Coordinated path-following control for networked unmanned surface vehicles

Dong-Liang Chen¹,²,³, Guo-Ping Liu⁴, Ru-Bo Zhang¹,³ and Xingru Qu⁵

Abstract
In this article, the coordinated path-following control problem for networked unmanned surface vehicles is investigated. The communication network brings time delays and packet dropouts to the fleet, which will have negative effects on the control performance of the fleet. To attenuate the negative effects, a novel networked predictive control scheme is proposed. By introducing the predictive error into the control scheme, the proposed control strategy admits some advantages compared with existing networked predictive control strategies, for example, a degree of robustness to disturbances, lower requirements for the computing capacity of the onboard processors, high flexibility in controller design, and so on. Conditions that guarantee the control performance of the overall system are derived in the theoretical analysis. At last, experiments on hovercraft test beds are implemented to verify the effectiveness of the proposed control scheme.

Keywords
Coordinated path-following, networked predictive control, unmanned surface vehicles, time delays

Date received: 10 March 2020; accepted: 7 May 2020

Topic: Mobile Robots and Multi-Robot Systems
Topic Editor: Yangquan Chen
Associate Editor: Xiao Liang

Introduction
Unmanned surface vehicles (USVs) play an important role in a wide range of applications, such as military uses, scientific research, and environmental missions.¹,² As a result, the research for USVs has drawn researchers’ attention, and many control problems have been studied for USVs such as collision avoidance,³,⁴ trajectory tracking,⁵,⁶ target tracking,⁷–⁹ and so on. Path-following control for USVs is a classical research topic in the control community. The aim of path-following control is to force a USV to follow a predefined spacial path.¹,¹⁰ Correspondingly, the aim of coordinated path-following control is to force a fleet of USVs to follow predefined paths meanwhile keep a given formation pattern. The coordinated path-following control problem for multiple USVs can be found in some important

¹ College of Mechanical and Electronic Engineering, Dalian Minzu University, Dalian, China
² Center for Control Theory and Guidance Technology, Harbin Institute of Technology, Harbin, China
³ Key Laboratory of Intelligent Perception and Advanced Control of State Ethnic Affairs Commission, Dalian Minzu University, Dalian, China
⁴ Department of Artificial Intelligence and Automation, Wuhan University, Wuhan, China
⁵ School of Naval Architecture and Ocean Engineering, Dalian Maritime University, Dalian, China

Corresponding authors:
Dong-Liang Chen and Ru-Bo Zhang, College of Mechanical and Electronic Engineering, Dalian Minzu University, Dalian 11660, China.
Emails: truther@hit.edu.cn; zhangrubo@dlnu.edu.cn
applications, for example, oceanographic sampling,\textsuperscript{11} pollution cleanup, and so on.\textsuperscript{1}

There have been abundant research results on both path-following control for a USV and coordinated path-following control for USVs. There are two traditional but effective path-following controller design methods: the line-of-sight method\textsuperscript{12–14} and the virtual-target method.\textsuperscript{15–17} Some excellent results have been achieved based on these two methods, for example, global asymptotic stability is guaranteed for multiple underactuated marine crafts derived from these two methods in the literature.\textsuperscript{15,18} Besides these two methods, backstepping technique is another dominant controller design tool for path-following control of USVs.\textsuperscript{19} In recent years, advanced control theories have been introduced to this field, for example, passivity theory,\textsuperscript{11} neural control theory,\textsuperscript{20} and invariant sets theory.\textsuperscript{21} Despite of the abundant research works, few results have been reported on the coordinated path-following control for USVs with communication constraints such as time delays. In practice, the information communication among vehicles are completed through various mediums such as networks. The introduction of networks will bring time delays and data losses to the system which will degrade the control performance of the overall system. A simple predictive strategy is proposed to mitigate the effects brought by time delays in the literature,\textsuperscript{22} where the formation velocity is used as the predicted velocities of USVs. However, this predictive controller neglects the transient process of formation when the practical velocity is not equal to the formation velocity and will thus cause deterioration of the formation performance. Networked predictive control (NPC) strategy is integrated with coordinated path-following controller to attenuate the negative effects of time delays in the literature.\textsuperscript{23} However, the control scheme can only be implemented on a centralized controller, that is, controllers of all USVs must be executed in a same processor, which degrades the flexibility and robustness of the multi-USV system. In fact, this is determined by the intrinsic weakness of the traditional NPC strategies.

The control strategies to attenuate negative effects brought by communication time delays can be categorized into two classes: the robust control strategies\textsuperscript{24–27} and the NPC strategies.\textsuperscript{28–30} For robust control strategies, the delayed information from other agents is used directly to design the controller such that a performance index can be satisfied for the overall system, while the NPC strategies try to predict the current states of other agents based on the delayed information to compensate for the time delays in transmission. The authors of the literature\textsuperscript{22,23} adopt an NPC strategy to improve the formation control performance of multi-USV systems. The NPC strategies have been proved to be effective to improve the control performance for a single networked system,\textsuperscript{31–33} therefore, many researchers try to extend this class of methods to multi-agent systems.\textsuperscript{34–36} However, the NPC strategies cannot be applied to multi-agent systems directly. First, most existing NPC strategies are designed for linear multi-agent systems, while most controlled plants in practice are nonlinear multi-agent systems; second, the computational efficiency of the NPC strategies is low, especially when applied to complicated plants. This drawback prevents the NPC strategy to be applied in practice, because the computing capacity of most onboard processors is limited; Third, the NPC strategies are based on precise models of the controlled plants, that is, no external disturbance nor model uncertainty can exist, or the predicting precision will be affected. Fourth, the predicting procedure requires further constraints on the communication topology of the system in implementation, this is the main reason why a centralized controller is adopted in the literature.\textsuperscript{23} At last, few experimental results have been reported to test the performance of the algorithms. To overcome these shortages, a modified NPC strategy is proposed in this article. Compared with existing NPC strategies, the proposed NPC strategy in this article admits advantages such as high computing efficiency, a degree of robustness to model uncertainties and external disturbances, and easiness to be applied to nonlinear plants (e.g. USVs in this article), and so on. At last, a control scheme based on the proposed NPC strategy is designed for the coordinated path-following control problem of multiple USVs with communication time delays. To make use of the limited bandwidth of the communication network efficiently, the communication period among USVs is set longer than the local path-following control system.

