as

$$\begin{aligned} \chi_i(t_k) &= [\bar{\mathbf{X}}_i(t_k) \ \bar{\mathbf{X}}_i(t_k) \ \dots \ \bar{\mathbf{X}}_i(t_k)] \\ &\quad + c [0_{L \times L} \ P_{c_i}(t_k) \ -P_{c_i}(t_k)], \end{aligned}$$

where $L = 2m$, $c = \sqrt{L + \lambda}$. $P_{c_i} = \text{chol}(P_i(t_k))$ and $\text{chol}(\cdot)$ denotes the Cholesky factorization.

Since the state-transition is given as a continuous-time function, the prediction of states' mean and covariance is formulated as follows.

$$\dot{\bar{\mathbf{X}}}_i^-(t_k) = \frac{d\chi_i(t_k)}{dt} = F_m(\chi_i(t_k), \mathbf{v}_i(t_k), \boldsymbol{\tau}_i(t_k))W^m, \quad (8)$$

$$\begin{aligned} \dot{P}x_i^-(t_k) &= \frac{dPx_i(t_k)}{dt} \\ &= \chi_i(t_k)W^{mc}F_m^T(\chi_i(t_k), \boldsymbol{\tau}_i(t_k)) \\ &\quad + F_m(\chi_i(t_k), \boldsymbol{\tau}_i(t_k))W^{mc}\chi_i^T(t_k) + V_i(t_k). \end{aligned} \quad (9)$$

Integrating the differential equations in (8) and (9) over $[t_k, t_{k+1}]$ gives the predicted mean and covariance as

$$\begin{aligned} \bar{\mathbf{X}}_i^-(t_{k+1}) &= \int_{t_k}^{t_{k+1}} \dot{\bar{\mathbf{X}}}_i^-(\tau) d\tau, \\ Px_i^-(t_{k+1}) &= \int_{t_k}^{t_{k+1}} \dot{P}x_i^-(\tau) d\tau. \end{aligned}$$

Update step Since the measurement is carried out in a discrete-time fashion, the posterior probability density is computed in discrete-time as

$$\begin{aligned} \chi_i^-(t_{k+1}) &= [\bar{\mathbf{X}}_i^-(t_{k+1}) \ \bar{\mathbf{X}}_i^-(t_{k+1}) \ \dots \ \bar{\mathbf{X}}_i^-(t_{k+1})] \\ &\quad + c [0_{L \times L} \ P_{c_i}^-(t_{k+1}) \ -P_{c_i}^-(t_{k+1})], \end{aligned}$$

$$\mathbb{Y}_{i,k+1}^- = H_m(\chi_i^-(t_{k+1}), t_{k+1}),$$

$$\hat{\mathbf{Y}}_{i,k+1}^- = \mathbb{Y}_{i,k+1}^- W^m,$$

where $P_{c_i}^-(t_{k+1}) = \text{chol}(Px_i^-(t_{k+1}))$.

The covariance of the measurement is given by

$$Py_{i,k+1} = \mathbb{Y}_{i,k+1}^- W^{mc} (\mathbb{Y}_{i,k+1}^-)^T + R_{i,k+1}. \quad (10)$$

The cross-covariance of the state and measurement are

$$Pxy_{i,k+1} = \chi_i^-(t_{k+1}) W^{mc} (\mathbb{Y}_{i,k+1}^-)^T.$$

Eventually the estimated state mean and covariance are

$$K_{k+1} = Pxy_{i,k+1} Py_{i,k+1}^{-1},$$

$$\mathbf{X}_i^+(t_{k+1}) = \mathbf{X}_i^-(t_{k+1}) + K_{k+1}(\mathbf{Y}_{i,k+1} - \hat{\mathbf{Y}}_{i,k+1}^-),$$

$$Px_i^+(t_{k+1}) = Px_i^-(t_{k+1}) - K_{k+1}Py_{i,k+1}K_{k+1}^T,$$

where K_{k+1} denotes the filter gain.

3.2 Neighbor Selection Policy

The CD-UKF provides a recursive estimation of the unknown state (and structural parameters when necessary). However, as shown in (9) and (10), the noise covariances are required as a priori, but very often that is not the case. The online covariance estimation method used to relax the requirement of noise information is presented as follows.

