Distributed Leader-Follower Formation Control for Autonomous Vessels based on Model Predictive Control

van Pampus, M.J. ; Haseltalab, A.; Garofano, V.; Reppa, V.; Deinema, Y.H.; Negenborn, R.R.

DOI
10.23919/ECC54610.2021.9654989

Publication date
2021

Document Version
Final published version

Published in
Proceedings of the European Control Conference (ECC 2021)

2021 European Control Conference (ECC)
June 29 - July 2, 2021. Rotterdam, Netherlands

Distributed Leader-Follower Formation Control for Autonomous Vessels based on Model Predictive Control*

M.J. van Pampus[1,2], A. Haseltalab[1], V. Garofano[1], V. Reppa[1], Y.H. Deinema[2], R.R. Negenborn[1]

Abstract— **Formation control of autonomous surface vessels (ASVs) has been studied extensively over the last few years since it offers promising advantages. In this paper, two control methods for distributed leader-follower formation control are proposed: A Nonlinear Model Predictive Control (MPC) method and an MPC method using Feedback Linearization. One agent per vessel performs planning and control. The agents exchange information on their current and predicted positions. The two proposed methods are compared with each other and also with a conventional Proportional-Integral (PI) control method. The performance of the proposed strategies is evaluated through simulations and field experiments using small scale vessels. The simulation and field experiment results show that the proposed MPC-based approaches outperform the conventional PI control method.**

Index Terms— **Autonomous surface vessels, model predictive control, feedback linearization, formation control**

I. INTRODUCTION

Formation control is seen as a promising opportunity for autonomous surface vessels (ASVs) and has been studied for years. Advantages of formation control include robustness, reliability and efficiency [1]. Furthermore, formation control is useful for applications such as search and rescue [2], ocean sampling [3], large object transport [4], and surveillance [5]. Autonomous ground, aerial, underwater and surface vehicles are among the main applications of formation control. Most research works have focused on ground and aerial vehicles [6], [7].

Four approaches of formation control have been developed. The behavior-based approach weighs the relative importance of desired prescribed behavior for each vehicle [8], [9]. The virtual structure approach handles the formation as a single entity and assigns a desired motion to the center of the virtual structure [10], [11]. A third method is graph theory that uses mathematical structures to model the relations between vehicles [6], [12], [13]. In the fourth approach, the leader-follower approach, a designated leader is followed by the followers respecting user-defined distances between the leader and followers [14], [15], [16]. The suitability of each approach depends on the purpose of the application.

In the case of multiple vessels, three main control structures have been used for formation control, namely centralized, decentralized and distributed control. While many research works apply centralized control [1], [11], [17], it is shown that centralized control does not scale well and is thus less suited for formation control [1], [18], [19]. To cope with these issues decentralized and distributed approaches have been applied. In decentralized control approaches, vehicles do not communicate [8], [14] in contrast to distributed control, where they do communicate to exchange information [1], [6], [18].

There is a wide range of applications of formation control. In [20] a Proportional-Integral-Derivative (PID) controller is used to control a formation of tailsitters. Linear-Quadratic Regulator is used to control quadcopters in [21]. The Backstepping approach is proposed in [10] to hold a desired intership formation pattern in a group of autonomous surface vehicles. Neural Networks are used in [16] to control a group of vessels using leader-follower formation control. In [6], Model Predictive Control (MPC) is applied to a group of quadrotors that is able to avoid dynamic obstacles. While MPC is frequently used and validated through simulations, research endeavors to validate MPC strategies are limited [7], [12]. To our best knowledge, no maritime formation control field experiments have been performed with predictive control. The main reason for this seems to be the high computation time [22]. The lack of field experiments with predictive control seems to be a major research gap in maritime formation control.

The objective of this work is to design a distributed leader-follower formation control architecture of multiple vessels where the leader tracks a trajectory and the followers keep a safe distance. The leader-follower formation is suitable for sailing in narrow waterways. Per vessel, one agent performs planning and control and the agents exchange their current and predicted positions. A Guidance, Navigation and Control (GNC) strategy is followed for the design of every agent. The guidance system solves an MILP optimization problem in order to produce a feasible, collision free trajectory that follows a desired trajectory as close as possible. Two different control strategies are used in this research work. The first strategy uses an NMPC algorithm, that has a high computation time. The second strategy uses a fast FL MPC algorithm. A model-free control approach is used for comparison. The performance of the strategies are evaluated

*This research is supported by the Amsterdam Institute for Advanced Metropolitan Solutions (AMS Institute), the Researchlab Autonomous Shipping (RAS) at Delft University of Technology, the European Regional Development Fund, as part of the Interreg North Sea Region project AVATAR, and by the project "GasDrive: Minimizing emissions and energy losses at sea with LNG combined prime movers, underwater exhausts and nano hull materials" (project 14504) of the Netherlands Organization for Scientific Research (NWO), domain Applied and Engineering Sciences (TTW).

[1]Department of Maritime and Transport Technology, Delft University of Technology, Delft, the Netherlands {a.haseltalab,v.garofano,v.reppa,r.r.negenborn}@tudelft.nl

[2]Amsterdam Institute for Advanced Metropolitan Solutions, Amsterdam, the Netherlands ynse.deinema@ams-institute.org

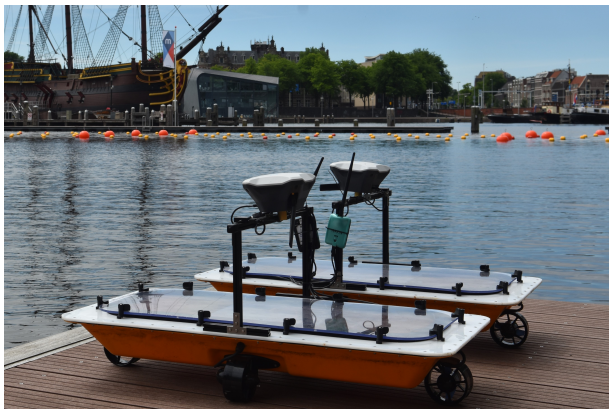


Fig. 1: 1:4 scale Roboat units on the quay in Amsterdam

using six simulation tests and five field experiments using 1:4 scale Roboat units [23], shown in Figure 1.