The remaining part of the article is organized as follows. The coordinated path-following control problem is formulated in the second section, the novel NPC scheme is designed in the third section, and the control performance of the proposed scheme is analyzed in the fourth section. In the fifth section, experiments on hovercraft test beds are designed and implemented to test the control performance of the proposed NPC scheme. At last, conclusion ends the article.

Problem formulation and preliminaries

The kinematics and kinetics of a USV can be expressed as\textsuperscript{19,37,38}

\[
\begin{align}
\dot{x} &= R^B_{0}(\phi)\nu \\
\dot{\nu} &= M_{0}^{-1}D(\nu)\nu + M_{0}^{-1}\tau
\end{align}
\] (1)

where \(x = [x, y, \phi]^T\) is the position vector in the earth-fixed frame, \(\nu = [u, v, \omega]^T\) is the velocity vector in body-fixed frame, \(\tau\) is the control input, \(M_{0} = M_{0}^{T} > 0\) is the inertial matrix, \(D(\nu)\) is the damping matrix, and \(R^B_{0}\) is the rotation matrix from body-fixed frame to earth-fixed frame.

Assume there are \(N\) vehicles in the fleet, a subscript \(i\) is used to index the \(i\)th USV, and the vehicles exchange information via networks. An undirected graph \(G(V, E)\) is
adopted to describe the communication topology, with \( V = \{v_i, i \in [1, 2, \ldots, N]\} \) the set of vertices and \( E = \{e_{ij}, i, j \in V, i \neq j\} \) the set of edges, the cardinality of \( E \) is \( r \). In the graph \( G(V,E) \), each vertex represents a USV, each edge represents an information channel. \( A = \{a_{ij}, i, j \in V\} \) is the adjacency matrix of the graph \( G \), \( N_r \) is the set of neighboring USVs of USV \( i \). \( D = \text{diag}(d_1, d_2, \ldots, d_N) \) is the degree matrix of graph \( G \), where \( d_i \) is the size of \( N_r \). \( L = D - A \) is defined as the Laplacian of the graph \( G \). A path from \( V_i \) to \( V_j \) is a sequence of edges starting from vertex \( V_i \) and ending in \( V_j \). A graph is connected if there exists at least one path for any two vertices \( V_i \) and \( V_j \), \( M = R^{N_r \times r} \) is the incidence matrix of graph \( G \). The method to induce incidence matrix for a bidirected graph can be found in the literature.\(^{39}\) As the incidence matrix is important in the theoretical analysis of this article, some important properties of the incidence matrix are given as follows:

1. The Laplacian \( L \) of a graph \( G \) is
   \[
   L = MM^T \quad (2)
   \]
2. If a graph \( G \) is connected, then \( \text{rank}(M^T) = N - 1 \) and \( \text{ker}(M^T) = 1 \).

Coordinated path-following control involves two objectives: path-following control and coordinated control. Path-following control requires that every USV tends to a prescribed path \( \chi'_i(\theta_i) \), and then moves along the path at given formation velocity \( v_f \). \( \theta_i \) is a scalar that parameterizes the path \( \chi'_i \), therefore, \( \theta_i \) is entitled as the path parameter. Coordinated control requires that the USVs form and maintain a predefined formation pattern. The path parameter \( \theta_i \) is of great importance in coordination control, the coordination among USVs is said to be accomplished when \( \theta_i = \theta_j, i, j \in V \). Other cases where coordination is defined by \( \theta_i \neq \theta_j \) cannot be converted to this form by performing coordinate transformation.\(^{39}\) It can be seen that the coordination can be achieved by adjusting \( \theta_i \) rather than coordinating the state \( \chi_i \) directly. Therefore, only \( \theta_i \) is needed to be exchanged among the USVs through network. The introduction of \( \theta_i \) reduces the required amount of information exchanged through the network thus improves the efficiency of the network. Meanwhile, it simplifies the coordination problem. The control objectives of coordinated path-following control can be concluded as

\[
\begin{align*}
\lim_{t \to \infty} v_f(t) - \chi'_i(\theta_i) &= 0 \quad (3) \\
\lim_{t \to \infty} \dot{\theta}_i(t) - v_f &= 0 \quad (4) \\
\lim_{t \to \infty} (\theta_i(t) - \theta_j(t)) &= 0, \quad \forall i, j \in V, i \neq j 
\end{align*}
\]

The objectives (3) and (4) represent path-following control, while the objective (5) represents coordinated control. As the control task can be decomposed into two parts: coordinated control and individual path-following control, we will discuss these two parts respectively in the following text.

To achieve coordination, we design the path parameter \( \theta_i \) as

\[
\dot{\theta}_i = v_f + v_i(t) \quad (6)
\]

with \( v_i \in \mathbb{R} \) a velocity component added to the formation velocity \( v_f \).

Besides time delays and packet dropouts, the digital nature of networks should be considered. In fact, most networks in practice are digital networks, such as the Internet. This feature makes the controllers update in discrete time instants. As networks are subjected to limited bandwidth, the sampling period cannot be set too small, or the increasing amount of data will cause the waste of bandwidth meanwhile increase the burden of networks. As a result, we design the communication period among USVs greater than the sampling period of the local system. In this case, the dynamics of \( \theta_i \) cannot be expressed as (6) anymore, a corresponding discrete-time model should be derived

\[
\theta_i(k + 1) = \theta_i(k) + T v_f + T v_i(k) \quad (7)
\]

with \( kT \) the time instant when the path parameter \( \theta_i \) is sampled, \( T \) is the sampling period. In the derivation of (7), the Forward Euler discrete approximation is applied.\(^{30}\) By designing \( v_i(k) \) properly, the objectives (4) and (5) can be achieved.