Given the prior parameters, shape $\triangleq \epsilon_{i,t_k} \in \mathbb{R}^1$ and rate $\triangleq \Phi_{i,t_k} \in \mathbb{R}^{L \times L}$, of an inverse Wishart distribution (IWD), their posteriors are calculated by

$$\epsilon_{i,t_{k+1}} = \epsilon_{i,t_k} + 1, \quad (11)$$

$$\Phi_{i,t_{k+1}} = \Phi_{i,t_k} + \Delta \mathbf{S}_i(t_{k+1}) \Delta \mathbf{S}_i^T(t_{k+1}), \quad (12)$$

where $\Delta \mathbf{S}_i(t_{k+1})$ represents the Gaussian noise sample that is given by

$$\Delta \mathbf{S}_i(t_{k+1}) \triangleq \mathbf{X}_i^+(t_{k+1}) - F_m(\mathbf{X}_i^+(t_k), \mathbf{v}_i(t_k), \boldsymbol{\tau}_i(t_k)) \quad (13)$$

for the estimation of V_i in (9) or

$$\Delta \mathbf{S}_i(t_{k+1}) \triangleq \mathbf{Y}_{i,k+1} - \hat{\mathbf{Y}}_{i,k+1}^- \quad (14)$$

for the estimation of R_i in (10).

Then, the noise level (or covariance matrix) is approximated by a point estimation (i.e., the mode of the IWD):

$$\boldsymbol{\omega}_{i,t_{k+1}} = \frac{\text{Diag}(\Phi_{i,t_{k+1}})}{\epsilon_{i,t_{k+1}} + 1 + p}, \quad (15)$$

where p is a positive constant. $\text{Diag}(\ast)$ returns a vector that contains all diagonal entries of an input square matrix. $\boldsymbol{\omega}_{i,t_{k+1}}$ represents the updated estimations of V_i and R_i , denoted by $\boldsymbol{\omega}_{i,t_{k+1}} \triangleq \{\hat{V}_i, \hat{R}_i\}$. Thus, V_i and R_i in (9) and (10) are replaced by \hat{V}_i and \hat{R}_i , respectively so that the above iteration can run along with the CD-UKF.

The noise level $\omega_{i,t_{k+1}}$ provides an index for the dynamic neighbor identification. The online neighbor selection policy is designed based primarily on the noise level of neighbor manipulators. By setting up a noise level threshold ξ , the transmitted neighbor signals stamped with estimated noise level are used to determine if a neighbor agent is “performing-well” or “performing-poorly”. As a result, a dynamic selection index Ω is designed as

$$\Omega_{i,t_{p+1}} = \frac{\text{diag}(\mathbf{1} - \omega_{i,t_{p+1}})(\text{sgn}(\xi - \omega_{i,t_{p+1}}) + \mathbf{1})}{2}, \quad (16)$$

$$\Omega_{i,t_{p+1}} = \text{diag}(\Omega_{i,t_{p+1}}), \quad (17)$$

where $\text{diag}(\ast)$ returns a diagonal matrix of an input vector. $\mathbf{1}$ denotes a vector of all ones. Let $T_{ns} = t_{p+1} - t_p$ denote the update sampling time of the subsequent neighbor selection. To avoid updating the selection law unnecessarily too frequently, T_{ns} is chosen to be not less than sampling time of the sensor measurements and assume that t_p is an integral multiple of t_k .

Note that the noise level threshold ξ is chosen at the designer’s discretion and the individual robots’ application-specific performance requirements at this stage. In addition, the dimension of Ω may vary due to the choice of state. For simplicity, we only use the position noise sample in the calculation of Ω such that $\Omega \in \mathbb{R}^{m \times m}$.

For compactness, the augment of all time-dependent signals is omitted and then the neighbor selection policy can be given by

$$\mathbf{e}_i = \sum_{j=1}^n a_{ij} \Omega_j (\mathbf{q}_i - \mathbf{q}_j) + b_i (\mathbf{q}_i - \mathbf{q}_0), \quad (18)$$

$$\dot{\mathbf{e}}_i = \sum_{j=1}^n a_{ij} \Omega_j (\dot{\mathbf{q}}_i - \dot{\mathbf{q}}_j) + b_i (\dot{\mathbf{q}}_i - \dot{\mathbf{q}}_0), \quad (19)$$

where $\mathbf{e}_i \in \mathbb{R}^m$ and $\dot{\mathbf{e}}_i \in \mathbb{R}^m$ are the position error and velocity error between the i^{th} agent and the leader (denoted by the subscript of 0) as well as the selected neighbors. Note that as the continuous controller is designed, the zero-order hold (ZOH) model is required to convert the discrete-time measurements to continuous-time signals.