The main contributions of this work are:

- The design of a modular distributed GNC architecture where information is exchanged in the guidance system instead of the control system.
- The design of a reconfigurable formation control strategy to produce and follow a collision-free trajectory.
- The design and performance evaluation of two model predictive control strategies, a nonlinear MPC and a Feedback Linearization based MPC, and a model-free control strategy, a PI.

The remainder of this paper is organized as follows. In Section II the maneuvering model is presented. In Section III, the MPC-based approaches are described. In Section IV the results of simulations and field experiments are presented. Finally, concluding remarks are given in Section V.

II. VESSEL MANEUVERING DYNAMICS

According to [24] the maneuvering of a vessel can be described in 3 Degrees of Freedom by the following nonlinear differential equation:

$$\mathbf{M}_j\dot{\nu}_j + \mathbf{C}_j(\nu_j)\nu_j + \mathbf{D}_j\nu_j = \tau_j + \mathbf{d}_j, \tag{1}$$

where $\nu_j = [u_j, v_j, r_j]^{\mathrm{T}}$ denotes the velocity; $\mathbf{M}_j \in \mathbb{R}^{3\times3}$ is the added mass and inertia matrix; $\mathbf{C}_j(\nu_j) \in \mathbb{R}^{3\times3}$ is the Coriolis and centripetal matrix; $\mathbf{D}_j \in \mathbb{R}^{3x3}$ is the drag matrix; $\tau_j \in \mathbb{R}^{3\times1}$ is the vector of internal forces and moments applied to the vessel; and $\mathbf{d}_j \in \mathbb{R}^{3\times1}$ the vector of unknown external forces and moments applied to the vessel. The position and orientation of the ship in the North-East-Down (NED) coordinate system is defined as $\eta_j = [x_j, y_j, \psi_j]^{\mathrm{T}}$. The kinematic equation relating the velocity components in the NED frame to those in the body fixed frame is:

$$\dot{\eta}_j = \mathbf{R}(\psi_j)\nu_j, \tag{2}$$

where $\mathbf{R}(\psi_j)$ is a Jacobian matrix described by:

$$\mathbf{R}(\psi_j) = \begin{bmatrix} cos(\psi_j) & -sin(\psi_j) & 0 \\ sin(\psi_j) & cos(\psi_j) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

Assuming that the origin of the body-fixed frame of the vessel coincides with the center of mass and that the vessel is moving at low speeds, the mass matrix (M_j), matrix of Coriolis and centripetal forces (C_j), and the drag matrix (D_j) can be described as follows according to [25]:

$$\mathbf{M}_j = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & 0 \\ 0 & 0 & m_{33} \end{bmatrix} \tag{4}$$

$$\mathbf{C}_j = \begin{bmatrix} 0 & 0 & -m_{22}v_j \\ 0 & 0 & m_{11}u_j \\ m_{22}v_j & -m_{11}u_j & 0 \end{bmatrix} \tag{5}$$

$$\mathbf{D}_j = \begin{bmatrix} X_u & 0 & 0 \\ 0 & Y_v & 0 \\ 0 & 0 & N_r \end{bmatrix}, \tag{6}$$

where m_{ij} represents the elements of the mass matrix and X_u, Y_v, and N_r represent the elements of the drag matrix.

Furthermore, τ_j can be written as function of the control vector $\mathbf{u}_j \in \mathbb{R}^{m\times1}$ that contains the forces produced per actuator, using thruster allocation matrix $\mathbf{B}_j \in \mathbb{R}^{3\times m}$, where m denotes the amount of actuators:

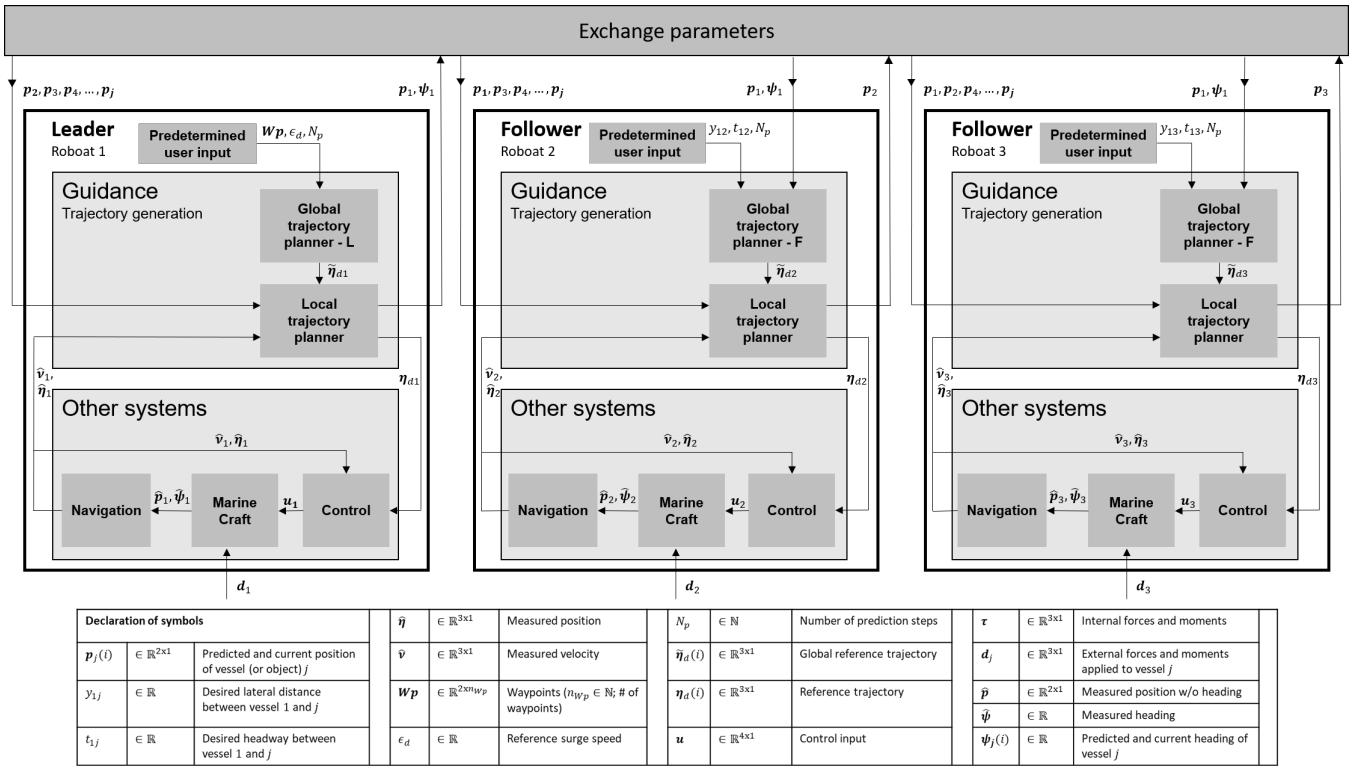
$$\tau_j = \mathbf{B}_j\mathbf{u}_j, \tag{7}$$