Now we turn to the path-following problem. Assume that there exists a path-following control law

\[
\tau_i = \kappa(\chi_i, \chi'_i, w_i) \quad (8)
\]

such that the closed-loop system takes the following form

\[
\begin{align*}
\dot{z}_i &= F_i(t, z_i, w_i) \\
y_i &= G_i(t, z_i)
\end{align*} \quad (9)
\]

where \( z_i \in \mathbb{R}^{n_i} \) is the state vector that includes the path-following error \( \chi_i - \chi'_i \) and its derivatives, \( w_i \in \mathbb{R}, y_i \in \mathbb{R} \) is the input and output of the USV \( i \), \( F_i : [0, \infty) \times \mathbb{R}^{n_i} \times \mathbb{R} \) is Lipschitz in \((z_i, w_i)\), continuous in \( t \), \( F_i(t, 0, 0) = 0 \), \( G_i : [0, \infty) \times \mathbb{R}^{n_i} \) is Lipschitz in \( z_i \) and continuous in \( t \). As there are many research results on this topic, existing methods can be directly adopted to design the path-following controller for a single USV. The following assumption is made for system (9).

**Assumption 1.** For the system (9), there exists a continuously differentiable function \( V_{2i} : [0, \infty) \times \mathbb{R}^{n_i} \rightarrow \mathbb{R}, \rho_0 > 0 \) and a class-K function \( f \) such that

\[
\begin{align*}
V_{2i}(t, z_i) &\geq 0 \\
\frac{\partial V_{2i}}{\partial t} + \frac{\partial V_{2i}}{\partial z_i} F_i(t, z_i, w_i) &\leq -\rho_0 \|z_i\|^2
\end{align*}
\]

for any given compact sets \( S_{z_i} \) and \( S_{w_i} \) that satisfy \( \|x\| \geq f(||w_i||) > 0 \).
Now each USV with the path-following controller (8) can be concluded as
\[
\begin{align*}
\dot{z}_i &= F_i(t, z_i, w_i) \\
y_i &= G_i(t, z_i) \\
\theta_i(k+1) &= \theta_i(k) + T_v y_i + T_v(k)
\end{align*}
\]

It can be seen that the system (10) is a hybrid system. Due to communication time delays, \(\theta_j(t - t_{ij})\), \(j \in N_i\) is available for USV \(i\) at time \(t\) with \(t_{ij}\) the communication time delay between USV \(i\) and USV \(j\). Meanwhile, the path parameter \(\dot{\theta}_j\) from neighboring USVs may not be available for USV \(i\) at time \(t\) due to packet dropouts. As a result, the control performance will be affected.

The time delays and the consecutive packet dropouts are commonly upper bounded for a properly functioning network, or faults may have occurred to the network and alternative networks have to be used to replace it. Meanwhile, controllers in practice need real-time data from the controlled plant and vice versa, historical data too old are not useful anymore for the controller/actuator, for the state of the controlled plant/controller may have changed drastically. As a result, it is reasonable to make following assumption:

**Assumption 2**

1. The consecutive packet dropouts and the time delays of the network are upper bounded by \(t_{d}\) and \(t_{c}\), respectively. Denote \(c = t_{c} + t_{d}\).
2. A time stamp is packaged with each packet.
3. The time of controllers is synchronized.

If the origin of system (9) is asymptotic stable, then control objective (3) is achieved. So the problem now is to design \(v_i(k)\) and \(w_i\) such that the origin of system (9) is asymptotic stable along with the achievement of objectives (4) and (5) under the communication constraints.

**Control scheme design**

In this section, the proposed NPC strategy is discussed in detail. A graphical diagram of the overall control scheme is given in Figure 1.

From Figure 1, it can be seen that the overall control scheme can be divided into two subsystems: the NPC subsystem and the local subsystem.

The NPC subsystem is designed to compensate for the communication time delays and packet dropouts, meanwhile, it adjusts the reference velocity \(w_i(t)\). It receives the predicted path parameters from its neighboring USVs and sends the reference velocity and its predicted path parameters to the path-following controller and the neighboring USVs, respectively. It comprises four modules: the data buffer 1 and 2, the predictor, and the coordination controller. Due to the randomness of time delays, packet sent earlier may arrive later and vice versa. The data buffer 1 is designed to solve this problem, it reorders the received packets by the enclosed time stamps and outputs the required path parameter. To compensate for packet dropouts, the data buffer 2 is designed. Considering the upper bound of the consecutive packet dropouts, the predicted path parameter of USV \(i\) is always available for its neighboring USVs if the length of data buffer 2 is properly designed. The coordination controller is designed as

\[
v_i(k + c) = k_{i1} \sum_{j \in N_i} a_j (\hat{\theta}_j(k + c|k) - \theta_j(k + c))
\]

where \(\hat{\theta}_j(k + c|k)\) is the predicted path parameter of time \(k + c\) based on the available information of time \(k\) and \(k_{i1}\) is the gain of the coordination controller.
The predictor is the core component of the NPC subsystem, it is designed to predict the future path parameters. The coordination performance relies heavily on the predictor.

If the existing NPC strategies are applied, the predictor will be designed as
\[ \hat{\theta}_i(k + q[k]) = \hat{\theta}_i(k + q - 1[k]) + TV_f + k_{i3}\hat{v}_i(k + q - 1[k]) \]

\[ \hat{v}_i(k + q - 1[k]) = k_{i3}\sum_{j \in N_i} \hat{\theta}_j(k + q - 1[k]) - \hat{\theta}_i(k + q - 1[k]) \]

with \( q \in [1, \ldots, C] \), \( \hat{\theta}_i(k|k) = \theta_i(k) \). It can be seen that the predictor needs \( \hat{\theta}_i(k + q - 1[k] - c) \) to calculate \( \hat{\theta}_i(k + q[k]) \) \( \hat{\theta}_j(k + q - 1[k]) \) is not available due to time delays. As a result, we need to compute \( \hat{\theta}_j(k + q - 1[k] - c) \) first to complete the predicting procedure. But it is impossible to calculate \( \hat{\theta}_j(k + q - 1[k] - c) \) without acquiring the path parameters of USV’s \( j \) neighboring USVs, which imposes further constraint on the communication topology. We can adopt a centralized controller to solve this dilemma, that is, all controllers are executed in a same workstation, but this solution destroys the decentralization of the multi-USV system thus degrades some advantages of multi-agent system, such as robustness and flexibility. From the discussion above, we can see that the predicting algorithm is complicated, which will lead to heavy computational burden of the onboard processor. These drawbacks make the existing NPC strategies hard to implement in practice.