3.3 Non-Singular Terminal Sliding Mode Control Design

With the neighbor selection scheme in (16)-(19), the sliding surface for the NTSM control design is written as

$$\mathbf{s}_i = \mathbf{e}_i + \beta_i (\dot{\mathbf{e}}_i)^{\alpha_i}, \quad (20)$$

where $\beta_i > 0$, $\alpha_i = p/q$, and p and q are selected as odd numbers such that $1 < \alpha_i < 2$.

The controller of the i^{th} follower is designed as follows

$$\begin{aligned} \boldsymbol{\tau}_i = & C_i \dot{\mathbf{q}}_i + \mathbf{g}_i + \left(\sum_{j=1}^n a_{ij} \Omega_j + b_i I \right)^{-1} M_i \left\{ \frac{-(\dot{\mathbf{e}}_i)^{(2-\alpha_i)}}{\alpha_i \beta_i} \right. \\ & \left. + \sum_{j=1}^n a_{ij} \Omega_j \hat{\mathbf{q}}_j + b_i \ddot{\mathbf{q}}_0 - (\mathbb{B}_i^{ae} + \kappa_i) \text{sgn}(\mathbf{s}_i) \right\}, \quad (21) \end{aligned}$$

where $\kappa_i > 0$. $\hat{\mathbf{q}}_j$ is the estimated acceleration from the CD-UKF. \mathbb{B}_i^{ae} represents the upper bound of the

acceleration estimating error, that is, $\|\boldsymbol{\varepsilon}_i^{ae}\|_1 \leq \mathbb{B}_i^{ae}$ and $\boldsymbol{\varepsilon}_i^{ae} = \sum_{j=1}^n a_{ij} \Omega_j (\hat{\mathbf{q}}_j - \ddot{\mathbf{q}}_j)$ with $(\hat{\mathbf{q}}_j - \ddot{\mathbf{q}}_j)$ approximated by diagonal of the predicted acceleration covariance in (9).

The stability and joint angular position convergence proof using the control input in (21) are presented as follows.

Proof. Substitute the control input in (21) into the dynamics in (1), we have

$$\begin{aligned} & M_i \ddot{\mathbf{q}}_i + C_i \dot{\mathbf{q}}_i + \mathbf{g}_i \\ = & C_i \dot{\mathbf{q}}_i + \mathbf{g}_i + \left(\sum_{j=1}^n a_{ij} \Omega_j + b_i I \right)^{-1} M_i \left\{ -\frac{\dot{\mathbf{e}}_i^{2-\alpha_i}}{\alpha_i \beta_i} \right. \\ & \left. + \sum_{j=1}^n a_{ij} \Omega_j \hat{\mathbf{q}}_j + b_i \ddot{\mathbf{q}}_0 - \mathbb{B}_i^{ae} \text{sgn}(\mathbf{s}_i) - \kappa_i \text{sgn}(\mathbf{s}_i) \right\}. \end{aligned}$$

Cancelling out the terms $C_i \dot{\mathbf{q}}_i$ and \mathbf{g}_i yields

$$\begin{aligned} & \left(\sum_{j=1}^n a_{ij} \Omega_j + b_i I \right) \ddot{\mathbf{q}}_i - \sum_{j=1}^n a_{ij} \Omega_j \hat{\mathbf{q}}_j - b_i \ddot{\mathbf{q}}_0 \\ = & -\frac{\dot{\mathbf{e}}_i^{2-\alpha_i}}{\alpha_i \beta_i} - \mathbb{B}_i^{ae} \text{sgn}(\mathbf{s}_i) - \kappa_i \text{sgn}(\mathbf{s}_i). \quad (22) \end{aligned}$$

The left-hand side of (22) satisfies the following equation

$$\left(\sum_{j=1}^n a_{ij} \Omega_j + b_i I \right) \ddot{\mathbf{q}}_i - \sum_{j=1}^n a_{ij} \Omega_j \hat{\mathbf{q}}_j - b_i \ddot{\mathbf{q}}_0(t) = \ddot{\mathbf{e}}_i + \boldsymbol{\varepsilon}_i^{ae}. \quad (23)$$