III. DISTRIBUTED FORMATION CONTROL ARCHITECTURE

The control structure used in this research is a guidance, navigation, and control (GNC) structure, as described in [24]. In this section, a detailed description of the guidance and control system is given. The guidance system is designed in such a way that distributed formation control is established. The architecture of the distributed leader follower formation control problem for the Roboat units is shown in Figure 2. The vessels share their current and predicted positions with each other. The follower Roboat units base their trajectory on the current and predicted positions of the leader Roboat unit. The positions of other vessels, together with positions of objects are taken into account in the local trajectory planner of the guidance system in order to produce a collision-free trajectory. In this section, first the proposed guidance system is presented. Afterwards, the proposed control systems are discussed.

A. Guidance system

For usage in inland waterways, the guidance system of the leader vessel must be able to produce a collision free route through the navigable area. Furthermore, the follower vessels have to determine their trajectory based on the trajectory of the leader. This is done by creating a *global trajectory planner* and a *local trajectory planner*. The global trajectory of the leader is based on user input, while the global trajectory for the followers is based on the (predicted)



Declaration of symbols						N_p	$\in \mathbb{N}$	Number of prediction steps	$\boldsymbol{\tau}$	$\in \mathbb{R}^{3x1}$	Internal forces and moments
$\boldsymbol{p}_j(i)$	$\in \mathbb{R}^{2x1}$	Predicted and current position of vessel (or object) j	$\tilde{\boldsymbol{\eta}}$	$\in \mathbb{R}^{3x1}$	Measured position	$\bar{\boldsymbol{\eta}}_d(i)$	$\in \mathbb{R}^{3x1}$	Global reference trajectory	$\boldsymbol{d}_j$	$\in \mathbb{R}^{3x1}$	External forces and moments applied to vessel j
y_{1j}	$\in \mathbb{R}$	Desired lateral distance between vessel 1 and j	$\hat{\boldsymbol{\nu}}$	$\in \mathbb{R}^{3x1}$	Measured velocity	$\boldsymbol{\eta}_d(i)$	$\in \mathbb{R}^{3x1}$	Reference trajectory	$\tilde{\boldsymbol{p}}$	$\in \mathbb{R}^{2x1}$	Measured position w/o heading
t_{1j}	$\in \mathbb{R}$	Desired headway between vessel 1 and j	$\boldsymbol{Wp}$	$\in \mathbb{R}^{2xn_{Wp}}$	Waypoints ($n_{Wp} \in \mathbb{N}$; # of waypoints)	$\boldsymbol{u}$	$\in \mathbb{R}^{4x1}$	Control input	$\tilde{\boldsymbol{\psi}}$	$\in \mathbb{R}$	Measured heading
			ϵ_d	$\in \mathbb{R}$	Reference surge speed				$\boldsymbol{\psi}_j(i)$	$\in \mathbb{R}$	Predicted and current heading of vessel j

Fig. 2: A schematic overview of the distributed formation control architecture.

sailing trajectory of the leader. The local trajectory planner produces a reference trajectory that minimizes the distance to the global trajectory, while making sure that the trajectory is collision free and inside the navigable area. In the following paragraphs the guidance algorithms for the leader and follower Roboat are explained.

a) Leader Roboat: The global trajectory planner for the leader Roboat has as user inputs: 1) waypoints W_p; 2) the desired surge speed ϵ_d; 3) the sampling time t_s; and 4) the prediction horizon N_p. With this input, a global reference trajectory is formed.

The local trajectory planner determines the reference trajectory based on the following optimization problem for $j \in \{2, \ldots, m\}$ (m is the number of considered vessels and obstacles):

$$J = \min_{\mathbf{p}_1} \sum_{i=0}^{N_p} \|\tilde{\mathbf{p}}_1(k+i) - \mathbf{p}_1(k+i)\| \tag{8}$$

subject to

$$\mathbf{p}_1(k) = \hat{\mathbf{p}}_1(k) \tag{9}$$

$$\|\mathbf{p}_1(k+i) - \mathbf{p}_j(k+i)\| \geq d_{\text{safe}},\ i = 0, \ldots, N_p, \quad j = 2, \ldots, m, \tag{10}$$

$$v_{\min} \leq \frac{\|\mathbf{p}_1(k+i) - \mathbf{p}_1(k+i-1)\|}{t_s} \leq v_{\max}, \quad i = 1, \ldots, N_p \tag{11}$$

where $\mathbf{p}_1(i) = \begin{bmatrix} x_d \\ y_d \end{bmatrix} \in \mathbb{R}^{2\times 1}$ is the reference position at time step i; $\tilde{\mathbf{p}}_1(i) \in \mathbb{R}^{2\times 1}$ is the global reference position; $\hat{\mathbf{p}}_1(k) \in \mathbb{R}^{2\times 1}$ is the measured position; $\mathbf{p}_j(i) \in \mathbb{R}^{2\times 1}$ is the expected position of object or vessel $j \in \mathbb{N}$, where $j = 1$ is the index for the own vessel and m is the total amount of considered objects and vessels; $d_{\text{safe}} \in \mathbb{R}_{>0}$ is the safety distance between the center of gravity of the vessel with its neighbor vessels; $v_{\min} \in \mathbb{R}$ is the minimum speed; $v_{\max} \in \mathbb{R}$ is the maximum speed; $t_s \in \mathbb{R}_{>0}$ is the sampling time; and $N_p \in \mathbb{N}$ is the number of prediction steps.

The boundaries of inland waterways are considered a sequence of point masses, represented by $\mathbf{p}_j$. Large objects or vessels can also be modeled as sequences of points masses, to ensure a correct working of the collision avoidance.