Different from existing NPC strategies, the predictor in this article is designed as
\[ \hat{\theta}_i(k + q|k) = \hat{\theta}_i(k + qTv_f + TV_i(k + q - 1[k]) + k_{i3}\hat{v}_i(l + q|l) \]

\[ \hat{v}_i(l + q|l) = \theta_i(l + q) - \hat{\theta}_i(l + q|l) \]

where \( \hat{\theta}_i(k + s[k]) \) and \( \hat{v}_i(k + s[k]) \) are the state prediction and the virtual control input of the time instant \( k + s \) based on the available information at time \( k \), respectively; \( \hat{v}_i(l + q|l), l \in [k - c, k - q] \) is the predicted error of the path parameter of the time \( l \), \( k_{i3} \) is the gain value of the predictor.

**Remark 1.** Compared with existing NPC strategies, the proposed NPC strategy is different in the following aspects:

1. The design of the virtual control input \( \hat{v}_i(k + q - 1[k]) \) is flexible, for example, \( \hat{v}_i(k + q - 1[k]) = \hat{v}_i(k) \), \( \hat{v}_i(k + q - 1[k]) = 0 \), \( \hat{v}_i(k + q - 1[k]) = k_{i3}\sum_{\ell \in N_i,\ell \neq i} \theta_j(k + c) - \theta_i(k + q - 1[k]) \), and so on. The predicting procedure can be simplified when \( \hat{v}_i(k + q - 1[k]) \) is chosen properly, for example, when \( \hat{v}_i(k + q - 1[k]) = 0 \), the virtual control input is eliminated in the predictor. The design of the predicting feedback term is also flexible, \( l \) can be any number in \( [k - c, k - q] \).

2. In (12), the expression of virtual control input \( \hat{v}_i(k + q - 1[k]) \) is not necessarily same as the actual control input \( v_i(k + q - 1[k]) \). As a result, an difference between the predicted path parameter and the true path parameter is inevitable. To solve this problem, the predicting feedback term \( \hat{v}_i(l + q|l) \) is introduced to the predictor, which makes it a closed-loop system. In contrast, existing NPC strategies are based on the precise mathematical models without predicting feedback error, which makes them open-loop predictors. It is well known that a closed-loop system admits properties such as robustness to external disturbances and model uncertainties, therefore, the proposed predictor admits a degree of robustness compared with existing NPC strategies. To distinguish the proposed NPC strategy with existing NPC strategies, we entitle the proposed NPC strategy closed-loop networked predictive control (CLNPC) strategy.

3. For most existing predicting algorithms, each agent predicts its neighboring agents’ states based on the delayed information it received, therefore, redundant computation is inevitable (every agent’s state is predicted by all its neighboring agents). In contrast, the proposed predictor in this article predicts only its own future states, which improves the computing efficiency exceedingly (every agent’s state is predicted only once).

4. Observing (12), it can be seen that the predicting procedure totally depends on its own state (excluding its neighboring agents’ states). As a result, the predictor can be implemented in decentralized processors, meanwhile, no further restrictions on the communication topology are needed to implement the predictor. And this feature makes the predicting procedure simpler than existing NPC strategies, therefore, onboard processors with high computing capacity are not needed to implement the algorithm.

To guarantee the control performance of the overall system, the convergence of the predicting error must be achieved along with the control objectives (3), (4), and (5), that is
\[ \lim_{l \to \infty} \hat{v}_i(l + q|l) = 0 \]

From the discussion above, it can be seen that the CLNPC scheme admits the following advantages: robustness to model uncertainties and external disturbances, low computational requirements for processors, and relaxed restriction on the topology.

At time \( t \), each USV predicts its future path parameters by (12), then sends the predicted sequence to its neighboring USVs. At time \( t + c \), each USV computes its control input by (11).
In Figure 1, the local system comprises the USV (1) and the path-following controller (8). The path-following controller (8) is designed such that Assumption 1 is fulfilled. The input of the local system is designed as

$$w_i(t) = v_i(k), t \in [kT, kT + T)$$  (15)

**Performance analysis**

In this section, the control performance of the system (10) with the predictor designed as (12) and the coordination controller designed as (11) is analyzed. As the predictor (12) has many variations, only the case where $$v_i(k+q-1|k) = 0$$ with the predicting error designed as $$\hat{e}_i(k-c + q|k-c)$$ is considered, other cases can be analyzed similarly.

Define an error variable of the fleet as $$\zeta(k) = M^T \theta(k),$$ with $$\theta(k) = [\theta_1(k), \theta_2(k), \ldots, \theta_N(k)]^T.$$ It is obvious that $$\zeta(k) = 0$$ is corresponding to the synchronization of the $$\theta(k),$$ that is, $$\theta_i(k) = \theta_j(k), i,j \in V.$$ At time $$k+c,$$ the reference velocity (11) for USV $$j$$ (USV's $$i$$ neighboring USV) can be expressed as

$$v_j(k+c) = -k_1M_jM^T \hat{\theta}(k+c|k) - k_1d_j \hat{e}_j(k+c|k)$$  (16)

where $$M_{ij}$$ is the $$j$$th row of $$M, \hat{\theta}(k+c|k) = [\hat{\theta}_1(k+c|k), \hat{\theta}_2(k+c|k), \ldots, \hat{\theta}_N(k+c|k)]^T.$$

Substituting (16) into (7) yields

$$\theta_j(k+c + 1) = \theta_j(k+c) + TV_j - k_1TM_jM^T \hat{\theta}(k+c|k)$$

Then the corresponding path parameter vector of the USVs can be concluded as

$$\theta(k+c+1) = \theta(k+c) - TK_1D\hat{e}(k+c|k)$$

$$-TK_1MM^T \hat{\theta}(k+c|k) + TV_1N$$  (17)

where $$K_1 = \text{diag}(k_{11}, k_{21}, \ldots, k_{N1}), \hat{e}(k+c|k) = [\hat{e}_1(k+c|k), \hat{e}_2(k+c|k), \ldots, \hat{e}_N(k+c|k)]^T.$$