From (22) and (23), we have

$$\ddot{\mathbf{e}}_i + \boldsymbol{\varepsilon}_i^{ae} = -\frac{\dot{\mathbf{e}}_i^{2-\alpha_i}}{\alpha_i \beta_i} - \mathbb{B}_i^{ae} \text{sgn}(\mathbf{s}_i) - \kappa_i \text{sgn}(\mathbf{s}_i),$$

which can be further expressed as

$$\begin{aligned} \dot{\mathbf{e}}_i + \alpha_i \beta_i \text{diag}[\dot{\mathbf{e}}_i^{\alpha_i-1}] \ddot{\mathbf{e}}_i \\ = -\alpha_i \beta_i \text{diag}[\dot{\mathbf{e}}_i^{\alpha_i-1}] (\mathbb{B}_i^{ae} + \boldsymbol{\varepsilon}_i^{ae} + \kappa_i) \text{sgn}(\mathbf{s}_i). \end{aligned}$$

The time derivative of the sliding surface in (20) is

$$\dot{\mathbf{s}}_i = -\alpha_i \beta_i \text{diag}[\dot{\mathbf{e}}_i^{\alpha_i-1}] (\mathbb{B}_i^{ae} + \boldsymbol{\varepsilon}_i^{ae} + \kappa_i) \text{sgn}(\mathbf{s}_i).$$

Constructing the Lyapunov function in a quadratic form as $V_i = \frac{1}{2} \mathbf{s}_i^T \mathbf{s}_i$, and its derivative is

$$\dot{V}_i = \mathbf{s}_i^T \dot{\mathbf{s}}_i = -\alpha_i \beta_i \text{diag}[\dot{\mathbf{e}}_i^{\alpha_i-1}] (\mathbb{B}_i^{ae} + \boldsymbol{\varepsilon}_i^{ae} + \kappa_i) \|\mathbf{s}_i\| \leq 0.$$

Therefore, the proposed control system is stable and the sliding surface can be reached, i.e., $\mathbf{s}_i = \mathbf{0}$. Furthermore, when the sliding surface is reached, we have

$$\dot{\mathbf{e}}_i = -\beta_i^{-\frac{1}{\alpha_i}} \mathbf{e}_i^{-\frac{1}{\alpha_i}}.$$

To prove the motion tracking can be reached, we construct another Lyapunov function as $V_i^e = \frac{1}{2} \mathbf{e}_i^T \mathbf{e}_i$, and its derivative becomes

$$\dot{V}_i^e = \mathbf{e}_i^T \dot{\mathbf{e}}_i = -\beta_i^{-\frac{1}{\alpha_i}} \|\mathbf{e}_i\|_1^{1-\frac{1}{\alpha_i}}.$$

As $\beta_i > 0$ we have $\dot{V}_i^e \leq 0$, which proves that the tracking error can ultimately converge to zero.

4. SIMULATION RESULTS

The numerical simulations were performed on four follower manipulators ($n = 4$) represented by the mathematical model of 6-DoF Phantom Omni haptic devices¹ (with 3 actuated joints, i.e., $m = 3$) with the following inertia, Coriolis, and gravity matrices/vector.

$$M = \begin{bmatrix} h_{11} & 0 & 0 \\ 0 & h_{22} & h_{23} \\ 0 & h_{32} & h_{33} \end{bmatrix}, \mathbf{g} = \begin{bmatrix} 0 \\ \theta_5 g c_2 + \theta_6 g c_{23} \\ \theta_6 g c_{23} \end{bmatrix},$$

$$C = \begin{bmatrix} -(a_1 \dot{q}_2 + a_2 \dot{q}_3) - a_1 \dot{q}_1 & -a_2 \dot{q}_1 & 0 \\ a_1 \dot{q}_1 & -a_3 \dot{q}_3 & -a_3 (\dot{q}_2 + \dot{q}_3) \\ a_2 \dot{q}_1 & a_3 \dot{q}_2 & 0 \end{bmatrix},$$

where $g = 9.8 \text{ N/kg}$, $h_{11} = \theta_1 + \theta_2 c_2^2 + \theta_3 c_{23}^2 + 2\theta_4 c_2 c_{23}$, $h_{22} = \theta_2 + \theta_3 + 2\theta_4 c_3$, $h_{23} = \theta_3 + \theta_4 c_3$, $h_{33} = \theta_3$, $a_1 = \theta_2 c_2 s_2 + \theta_3 c_{23} s_{23} + \theta_4 c_{2*23}$, $a_2 = \theta_3 c_{23} s_{23} + \theta_4 c_2 s_{23}$, $a_3 = \theta_4 s_3$. $s_2 = \sin(q_2)$, $c_2 = \cos(q_2)$, $s_3 = \sin(q_3)$, $c_3 = \cos(q_3)$, $s_{23} = \sin(q_2 + q_3)$, $c_{23} = \cos(q_2 + q_3)$, and $c_{2*23} = \cos(2q_2 + q_3)$. The relevant parameters to calculate M , C , and \mathbf{g} are given in Table 1.