This optimization problem is transformed into an MILP problem and solved at every time step k. Using the reference position, the reference trajectory is calculated as follows for $i \in \{1, \ldots, N_p\}$:

$$\psi_d(k+i) = \arctan\left(\frac{y_d(k+i) - y_d(k+i-1)}{x_d(k+i) - x_d(k+i-1)}\right), \tag{12}$$

$$\eta_d(k+i) = \begin{pmatrix} \mathbf{p}_1(k+i) \\ \psi_d(k+i) \end{pmatrix} \equiv \begin{pmatrix} x_d(k+i) \\ y_d(k+i) \\ \psi_d(k+i) \end{pmatrix}, \tag{13}$$

where $\eta_d(i) \in \mathbb{R}^{3\times 1}$ is the reference trajectory at time step i, consisting of the components in x, y, and ψ direction: $x_d(i)$, $y_d(i)$, and $\psi_d(i)$.

b) Follower Roboat: The global trajectory planner for the follower Roboat determines the global reference trajectory based on the trajectory of the leader, a user determined headway, and a user determined desired lateral distance to the leader. This leads to the following equations where η_1 indicates the position of the leader Roboat, η_j the position of follower j and $n_{1\mathrm{j}}$ is an integer that represents the desired headway ($t_{1\mathrm{j}}$) in number of sampling times (t_s):

$$n_{1\mathrm{j}} \approx \frac{t_{1\mathrm{j}}}{t_s} \tag{14}$$

$$\eta_1 = \begin{pmatrix} \mathbf{p}_1 \\ \psi_1 \end{pmatrix} \tag{15}$$

$$\tilde{\eta}_d(k+i) = \mathbf{R}\big(\psi_1(a)\big)\Big(\eta_1^b(a) + \begin{pmatrix} 0 \\ y_{1\mathrm{j}} \\ 0 \end{pmatrix}\Big) \qquad i = 1, \ldots, N_p; \qquad a = k + i - n_{1\mathrm{j}}. \tag{16}$$

Although it has not been tested in this research, the desired headway and lateral distance could change during a test to achieve dynamically changing formations. The local trajectory planner of the follower Roboat units is the same as that of the leader Roboat.

B. Control system

The control system determines the control input of the vessel's thrusters, based on the reference trajectory and measured states. In this paper, two distributed control approaches are proposed: Nonlinear Model Predictive Control (NMPC) and Feedback Linearization Model Predictive Control (FL-MPC).

a) NMPC: The first control method uses Non-linear Model Predictive Control (NMPC). The objective function of the NMPC control method is the following:

$$J = \min_{\mathbf{u}(k+i)} \sum_{i=1}^{N_p} \Bigg(\Big(\eta(k+i) - \eta_d(k+i)\Big)^{\mathrm{T}} \mathbf{Q}_{\mathrm{NMPC}} \Big(\eta(k+i) - \eta_d(k+i)\Big) + \mathbf{u}(k+i)^{\mathrm{T}} \mathbf{R}_{\mathrm{NMPC}} \mathbf{u}(k+i) \Bigg) \tag{17}$$

subject to

$$\mathbf{x}(k+i|k) = \mathbf{x}(k) + \int_{kt_s}^{(k+i)t_s} \mathbf{f}(\mathbf{x}, \tau) dt \qquad i = 1, \ldots, N_{\mathrm{P}}; \tag{18}$$

$$\mathbf{x}(k+i) = \begin{bmatrix} \eta(k+i) \\ \nu(k+i) \end{bmatrix}; \tag{19}$$

$$\tau_d(k+i) = \mathbf{B}\mathbf{u}(k+i); \tag{20}$$

$$\mathbf{u}_{\min} \leq \mathbf{u}(k+i) \leq \mathbf{u}_{\max}. \tag{21}$$

where $\mathbf{Q}_{\mathrm{NMPC}} \in \mathbb{R}^{3\times3}_{>0}$ and $\mathbf{R}_{\mathrm{NMPC}} \in \mathbb{R}^{3\times3}_{>0}$ are positive definite penalty matrices, $\mathbf{x}_j = \begin{bmatrix} \eta_j^{\mathrm{T}} & \nu_j^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ with the function $\mathbf{f}$ defined based on (1)-(2), i.e.,

$$\begin{aligned} \dot{\mathbf{x}}_j =& \mathbf{f}(\mathbf{x}_j, \tau_j, \mathbf{d}_j) \\ =& \begin{bmatrix} \mathbf{0}^{3\times3} & \mathbf{R}(\psi_j) \\ \mathbf{0}^{3\times3} & \mathbf{M}_j^{-1}(-\mathbf{C}_j(\nu_j) - \mathbf{D}_j) \end{bmatrix} \mathbf{x}_j + \\ & \begin{bmatrix} \mathbf{0}^{3\times3} \\ \mathbf{M}_j^{-1} \end{bmatrix} \tau_j + \begin{bmatrix} \mathbf{0}^{3\times3} \\ \mathbf{M}_j^{-1} \end{bmatrix} \mathbf{d}_j \end{aligned} \tag{22}$$

The dynamic model (22) is discretized with sampling time t_s to calculate the state at time step $k+1$:

$$\mathbf{x}_j(k+1|k) = \mathbf{x}_j(k) + \int_{kt_s}^{(k+1)t_s} f(\mathbf{x}_j, \tau_j, \mathbf{d}_j) dt, \tag{23}$$

As can be noticed, disturbances are not taken into account in the NMPC control method, since they cannot be measured. Besides, this control method incorporates the thrust allocation in the optimization problem. It is suggested by [26] that this results in a more optimal control output compared to a separate thrust allocation system. It is emphasized that the system has a distributed control structure due to the communication of current and predicted positions in the guidance system.

b) FL-MPC: A downside of nonlinear MPC with linearization in each sampling time is the high computational time. To address this problem, the MPC method is combined with Feedback Linearization (FL) [27]. We use the same model as (1) -(2) and (22)-(23) but we do not take disturbances into account, since environmental disturbances are unknown. For the sake of FL, an auxiliary control input is introduced to establish a linear relationship between the system outputs and auxiliary inputs. Furthermore, the constraints of the optimization problem are linearized, enabling the solution to the problem using quadratic programming methods. This leads to significantly lower computational costs compared to the NMPC problem. The FL-MPC method is based on the velocity dynamics that can be rewritten as:

$$\dot{\nu}_j(t) = \mathbf{M}_j^{-1}\Big(\tau_j(t) - \mathbf{C}_j(\nu_j(t))\nu_j(t) - \mathbf{D}_j\nu_j(t)\Big), \tag{24}$$