Denoting $$\tilde{\zeta}(k+c|k) = M^T \hat{\theta}(k+c|k),$$ then equation (17) can be further expressed as

$$\theta(k+c+1) = \theta(k+c) + TV_1N - TK_1D\hat{e}(k+c|k)$$

$$-TK_1M \tilde{\zeta}(k+c|k)$$  (18)

with $$1_N$$ all-one vector of size $$N.$$ Assume the graph $$G$$ is connected, then rank $$M^T = N - 1$$ and $$M^T 1_N = 0.$$ Then multiply both sides of (18) by $$M^T,$$ one gets

$$\tilde{\zeta}(k+c+1) = \tilde{\zeta}(k+c) - TM^T K_1 M \tilde{\zeta}(k+c|k)$$

$$-TM^T K_1 D \hat{e}(k+c|k)$$  (19)

By (12), the predicted parameters of the fleet can be inferred as

$$\hat{\theta}(k+c|k) = \theta(k) + cTV_f 1_N + K_3 \hat{e}(k+c|k)$$  (20)

where $$K_3 = \text{diag}(k_{13}, k_{23}, \ldots, k_{N3}).$$

From (20), one gets

$$\tilde{\zeta}(k+c|k) = \tilde{\zeta}(k) + MTK_3 \hat{e}(k+c|k)$$  (21)

Subtracting (7) from (12) yields the predicting error

$$\hat{e}(k+c|k) = T \sum_{s=0}^{c-1} v(k+s) - K_3 \hat{e}(k+c|k)$$  (22)

where $$v(k+s) = [v_1(k+s), v_2(k+s), \ldots, v_N(k+s)]^T$$ is the collective velocity input of the fleet at time instant $$k+s.$$ Combining (16), (21), and (22), the collective velocity input $$v(k+c)$$ can be induced as

$$v(k+c) = -K_1M \tilde{\zeta}(k) - TK_1D \sum_{l=0}^{c-1} v(k+l)$$

$$-(K_1MM^T K_3 - K_1DK_3) \hat{e}(k+c|k)$$  (23)

Define a compact vector $$X_1(k)$$ as

$$X_1(k) = [r^T(k), \ldots, r^T(k-c), v^T(k-1), \ldots, v^T(k-c), \hat{e}^T(k-1|k-c-1), \ldots, \hat{e}^T(k-c|k-2c)]^T$$

It can be seen that $$\lim_{k \to \infty} X_1(k) = 0$$ results in $$\lim_{k \to \infty} \tilde{\zeta}(k) = 0, \lim_{k \to \infty} v(k) = 0, \lim_{k \to \infty} \hat{e}(k|k-c) = 0.$$ Combining (19), (22), and (23) yields

$$X_1(k+1) = A_1X_1(k)$$

$$Y_1(k) = C_1X_1(k)$$  (24)

where the system matrix is

$$A_1 = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$

$$A_{11} = \begin{bmatrix} I_r & O_{r \times (r-r)} & -TM^T K_1 M \\ I_r & O_{r \times r} \end{bmatrix}$$

$$A_{21} = \begin{bmatrix} O_{(Nc-N)^{r \times r}} & -K_1 M \\ O_{(Nc-N) \times r} \end{bmatrix}$$

$$A_{12} = \begin{bmatrix} -I_c^T \otimes T^2 M^T K_1 D \\ O_{r \times N\epsilon} \end{bmatrix}$$

$$A_{22} = \begin{bmatrix} -I_c^T \otimes TK_1 D \\ I_{N\epsilon-N} \otimes O_{(Nc-N) \times N} \end{bmatrix}$$

$$A_{31} = O_{(Nc-N) \times (r-r)}, A_{32} = \begin{bmatrix} I_c^T \otimes TI_N \\ O_{(Nc-N) \times N} \otimes O_{(r-c) \times N} \end{bmatrix}$$

$$A_{13} = \begin{bmatrix} O_{r \times (Nc-N)} & TM^T K_1 DK_3 - TM^T K_1 M M^T K_3 \end{bmatrix}$$
The overall system can be represented by the diagram in Figure 2. From Figure 2, it can be seen that the overall system admits a cascade structure. The following Lyapunov function is designed for the overall system

\[
V_0(k) = V_1(X_1(k)) + \sum_{i=1}^{N} V_2(t, z_i(k))
\]

Due to (25) and (27), the difference of \(V_0\) is

\[
\nabla V_0(k) = V_0(k + 1) - V_0(k) \\
\leq -X_1^T(k)QX_1(k) - \sum_{i=1}^{N} (T\rho_0 - T^2K_z)[z_i(k)]^2 - (\mu - T^2K_w)Y_1^T(k)Y_1(k) < 0
\]

where \(w(k) = Y_1(k), w(k) = [w_1(k), w_2(k), \ldots, w_N(k)]^T\) are applied. Given \(X_1 \in S_x\) and a set \(S \supseteq S_x \times S_x\), \((X_1(k), z(k))\) are bounded and stays in \(S\) (see proof for theorem 3 in the literature), that is, for any \((X_1(0), z(0)) \in S\), there exists a class-K function \(\alpha_1\) such that

\[
||X_1(k), z(k)|| \leq \alpha_1(||X_1(0), z(0)||)
\]

The output of the coordination subsystem (24) satisfies

\[
||Y_1(k)|| \leq ||C_1|| ||X_1(k)|| =: \alpha_2(||X_1(k)||)
\]

As the subsystem (9) is Input-to-State Stable (Assumption 1), there exist a class-K function \(\alpha_3\) and a class-KL function \(\beta_3\) such that following inequality holds true for \(t \in [kT, (k + 1)T]\)

\[
||z(t)|| \leq \beta_3(z(kT), t - kT) + \alpha_3(||Y_1(kT)||) \\
\leq \beta_3(z(kT), 0) + \alpha_3(\alpha_2(||X_1(k)||))
\]

As \(X_1(t), t \in [kT, (k + 1)T]\) remains constant, there exist class-K functions \(\alpha_4\) and \(\alpha_5\) such that

\[
||X_1(t), z(t)|| \leq \alpha_4(||X_1(kT), z(kT)||) \\
\leq \alpha_4(\alpha_1(||X_1(0), z(0)||))
\]

Therefore, the equilibrium \((X_1, z) = (0, 0)\) is asymptotic stable for the given set \(S\). In fact, it can be verified that (29) and (33) can be satisfied for any given set \(S \supseteq S_x \times S_x\). As a result, the semi-global asymptotic stability of the origin can be guaranteed, which leads to the accomplishment of the control objectives (3), (4), (5), and (14).