Table 1. Dynamic parameters of the Phantom Omni device (Shen and Pan (2019))

Parameter	Value	Parameter	Value
$l_1(m)$	0	$\theta_3(kg \cdot m^2)$	8.0×10^{-3}
$l_2(m)$	0.135	$\theta_4(kg \cdot m^2)$	0.4×10^{-3}
$l_3(m)$	0.130	$\theta_5(kg \cdot m)$	9.1×10^{-3}
$\theta_1(kg \cdot m^2)$	3.7×10^{-3}	$\theta_6(kg \cdot m)$	5.2×10^{-3}
$\theta_2(kg \cdot m^2)$	7.0×10^{-3}		

The initial joint positions of manipulators were selected within the physical limit (unit: *rad*) as $q_1 \in [-\frac{\pi}{3}, \frac{\pi}{3}]$, $q_2 \in [0, 1.79]$ and $q_3 \in [-2.45, -0.25]$. The initial joint velocities were zero for all manipulators. The leader trajectory was assigned as $\mathbf{q}_0 = [-\pi/5 \sin(ft) \ 1.0 \ -1.6]^T \text{ rad}$ with $f = \pi/3$. The same control gains were chosen for all agents' controllers as $\alpha_i = 7/5$ and $\beta_i = 1$, $\kappa_i = 8$. $T_{ns} = 0.1 \text{ s}$. $\xi = [0.2 \ 0.2 \ 0.2]^T$. $\varkappa = 0$, $\mu = 1 \times 10^{-3}$, and $\varsigma = 2$. $p = 1$ as in (15). The adjacent matrix A and b were chosen as $A = \mathbf{1}_{4 \times 4} - I_{4 \times 4}$ and $b = [1 \ 0 \ 0 \ 1]$, where $\mathbf{1}_{4 \times 4}$ is a square matrix of all ones.

For the sake of compactness, we will look at the first joint of agent 2 only as an example, unless otherwise specified. In the covariance matrix estimation, since integral computation of the probability density is approximated by a point estimation where the IWD mode in (15) is used, the estimating accuracy improves with the increasing sample data. Fig. 1 presents the density distribution evolution where four sets of shape ϵ_i and scalar parameters Φ_i as of 5, 15, 25, and 35 noise samples (i.e (13) and (14)) are executed over 500 sample data ranging from 0 to 5×10^{-3} . As expected, when the noise sample increases from 5 to 35, the density distribution becomes narrower and the mode is closer to the true value. Therefore, it is feasible to use the point estimation to approximate the noise variance. Fig. 2 illustrates an example of a joint that experiences small noises with $R_2 = \text{diag}([0.01 \ 0.01 \ 0.01])$ during $t = 1 - 4 \text{ s}$ and it increases to $R_2 = \text{diag}([0.1 \ 0.1 \ 0.1])$ from $t = 4 \text{ s}$ onwards. Fig. 2(a) shows that the noise variance can be estimated quickly and accurately. Fig. 2(b) illustrates good

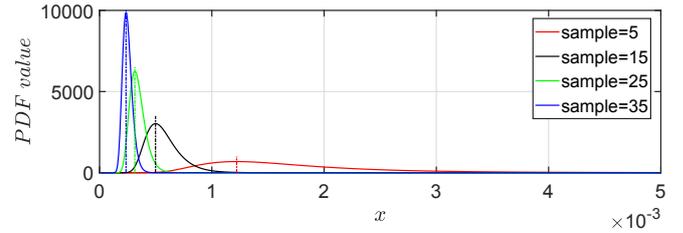


Fig. 1. The evolution of the density distribution and mode over the noise sample of 5, 15, 25, 35.