The dynamical model of the system can be linearized using the FL law:

$$\tau_j(t) = \mathbf{M}_j\Big(A_j\nu_j(t) + B_j\zeta_j(t)\Big) + \mathbf{C}_j(\nu_j(t))\nu_j(t) + \mathbf{D}_j\nu_j(t), \tag{25}$$

where $\zeta_j(t) \in \mathbb{R}^{3\times1}$ is the input vector of the linear closed-loop system, and $A_j \in \mathbb{R}^{3\times3}$ and $B_j \in \mathbb{R}^{3\times3}$ are states and input matrices of the linear system, respectively. The values of A_j and B_j can be chosen by the designer as long as it leads to stability of the created linear closed-loop system.

This system can be discretized in the same way as in (23), the new matrices of the state-space representation are now indicated with A_d, B_d. Since this linear system contains only velocities and no positions, the vectors in

the dynamic system have three entries instead of six and thus the computation time is relatively low. However to accomplish this, the reference position must be transformed to a reference velocity. To this purpose, the reference velocity is calculated in two steps. First the average velocity that is required between two time steps is determined. Then this is extrapolated to the required velocity at the second time step, based on the velocity at the first time step and under the assumption that the acceleration is constant:

$$\overline{\nu}_d(k+i|k) = \mathbf{R}^{-1}(\psi(k))\frac{\eta_d(k+i) - \hat{\eta}(k)}{\beta i t_s} \qquad i = 1, \ldots, N_p; \quad (26)$$

$$\nu_d(k+i|k) = 2\Big(\overline{\nu}_d(k+i|k) - \hat{\nu}(k)\Big) + \hat{\nu}(k) \qquad i = 1, \ldots, N_p; \quad (27)$$

where $\overline{\nu}_d(k+i|k) \in \mathbb{R}^{3\times 1}$ is the average reference velocity of time step $k+i$, calculated at time step k; $\hat{\eta}_d(k) \in \mathbb{R}^{3\times 1}$ is the measured position at time step k; $\beta \in \mathbb{R}_{>0}$ is a tuning parameter; $t_s \in \mathbb{R}_{>0}$ is the sampling time; $\nu_d(k+i|k) \in \mathbb{R}^{3\times 1}$ is the reference velocity of time step $k+i$, calculated at time step k; and $\hat{\nu}(k) \in \mathbb{R}^{3\times 1}$ is the measured velocity at time step k. The following optimization problem is now created:

$$J = \min_{\zeta_j} \sum_{i=1}^{N_p} \Bigg(\Big(\nu_j(k+i) - \nu_d(k+i)\Big)^{\mathrm{T}} \mathbf{Q}_{\mathrm{FL}} \Big(\nu_j(k+i) - \nu_d(k+i)\Big)\Bigg), \quad (28)$$

subject to

$$\nu(k+i+1) = A_d\nu(k+i) + B_d\zeta(k+i), \quad (29)$$

$$\tau_{\min} \leq \tau_d(k+i) \leq \tau_{\max}, \quad (30)$$

$$\tau_d(k+i) = \mathbf{M}\Big(A_d\nu(k+i) + B_d\zeta(k+i)\Big) + \mathbf{C}\big(\hat{\nu}(k)\big)\hat{\nu}(k) + \mathbf{D}\hat{\nu}(k) \qquad i = 1, \ldots, N_p, \quad (31)$$

where $\mathbf{Q}_{\mathrm{FL}} \in \mathbb{R}^{3\times 3}_{>0}$ is a positive definite penalty matrix.

To address the thrust allocation, an optimization function that minimizes the amount of force and the amount of change of force per thruster is proposed. The proposed thrust allocation algorithm is made predictive, this is especially useful for vessels that have slow reacting actuators. The algorithm is explained in [28].

IV. SIMULATION AND FIELD EXPERIMENT RESULTS

In this section, the performance of the proposed formation control strategies is evaluated during six tests in simulation and five field experiments using 1:4 scale Roboat units [23]. First, the details of the Roboat units are presented. Then, the methodology for the experiments is discussed. Finally, the results of the simulations and field experiments are analysed.

A. System description

The model parameters of the 1:4 scale Roboat unit used, have been determined experimentally in [25] and are shown in Table I.

TABLE I: Results of the parameter identification conducted by [25].

Item	m_{11}	m_{22}	m_{33}	X_u	Y_v	N_r
Value	12.982	23.318	1.273	6.012	7.112	0.771

The thrust configuration of a Roboat unit is:

$$\mathbf{B} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ a/2 & -a/2 & b/2 & -b/2 \end{bmatrix} \quad (32)$$

where a is the distance between the transverse propellers, and b the distance between the longitudinal propellers, with $a = 0.90$ m, and $b = 0.45$ m. Each propeller is fixed and can generate forward and backward forces. Note that the designed approaches can be applied to an underactuated vessel following a similar procedure as in [29].

During the field experiments, the communication platform used is ROS [30], hosted by a pc using a quad core 1.8 GHz Intel(R) Core(TM) i7-8550U processor with 16 GB of RAM. A central pc with a simulated distributed control approach runs a MATLAB script that determines the control inputs of the Roboat units, this pc has the same processor and 8 GB of RAM. The control action is sent to the mini pcs on board the Roboat units that transform the signal to an input signal of the thrusters. Sensor data is continuously sent to the central MATLAB computer. The code on the mini pcs is written in C++. The MATLAB control loop is executed at 2 Hz.

B. Methodology

In simulation, the control methods are PI control, FL-MPC, and NMPC respectively. The first three tests are conducted by using both the global and the local trajectory planners, while the remaining three tests, the guidance system of the leader Roboat consists of only the global trajectory planner. The reason for using only the global trajectory planner is to determine the influence of the local trajectory planner on the control systems. During the field experiments, the same tests except for Test 1 are performed, and Test 4 (PID control with partial guidance system of the leader) uses FL-MPC as control system for the follower Roboat. An overview of the tests is displayed in Table II. The FL-MPC problem is solved in MATLAB using Quadratic

Programming, while the NMPC problem is solved using the ACADO toolkit [31].