The proof is completed.

Simulations and experiments

To validate the effectiveness of the CLNPC scheme, experiments on hovercraft test beds are carried out.

The main components of a hovercraft test bed are marked in Figure 3. Pressured air is stored in the tank. The
thrusters are used to generate forces. When the relays are actuated, the air goes through the corresponding thrusters, the reacting forces and torques will be acted on the hovercraft test bed. The planar air bearings are used to lift the hovercraft test bed off the polished marble table such that the hovercraft test bed can move on the table with little friction. The wireless actuator is used to receive control commands from the controller (an Android cellphone) through wireless networks, then the received signals will be magnified and outputted to corresponding relays. The optical markers are used by the overhead cameras to capture the location and the attitude of the test bed.

The experimental platform is illustrated in Figure 4 where the dotted lines represent the information flows. The infrared cameras and the Vicon server constitute the positioning system, which is used to capture the position and attitude of the hovercraft test bed. Based on the feedback information from the positioning system, the Android cellphone computes the control commands for the hovercraft test bed. Then the control commands are sent to the wireless actuator. All information exchange is accomplished through the Wireless Local Area Network in the laboratory.

Software Netconlink is adopted to implement the CLNPC strategy. Details of the software can be found in the literature.43

The model of the hovercraft test bed can be expressed as (1),19,44 where the matrices are $M = \text{diag}(m, m, J)$

$R^E_B = \begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $D(\nu) = \begin{bmatrix} 0 & \omega & 0 \\ -\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

with $m = 17.2\text{kg}$ and $J = 1.03\text{kg} \cdot \text{m}^2$ the mass and the moment of inertia of the hovercraft test bed, respectively.

The given paths are sinusoid curves $y_i(x_i) = y_{i0} + b_i \sin(ax_i)$, $i = 1, 2, 3$, where $b_i \in \mathbb{R}$, $y_{i0} \in \mathbb{R}$, $a$ is a constant. The reference orientation $\phi_i$ is set to be tangent to the given path, that is, $\phi_i = \tan^{-1}\left(\frac{\partial y_i}{\partial x_i}\right)$. The formation velocity is set as $v_f = 0.05\text{ m/s}$. $x_i$ is chosen as the path parameter, that is, $\theta_i = x_i$.

The communication topology is given in Figure 5, where each vertex represents a hovercraft test bed. It is obvious that the communication topology is connected. The incidence matrix of the graph can be derived as

$M = \begin{bmatrix} -1 & -1 \\ 1 & 0 \\ 0 & +1 \end{bmatrix}$
The path-following controller is designed utilizing the backstepping technique, then the closed-loop subsystem takes the following form

$$\dot{z}_i = H_i(x_i)z_i + I_i(t,x_i,\theta)\theta_i$$

(34)

where $z_i \in \mathbb{R}^{n_i}$ is the state that includes the path-following error and its derivatives. $H_i(x_i)$ and $I_i(t,x_i,\theta)$ are system matrices with proper dimensions. It can be verified that Assumption 1 is satisfied.

For a given $\rho_0 < \infty$, the corresponding control law can be derived by adopting the backstepping technique in the literature. Combining with the given set $S_{zi}$, $K_z$ and $K_w$, can be derived from (25) and (28), respectively.

The compact sets $S_{zi}$ and $S_w$ are defined by $||z_i|| < 2$ and $||w_i|| < 0.1$, respectively. The parameters of the CLNPC strategy are designed as: $\rho_0 = 1$, $K_z = 0.5$, $K_w = 0.1$, $\mu = 0.5$, $K_1 = \text{diag}(0.03, 0.03, 0.03)$, $K_3 = \text{diag}(0.1, 0.1, 0.1)$, $Q = 0.1I$. The matrix $P$ can be achieved by the linear matrix inequality technique such that the conditions of theorem 1 are satisfied.

The sampling period of the local subsystem (34) is set to be 0.2 s, while $T = 0.6$ s for the subsystem (7). Experiments when $c = 0T$ (no time delays), $c = 5T$ (3.0 s) are implemented to exhibit the effectiveness of the CLNPC strategy. In the following figures, “Ref” is the abbreviation of “Reference,” “$T_i$” $i = 1, 2, 3$ represents “Hovercraft test bed $i$.”

**Simulation results**

Figures 1 and 2 in the supplement files give the trajectories and orientations of the hovercraft test beds, respectively. It can be seen that each hovercraft test bed can track its predefined path efficiently despite that the orientation is slightly affected by the time delays. This is because the time delays occur in the communication channels among hovercraft test beds thus have little effect on the path-following performance of individual hovercraft test bed.

The velocities of the hovercraft test beds are compared in Figure 6. The static error of the formation velocity increases from 0 m/s to 0.01 m/s after the time delays are introduced into the system, while the static error is reduced to 0 m/s after the CLNPC strategy is applied.

The time evolution of the path parameters $\theta_i$, $i = 1, 2, 3$ is demonstrated in Figure 7. The largest static distance of $\theta_i$ among the hovercraft test beds increases from 0.01 m to 0.043 m, while this index is reduced to 0.02 m after the CLNPC strategy is applied.

The time evolution of the predicting errors $\hat{e}_i(k + c|k), i = 1, 2, 3$ is demonstrated in Figure 8, it can be seen that the predicting errors converge to 0 rapidly.