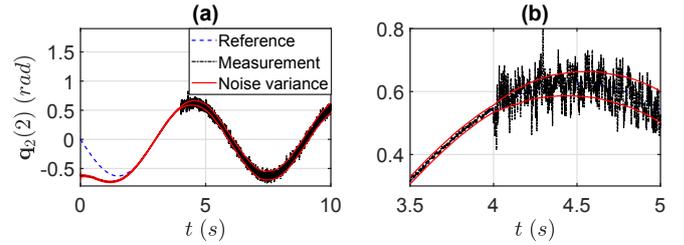


Fig. 2. (a) The measurement noise variance estimation; (b) (Zoomed-in) The estimation adaptability to a suddenly increased noise level at $t = 4 \text{ s}$.

adaptability of the variance estimation approach towards the sudden change of noise levels.

With the estimated noise variance, the noise-level based neighbor selection policy was tested and the results are shown as follows. Joints of the follower manipulators were assumed to experience different levels of noise. The assigned noise covariances for the four manipulators are stacked into $R = [\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{r}_3 \ \mathbf{r}_4]$, where $\mathbf{r}_i = [r_{j1} \ r_{j2} \ r_{j3}]^T$ ($i = 1, 2, 3, 4$) includes the variance for the three joints. Then, the noise covariances (only consider measurement noise) were assigned piecewise as

$t = 0 - 6 \text{ s}$:

$$R = \begin{bmatrix} 0.01 & 0.01 & 0.01 & 0.03 \\ 0.01 & 0.01 & 0.08 & 0.03 \\ 0.01 & 0.01 & 0.08 & 0.03 \end{bmatrix};$$

$t = 6 - 8 \text{ s}$:

$$R = \begin{bmatrix} 0.01 & 0.01 & 1.00 & 0.03 \\ 0.3 & 0.01 & 0.08 & 0.03 \\ 0.01 & 0.01 & 0.08 & 0.03 \end{bmatrix};$$

$t = 8 - 15 \text{ s}$:

$$R = \begin{bmatrix} 0.3 & 0.01 & 0.01 & 0.03 \\ 0.3 & 0.01 & 0.08 & 0.03 \\ 0.01 & 0.01 & 0.08 & 0.03 \end{bmatrix}.$$

According to (16), the selection index ω_i in comparison with the threshold ξ_i determines if a connected agent could be selected as a trustworthy neighbor for the subsequent control design. The selection index variations in the control of joint 1 and joint 2 of agent 2 are illustrated in Fig. 3. For example, the selection indexes for joint 1 of the agent 3 and agent 1 are over the threshold after $t = 6 \text{ s}$ and $t = 8 \text{ s}$ respectively, and thus they are no longer considered as neighbors for the control of other agents' respective joints. A comparison of joint position tracking with and without the neighbor selection is demonstrated in Fig. 4, where the tracking error presents a significant fluctuation without selecting proper neighbor candidates in the control design.

¹ <https://www.3dsystems.com>

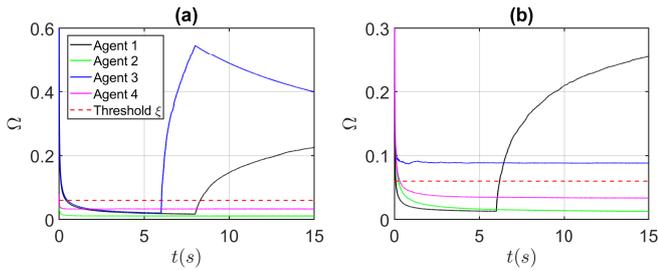


Fig. 3. The change of neighbor selection indexes in comparison with the threshold for (a) joint 1 and (b) joint 2 of agent 2.

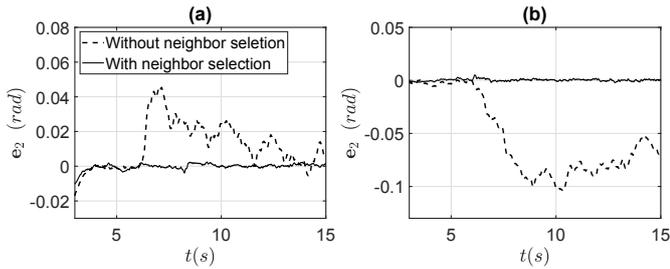


Fig. 4. Comparison of tracking errors of the control design with neighbor selection and without neighbor selection for (a) joint 1 and (b) joint 2 of agent 2.