TABLE II: Overview of the simulations (S1-S6) and field experiments (E1-E6)

Test	# vessels	Guidance		Control		
		Local Planner	Global Planner	PI	NMPC	FL-MPC
S1	3	all vessels	all vessels	X		
S2	3	all vessels	all vessels		X	
S3	3	all vessels	all vessels			X
S4	3	followers	all vessels	X		
S5	3	followers	all vessels		X	
S6	3	followers	all vessels			X
E1	-	-	-			
E2	2	all vessels	all vessels		X	
E3	2	all vessels	all vessels			X
E4	2	followers	all vessels	X(L)		X(F)
E5	2	followers	all vessels		X	
E6	2	followers	all vessels			X

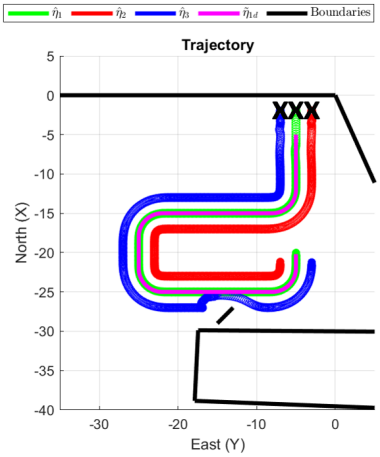
A trajectory is designed in front of the AMS Institute building on the *Marineterrein* in Amsterdam, as shown in Figures 3a and 5a. In simulation an object that has to be avoided is added. During the field experiments, it appeared that the computation time to avoid the object was too high. In simulation we considered three and during the field experiments two Roboat units. One dedicated leader Roboat is followed by the follower(s) with a headway of 5 s and lateral distance of 2 m (in simulation this is -2 m for one of the follower Roboat units). The prediction horizon for the FL-MPC method is 10 s, for the NMPC method it is 5 s.

C. Results

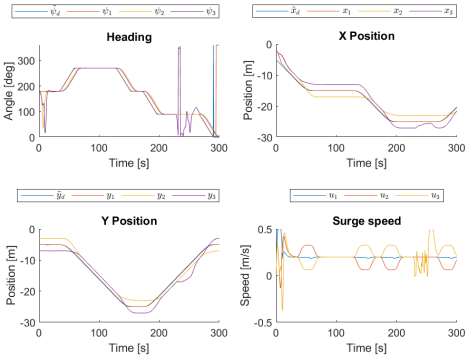
Two full sets of results are shown in this section. In Figure 3 the results of simulation test 6: NMPC & half guidance system are shown. In Figure 5 the results of field experiment test 6 are shown. In the tests, a smooth, collision-free trajectory is sailed. Besides the formation is kept wherever possible. In Figure 3 the object avoidance ability is clearly shown. In Figure 4 it is shown that the safety distance with the other vessels is kept throughout the entire field experiment test.

During the tests, each vessel is assessed on five KPIs defined in the Appendix. In short, KPI 1: RMSE trajectory guidance system shows how well the guidance system creates a reference trajectory that resembles the desired trajectory. KPI 2: RMSE trajectory control system shows how well the vessel follows the created reference trajectory. KPI 3: RMSE heading control system shows how well the actual heading resembles the desired heading of the vessel. KPI 4: Applied forces by thrusters shows how much forces the thrusters produce in total. Finally, KPI 5: Change of forces by thrusters gives insight in the amount of changes in thrust. The ideal value of all KPIs is zero.

Valuable insights provided by Figure 6 and Figure 7 that show the KPIs are the following. It can be seen that the NMPC method scored better than the other methods in most



(a) The sailed trajectory (start at X)



(b) Position over time and surge speed

Fig. 3: Results of simulation test 6

tests. The conventional PI control method scored lowest in both the simulation and field experiments.

Furthermore, the results of the simulation KPIs are normalized to the results of the PI control method. The average of these normalized KPIs are 62% and 65% for the NMPC and FL-MPC control methods respectively. This clearly demonstrates that in simulation the NMPC and FL-MPC methods have similar overall performance - tested with equally weighted KPIs - and both outperform the PI control method. The results of the KPIs of the field experiments are normalized to the values of the FL-MPC method, since only half of the PI control method tests were carried out.

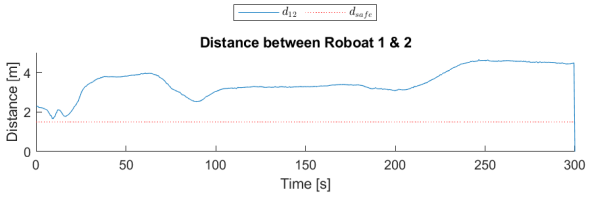
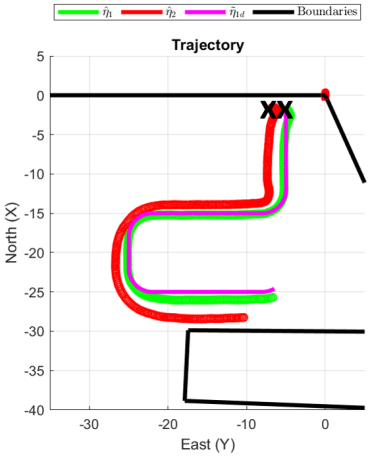
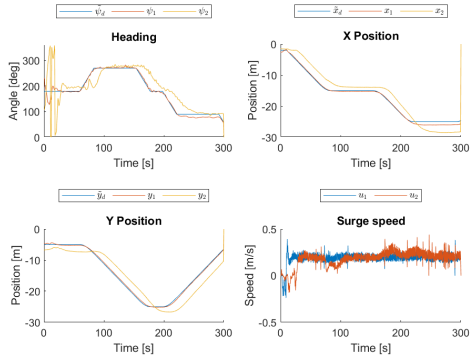


Fig. 4: Distance between vessels during field experiment test 6



(a) The sailed trajectory (start at X)