From the figures above, we can see that the communication time delays have negative effects on the coordination performance of the system, while the coordination performances are improved obviously after the CLNPC strategy is applied.

**Experimental results**

The trajectories of the hovercraft test beds are compared in Figure 9. As the simulation results, the path-following performance for each hovercraft test bed is almost not affected by the communication time delays.

The velocities of the hovercraft test beds are compared in Figure 10. The communication time delays enlarge the static error of the formation velocity (from 0.02 m/s to 0.1 m/s), while this static error is reduced to 0.04 m/s after the CLNPC strategy is applied.

The time evolution of the path parameters is demonstrated in Figure 11. The largest static distance of $\theta_i$ among the hovercraft test beds increases from 0.054 m to 0.069 m after communication time delays are introduced, while this
index is reduced to 0.048 m after the CLNPC strategy is applied.

The trajectories of the predicting errors $\hat{e}_i$, $i = 1, 2, 3$ are illustrated in Figure 12. It can be seen that the predicting errors converge to 0, which validate the achievement of the control objective (14).

Both the simulation results and the experimental results exhibit that the time delays have manifest effects on coordination performances, that is, control objectives (4) and (5). In contrast, the individual path-following performance, that is, control objective (3), is almost not affected. In addition, the results validate the convergence of the predicting error, that is, the achievement of the control objective (14). In conclusion, the proposed CLNPC strategy can compensate for the communication time delays effectively thus improves the coordination control performances. The proposed NPC strategy can guarantee the achievement of the control objectives (3), (4), (5), and (14) if the upper bound of the time delays satisfy $c \leq 9T$, otherwise, the predicting errors diverge.

Obvious quantitative differences can be observed between simulation results and experimental results such as the difference in static errors of the formation velocity, this is mainly caused by model mismatches and/or external disturbances. The major factors that have heavy effects on the control performance include the dead-zone nonlinearity of the relays, the parameter variation of the thrusting system. Besides, the time delays in simulation strictly less or equal to the assumed upper bound while the communication time delays in practice may violate the assumed upper bound. Despite of the quantitative differences between the simulation results and the experimental results, qualitative consistency can be observed between the simulation results and the experimental results, for example, the communication time delays enlarge the static error of the formation velocity in both simulations and experiments. Simulations can validate the effectiveness of the proposed control scheme and the correctness of the theoretical analysis in ideal circumstances while experiments can test the control performance of the proposed control scheme in a more practical environment where disturbances are inevitable.
Conclusion

In this article, the authors investigate the formation control problem for networked USVs. A novel CLNPC strategy is proposed to attenuate the negative effects on the control performance of the fleet brought by network-induced time delays. Theoretical analysis is given to derive the conditions to guarantee the performance of the proposed CLNPC strategy. Experiments on hovercraft test beds are carried out to test the performance of the control scheme, and both the simulation results and the experimental results validate the effectiveness of the CLNPC strategy.
Declaration of conflicting interests
The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported in part by the National Natural Science Foundation of China (grant nos: 61673084, 61690212, 61773144) and by the Key Laboratory of Intelligent Perception and Advanced Control of State Ethnic Affairs Commission (grant no. MD-IPAC-2019103).

ORCID iDs
Dong-Liang Chen https://orcid.org/0000-0003-3637-9764
Ru-Bo Zhang https://orcid.org/0000-0002-6133-8078

Supplemental material
Supplemental material for this article is available online.

References
1. Liu Z, Zhang Y, Xiang Yu, et al. Unmanned surface vehicles: an overview of developments and challenges. Annu Rev Control 2016; 41: 71–93.
2. Bertram V. Unmanned surface vehicles—a survey. Skibsøknisk Selskab, Copenhagen, Denmark 2008; 1: 1–14.
3. Sun X, Wang G, Fan Y, et al. Fast collision avoidance method based on velocity resolution for unmanned surface vehicle. In: 2019 Chinese control and decision conference (CCDC), Nanchang, 3–5 June 2019, pp. 4822–4827. Washington, DC: IEEE.
4. Larson J, Bruch M, and Ebken J. Autonomous navigation and obstacle avoidance for unmanned surface vehicles. In: Unmanned systems technology VIII, vol. 6230, p. 623007. Pittsfield, MA, USA: International Society for Optics and Photonics, 2006.
5. Sun X, Wang G, and Fan Y. Model identification and trajectory tracking control for vector propulsion unmanned surface vehicles. Electronics 2020; 90 (1): 22.
6. Yu R, Qi H, Upadhyay K, et al. Sliding-mode control with switching-gain adaptation for trajectory tracking of underactuated unmanned surface vessels. In: ASME 2019 dynamic systems and control conference, Park City, UT, USA, 8–11 October 2019. New York, NY: American Society of Mechanical Engineers Digital Collection.
7. Sinisterra AJ, Dhanak MR, and Ellenrieder KV. Stereo vision-based target tracking system for an USV. In: 2014 Oceans—St. John’s, St. John’s, Canada, 14–19 September 2014, pp 1–7. Washington, DC: IEEE.
8. Sinisterra AJ, Dhanak MR, and Ellenrieder KV. Stereovision-based target tracking system for usv operations. Ocean Eng 2017; 133: 197–214.
9. Han J, Son N-S, and Kim J. Automatic target tracking with time-delayed measurements for unmanned surface vehicles. In: SPIE future sensing technologies, vol. 11197, p. 111971B. Pittsfield, MA, USA: International Society for Optics and Photonics, 2019.
10. Caccia M, Bibuli M, Bono R, et al. Basic navigation, guidance and control of an unmanned surface vehicle. Auton Robot 2008; 250(4): 349–365.
11. Ihle IF, Arcek M, and Fossen TI. Passivity-based designs for synchronized path-following. Automatica 2007; 430 (9): 1508–1518.
12. Fossen TI and Lekkas AM. Direct and indirect adaptive integral line-of-sight path-following controllers for marine craft exposed to ocean currents. Int J Adapt Control Signal Process 2017; 310(4): 445–463.
13. Fossen Thor I., Kristin Y., et al. Line-of-sight path following for dubins paths with adaptive sideslip compensation of drift forces. IEEE Trans Control Syst Technol 2015; 230(2): 820–827.
14. Li M, Guo C, and Yu H. Filtered extended state observer based line-of-sight guidance for path following of unmanned surface vehicles with unknown dynamics and disturbances. IEEE Access 2019; 7: 178401–178412.
15. Bibuli M, Gasparri A, Priolo A, et al. Virtual target based path-following guidance system for cooperative USV swarms. IFAC Proc Vol 2012; 450(27): 362–367.
16. Wang N, Sun Z, Yin J, et al. Fuzzy unknown observer-based robust adaptive path following control of underactuated surface vehicles subject to multiple unknowns. Ocean Eng 2019; 176: 57–64.
17. Mata EX. Time-critical cooperative path-following control of multiple unmanned aerial vehicles. Dissertations & Theses – Gradworks. Berlin: Springer, 2013.
18. Borhaug E, Pavlov A, Ghabcheloo R, et al. Formation control of underactuated marine vehicles with communication constraints. In Proceedings 7th IFAC Conference on Manoeuvring and Control of Marine Craft, 2006.
19. Skjetne R, Fossen TI, and Kokotovic PV. Robust output maneuvering for a class of nonlinear systems. Automatica 2004; 400(3): 373–383.
20. Wang H, Wang D, and Peng ZH. Adaptive neural control for cooperative path following of marine surface vehicles: state and output feedback. Int J Syst Sci 2016; 470(2): 343–359.
21. Doosthoseini A and Nielsen C. Coordinated path following for unicycles: a nested invariant sets approach. Automatica 2015; 60: 17–29.
22. Almeida J, Silvestre C, and Pascoal AM. Cooperative control of multiple surface vessels with discrete-time periodic communications. Int J Robust Nonlinear Control 2012; 220(4): 398–419.
23. Liu Z-Q, Wang Y-L, and Wang T-B. Incremental predictive control-based output consensus of networked unmanned surface vehicle formation systems. Inf Sci 2018; 457: 166–181.
24. Wang R, Jing H, Hu C, et al. Robust $h_\infty$ path following control for autonomous ground vehicles with delay and data dropout. IEEE Trans Intell Trans Syst 2016; 170(7): 2042–2050.
25. Millan P, Orihuela L, Jurado I, et al. Formation control of autonomous underwater vehicles subject to communication
delays. *IEEE Trans Control Syst Technol* 2014; 220(2): 770–777.