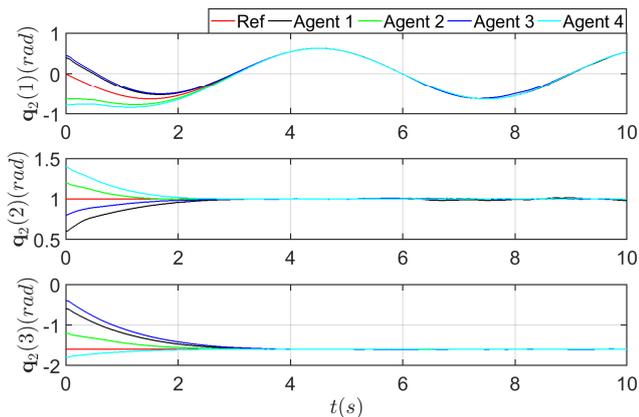


Fig. 5. Joint position tracking between the leader and the follower manipulators.

For example, the tracking error jumps up to 0.045 rad for joint 1 and -0.104 rad for joint 2 as the noise level increases. In contrast, the tracking error experiences less influences from the propagated noises and remain within a small error bound when using the controller with the neighbor selection approach. Fig. 5 presents good tracking performance using the NTSM method.

5. CONCLUSIONS

This paper has presented a novel online neighbor selection policy for networked nonlinear multi-manipulator systems that are subject to different noise levels. The quality of the neighbor information is recursively evaluated online via a nonlinear CD-UKF incorporating a noise covariance estimation, and then the neighbor signals are actively selected for the robust NTSM control. The proposed noise-level-based neighbor selection approach has been successfully

verified by numerical simulations which showed improvement of the overall performance.

REFERENCES

- Franchi, A., Bühlhoff, H.H., and Giordano, P.R. (2011). Distributed online leader selection in the bilateral teleoperation of multiple uavs. In *2011 50th IEEE Conference on Decision and Control and European Control Conference*, 3559–3565. IEEE.
- Hoang, K.D., Fioretto, F., Yeoh, W., Pontelli, E., and Zivan, R. (2018). A large neighboring search schema for multi-agent optimization. In *International Conference on Principles and Practice of Constraint Programming*, 688–706. Springer.
- Julier, S.J., Uhlmann, J.K., and Durrant-Whyte, H.F. (1995). A new approach for filtering nonlinear systems. In *Proceedings of 1995 American Control Conference-ACC'95*, volume 3, 1628–1632. IEEE.
- Khoo, S., Xie, L., and Man, Z. (2009). Robust finite-time consensus tracking algorithm for multirobot systems. *IEEE/ASME transactions on mechatronics*, 14(2), 219–228.
- Kontoroupi, T. and Smyth, A.W. (2015). Online noise identification for joint state and parameter estimation of nonlinear systems. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 2(3), B4015006.
- Krishnanand, K. and Ghose, D. (2009). A glowworm swarm optimization based multi-robot system for signal source localization. In *Design and control of intelligent robotic systems*, 49–68. Springer.
- Mao, Y. and Zhang, Z. (2018). Second-order consensus for multi-agent systems by state-dependent topology switching. In *2018 Annual American Control Conference (ACC)*, 3392–3397. IEEE.
- Olfati-Saber, R. and Murray, R.M. (2004). Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on automatic control*, 49(9), 1520–1533.
- Sarkka, S. (2007). On unscented kalman filtering for state estimation of continuous-time nonlinear systems. *IEEE Transactions on automatic control*, 52(9), 1631–1641.
- Shen, H. and Pan, Y.J. (2019). Improving tracking performance of nonlinear uncertain bilateral teleoperation systems with time-varying delays and disturbances. *IEEE/ASME Transactions on Mechatronics*.
- Shen, H., Pan, Y.J., and He, B. (2017). Teleoperation of multiple cooperative slave manipulators using graph-based non-singular terminal sliding-mode control. In *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 1430–1435. IEEE.
- Xiao, F. and Wang, L. (2008). Asynchronous consensus in continuous-time multi-agent systems with switching topology and time-varying delays. *IEEE Transactions on Automatic Control*, 53(8), 1804–1816.
- Xie, G. and Wang, L. (2005). Consensus control for networks of dynamic agents via active switching topology. In *International Conference on Natural Computation*, 424–433. Springer.
- Yang, W., Wang, X., and Shi, H. (2014). Optimal control nodes selection for consensus in multi-agent systems. *IFAC Proceedings Volumes*, 47(3), 11697–11702.