(b) Position over time and surge speed

Fig. 5: Results of field experiment test 6

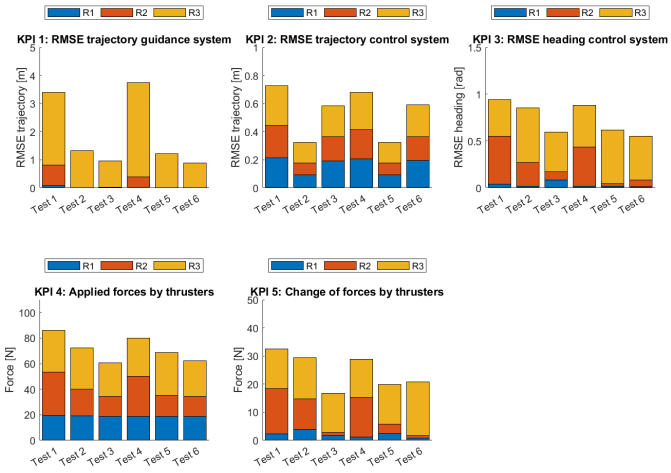


Fig. 6: KPIs during the tests in simulation after the initialization phase of 1 minute

The average of these normalized KPIs are 80% and 141% for the NMPC and PI control methods respectively. This

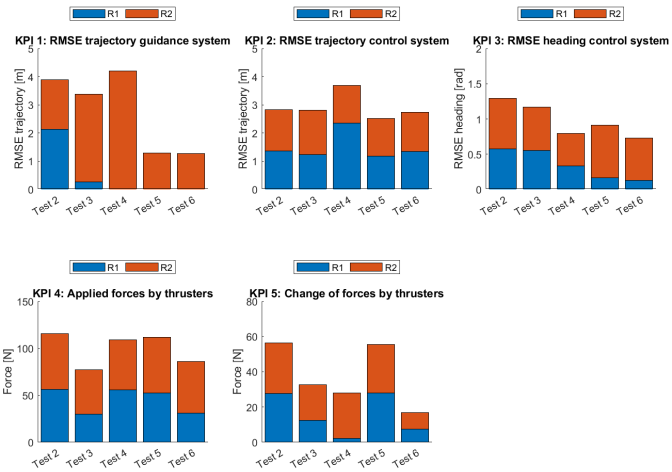


Fig. 7: KPIs during the field experiments after the initialization phase of 1 minute

demonstrates that rated on equally weighted KPIs, in practice the NMPC method performs better than the FL-MPC control method. The PI control method yields the worst results of the three control methods.

V. CONCLUSIONS AND FUTURE RESEARCH

In this paper, distributed leader-follower formation control strategies based on Model Predictive Control have been proposed for the control of autonomous vessel application in inland waterways. A guidance system using a Mixed-Integer Linear Programming (MILP) optimization problem that is able to perform object avoidance and three control systems have been designed. Both the Nonlinear Model Predictive Control (NMPC) and the Feedback Linearization Model Predictive Control (FL-MPC) method gave better results than the conventional PI control method in simulation and during field experiments. The total costs in simulation were 38 and 35 percentage points less for the NMPC and FL-MPC methods respectively compared to the PI control method. During the field experiments the total costs were 43 and 29 percentage points less for the NMPC and FL-MPC methods respectively. In future research the computation time can be lowered so the frequency of the control loop can be increased and with that the performance of the system.

Future research will involve the optimal configuration of formation based on hydrodynamic effects, dynamic model predictive control to account for different weight distributions, and combining different tasks such as entering a formation, leaving the formation, and docking. Thorough analysis on the sensitivity to modelling errors and on communication delays will also be investigated.

APPENDIX
DETERMINATION OF THE KPIS

During the tests, the following KPIs are used:

1) Root mean square error (RMSE) of the reference trajectory relative to the global trajectory (guidance):

$$\sqrt{\frac{\sum_{k=1}^{N}\left(\left(\left\|\mathbf{p}_d(k)-\tilde{\mathbf{p}}(k)\right\|\right)^2\right)}{N}} \tag{33}$$

2) RMSE of the measured position relative to the reference trajectory (control):

$$\sqrt{\frac{\sum_{k=1}^{N}\left(\left(\left\|\hat{\mathbf{p}}(k)-\mathbf{p}_d(k)\right\|\right)^2\right)}{N}} \tag{34}$$

3) RMSE of the measured heading relative to reference heading (control):

$$\sqrt{\frac{\sum_{k=1}^{N}\left(\left(\left\|\hat{\psi}(k)-\psi_d(k)\right\|\right)^2\right)}{N}} \tag{35}$$

4) Square root of the total sum of forces applied by each thruster squared (control):

$$\sqrt{\sum_{k=1}^{N}\left(\left(\left\|\mathbf{u}(k)\right\|\right)^2\right)} \tag{36}$$

5) Square root of the total sum of the differences in successive control inputs of each thruster over time squared (control):

$$\sqrt{\sum_{k=1}^{N-1}\left(\left(\left\|\mathbf{u}(k+1)-\mathbf{u}(k)\right\|\right)^2\right)} \tag{37}$$

REFERENCES

[1] L. Chen, H. Hopman, and R. R. Negenborn, "Distributed model predictive control for vessel train formations of cooperative multi-vessel systems," *Transp. Res. Part C Emerg. Technol.*, vol. 92, no. May, pp. 101–118, 2018.

[2] R. R. Murphy, E. Steimle, M. Hall, M. Lindemuth, D. Trejo, S. Hurlebaus, Z. Medina-Cetina, and D. Slocum, "Robot-Assisted Bridge Inspection," *J. Intell. Robot. Syst.*, vol. 64, no. 1, pp. 77–95, 2011.

[3] E. Pinto, P. Santana, F. Marques, R. Mendonça, A. Lourenço, and J. Barata, "On the Design of a Robotic System Composed of an Unmanned Surface Vehicle and a Piggybacked VTOL," in *Technol. Innov. Collect. Aware. Syst.*, L. M. Camarinha-Matos, N. S. Barrento, and R. Mendonça, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 193–200.

[4] L. Chen, H. Hopman, and R. R. Negenborn, "Distributed Model Predictive Control for cooperative floating object transport with multi-vessel systems," *Ocean Eng.*, vol. 191, no. September, 2019.

[5] L. Elkins, D. Sellers, and W. R. Monach, "The Autonomous Maritime Navigation (AMN) project: Field tests, autonomous and cooperative behaviors, data fusion, sensors, and vehicles," *J. F. Robot.*, vol. 27, no. 6, pp. 790–818, 2010.