26. Zuo Z and Tie L. Distributed robust finite-time nonlinear consensus protocols for multi-agent systems. *Int J Syst Sci* 2016; 47(6): 1366–1375.

27. Gao H and Chen T. Network-based $h_{\infty}$ output tracking control. *IEEE Trans Autom Control* 2008; 530(3): 655–667.

28. Tan C, Liu G-P, and Shi P. Consensus of networked multi-agent systems with diverse time-varying communication delays. *J Frankl Inst* 2015; 3520(7): 2934–2950.

29. Wei Y and Liu GP. Consensus tracking of heterogeneous discrete-time networked multiagent systems based on the networked predictive control scheme. *IEEE Trans Cybern* 2017; 47(8): 2173–2184.

30. Zhonghua P, Liu G, and Zhou D. Data-based predictive control for networked nonlinear systems with packet dropout and measurement noise. *J Syst Sci Complex* 2017; 300(5): 1072–1083.

31. Yang R, Liu G-P, Shi P, et al. Predictive output feedback control for networked control systems. *IEEE Trans Ind Electron* 2013; 60(1): 512–520.

32. Liu G-P, Xia Y, Chen J, et al. Networked predictive control of systems with random network delays in both forward and feedback channels. *IEEE Trans Ind Electron* 2007; 54(3): 1282–1297.

33. Liu G-P, Xia Y, Rees D, et al. Design and stability criteria of networked predictive control systems with random network delay in the feedback channel. *IEEE Trans Syst Man Cybern C Appl Rev* 2007; 370(2): 173–184.

34. Tan C and Liu GP. Consensus of discrete-time linear networked multi-agent systems with communication delays. *IEEE Trans Autom Control* 2013; 580(11): 2962–2968.

35. Liu GP. Predictive controller design of networked systems with communication delays and data loss. *IEEE Trans Circuits Syst II Express Br* 2010; 570(6): 481–485.

36. Subha NAM and Liu GP. Design and practical implementation of external consensus protocol for networked multiagent systems with communication delays. *IEEE Trans Control Syst Tech* 2015; 230(2): 619–631.

37. Fossen TI. *Handbook of marine craft hydrodynamics and motion control*. Hoboken: John Wiley & Sons, 2011.

38. Chobotov VA. *Spacecraft attitude dynamics and control*. NASA STI/Recon Technical Report A, 92. Malabar, FL: Krieger Publishing Co, 1991.

39. Ghabcheloo R, Pascoal A, Silvestre C, et al. Nonlinear coordinated path following control of multiple wheeled robots with bidirectional communication constraints. *Int J Adapt Control Signal Process* 2007; 21: 133–157.

40. Luo ACJ. Discretization and implicit mapping dynamics. In: *Nonlinear physical science* (eds. CJ Albert, HN Luo and Ibragimov), Berlin: Springer, 2015, pp. 56–59.

41. Laila DS, Nesic D, and Teel AR. Open and closed loop dissipation inequalities under sampling and controller emulation. *Eur J Cont* 2002l; 80 (2): 109–125.

42. Byk E and Arcak M. *Passivity-based agreement protocols: continuous-time and sampled-data designs*. Berlin: Springer, 2006.

43. Chen D and Liu G. A real-time networked control framework based on mobile phones. In: *2015 IEEE international conference on mechatronics and automation (ICMA)*, Beijing, 2–5 August 2015, pp. 1772–1778. Washington, DC: IEEE.

44. Fantoni I, Lozano R, Mazenc F, et al. Stabilization of a nonlinear underactuated hovercraft. In: *Proceedings of the 38th IEEE conference on decision and control*, vol. 3, Phoenix, AZ, USA, 7–10 December 1999, pp. 2533–2538. Washington, DC: IEEE.