[6] H. Zhu, J. Juhl, L. Ferranti, and J. Alonso-Mora, "Distributed Multi-Robot Formation Splitting and Merging in Dynamic Environments," *IEEE Int. Conf. Robot. Autom.*, pp. 9080–9086, 2019.

[7] J. Alonso-Mora, E. Montijano, T. Nägeli, O. Hilliges, M. Schwager, and D. Rus, "Distributed multi-robot formation control in dynamic environments," *Auton. Robots*, vol. 43, no. 5, pp. 1079–1100, 2018. [Online]. Available: https://doi.org/10.1007/s10514-018-9783-9

[8] J. Lawton, R. Beard, and B. Young, "A Decentralized Approach to Formation Maneuvers," *IEEE Trans. Robot. Autom.*, vol. 19, no. 6, pp. 933–941, 2003.

[9] G. Antonelli, F. Arrichiello, and S. Chiaverini, "Experiments of formation control with multirobot systems using the null-space-based behavioral control," *IEEE Trans. Control Syst. Technol.*, vol. 17, no. 5, pp. 1173–1182, 2009.

[10] J. Ghommem, F. Mnif, G. Poisson, and N. Derbel, "Nonlinear formation control of a group of underactuated ships," *Ocean. 2007 - Eur. Aberdeen*, pp. 1–8, 2007.

[11] I. Mas and C. A. Kitts, "Dynamic control of mobile multirobot systems: The cluster space formulation," *IEEE Access*, vol. 2, pp. 558–570, 2014.

[12] K. Guo, X. Li, and L. Xie, "Ultra-wideband and Odometry-Based Cooperative Relative Localization With Application to Multi-UAV Formation Control," *IEEE Trans. Cybern.*, pp. 1–14, 2019.

[13] S. K. Pang, Y. H. Li, and H. Yi, "Joint formation control with obstacle avoidance of towfish and multiple autonomous underwater vehicles based on graph theory and the null-space-based method," *Sensors (Switzerland)*, vol. 19, no. 11, 2019.

[14] M. Breivik, V. E. Hovstein, and T. I. Fossen, *Ship formation control: A guided leader-follower approach*. IFAC, 2008, vol. 17, no. 1 PART 1. [Online]. Available: http://dx.doi.org/10.3182/20080706-5-KR-1001.02706

[15] D. Schoerling, C. Van Kleeck, F. Fahimi, C. R. Koch, A. Ams, and P. Löber, "Experimental test of a robust formation controller for marine unmanned surface vessels," *Auton. Robots*, vol. 28, no. 2, pp. 213–230, 2010.

[16] A. Riahifard, S. M. H. Rostami, J. Wang, and H. J. Kim, "Adaptive leader-follower formation control of under-actuated surface vessels with model uncertainties and input constraints," *Appl. Sci.*, vol. 9, no. 18, 2019.

[17] B. Yu, X. Dong, Z. Shi, and Y. Zhong, "Formation control for quadrotor swarm systems: Algorithms and experiments," *Chinese Control Conf. CCC*, pp. 7099–7104, 2013.

[18] Y. Kuwata and J. P. How, "Cooperative distributed robust trajectory optimization using receding horizon MILP," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 2, pp. 423–431, 2011.

[19] R. R. Negenborn and J. M. Maestre, "Distributed model predictive control: An overview and roadmap of future research opportunities," *IEEE Control Syst.*, vol. 34, no. 4, pp. 87–97, 2014.

[20] D. Liu, H. Liu, F. L. Lewis, and Y. Wan, "Robust fault-tolerant formation control for tail-sitters in aggressive flight mode transitions," *IEEE Trans. Ind. Informatics*, vol. 16, no. 1, pp. 299–308, 2020.

[21] J. Alonso-Mora, S. Baker, and D. Rus, "Multi-robot formation control and object transport in dynamic environments via constrained optimization," *Int. J. Rob. Res.*, vol. 36, no. 9, pp. 1000–1021, 2017.

[22] Z. Li, Z. Liu, and J. Zhang, "Multi-under-actuated unmanned surface vessel coordinated path tracking," *Sensors (Switzerland)*, vol. 20, no. 3, 2020.

[23] MIT and AMS, "Roboat," 2019. [Online]. Available: roboat.org

[24] T. I. Fossen, "Handbook of Marine Craft Hydrodynamics and Motion Control," 2011.

[25] W. Wang, L. A. Mateos, S. Park, P. Leoni, B. Gheneti, F. Duarte, C. Ratti, and D. Rus, "Design, Modeling, and Nonlinear Model Predictive Tracking Control of a Novel Autonomous Surface Vehicle," *2018 IEEE Int. Conf. Robot. Autom. (ICRA), Brisbane*, pp. 6189–6196, 2018.

[26] A. Veksler, T. A. Johansen, F. Borrelli, and B. Realfsen, "Dynamic Positioning With Model Predictive Control," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 4, pp. 1340–1353, 2016.

[27] A. Haseltalab and R. R. Negenborn, "Model predictive maneuvering control and energy management for all-electric autonomous ships," *Appl. Energy*, vol. 251, no. March, 2019.

[28] A. Haseltalab, V. Garofano, M. Van Pampus, and R. Negenborn, "Model Predictive Trajectory Tracking Control and Thrust Allocation for Autonomous Vessels," *Proc. 21st IFAC World Congr. (IFAC2020), Berlin, Ger.*, no. July 2020, pp. 14 733–14 739, 2020.

[29] M. Abdelaal, M. Fränzle, and A. Hahn, "NMPC-based Trajectory Tracking and Collison Avoidance of Underactuated Vessels with Elliptical Ship Domain," *IFAC-PapersOnLine*, vol. 49, no. 23, pp. 22–27, 2016. [Online]. Available: http://dx.doi.org/10.1016/j.ifacol.2016.10.316

[30] Stanford Artificial IntelligenceLaboratory, "Robotic Operating System," 2018. [Online]. Available: https://www.ros.org

[31] B. Houska, H. Ferreau, and M. Diehl, "ACADO Toolkit - An Open Source Framework for Automatic Control and Dynamic Optimization," *Optim. Control Appl. Methods*, vol. 32, no. 3, pp. 298–312, 2011.