A Retrofitted Dynamic Window Approach with Pivot Point Control for Maneuvering Inland Vessels on Constrained Surfaces

Dingrui Wang
System Hub
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
dingruiwang@hkust-gz.edu.cn

Yanyun Zhang
Department of Mechanical Engineering
KU Leuven
Leuven, Belgium
yanyun.zhang@kuleuven.be

Peter Slaets
Department of Mechanical Engineering
KU Leuven
Leuven, Belgium
peter.slaets@kuleuven.be

***Abstract*—Inland waterways play an important role in the transportation of goods and passengers. But maneuvering vessels in this type of restricted area can be challenging due to the limited maneuverability and the presence of obstacles. Hence, these areas can be abstracted as constrained surfaces. The dynamic window approach (DWA) is a path planning strategy for real-time navigation in constrained environments that considers dynamic constraints. While the pivot point concept is a useful technique which analyze the spot that vessels rotate around. Concerning these two concepts, this paper proposes a real-time local path planning algorithm named retrofitted dynamic window approach (RDWA) for maneuvering inland vessels on constrained surfaces. The algorithm retrofitted the DWA to satisfy the control behavior constraints stemming from the limited maneuverability. Further, the pivot point technique is leveraged to enable the vessel to pivot around desired points. The algorithm is evaluated by simulations, which demonstrate the approach can pivot the vessel around a number of pre-defined points for more than 90 degrees and form an arc motion trajectory in less than three minutes.**

Keywords—dynamic window approach, pivot point, collision avoidance, confined surfaces, unmanned surface vehicles

I. Introduction

The field of navigation and control of unmanned surface vehicles (USVs) on constrained surfaces such as inland waterways and canals has long been an area of interest for researchers. A lot of progress has been made in researching and testing unmanned surface vehicles (USVs) [1]. Path planning is crucial in the USV navigation system, which highly indicates the intelligence level. Path planning can be categorized into global path planning and local path planning. Global path planning is mainly based on a known map and includes the A* algorithm [2], Dijkstra [3], etc. In contrast to global path planning, local path planning is often computed in real-time with unknown surroundings. For constrained surfaces such as a harbor where the environment prunes to dynamically change, a real-time path planning scheme is needed. The dynamic window approach (DWA) is a well-known local path planning strategy in terms of real-time navigation [4]. There are methods integrated with the fuzzy logic control algorithm and DWA to find an optimal path in complex environments [5, 6]. In [7], a path planning algorithm is proposed for USVs based on the improved DWA which incorporates the concept of obstacle search angle. A work from [8] studies ocean environmental factors like wave and redesigns the kinematic model of a USV and the evaluation function based on DWA. For USVs, the maneuverability constraints incorporate not only dynamic constraints, but also actuator constraints. However, the above studies are all based on velocity space sampling which means actuator constraints are not well considered. To meet the actuator constraints, a study modified the DWA which highlighted the design of candidate control behavior space is proposed in [9]. Nevertheless, this research has not emphasized the complexity of hydrodynamics which is essential for the USV's motion model.

The concept of pivot point has been frequently used in ship operation. It is a technique that allows a vessel to rotate around a certain point, i.e., the pivot point, to avoid collision while making sharp turning. When the water surface is constrained regarding maneuvering, the pivot point can be utilized to make a predictable turn [10]. It is shown in [11] that incorporates the vessel states and the pivot point information can successfully predict future vessel behavior.

Inspired by the above studies, this paper proposes a real-time local path planning algorithm named retrofitted dynamic window approach (RDWA) which satisfies the control behavior constraints. Compared to traditional DWA, the discretization happens in the control space of the actuators, instead of the velocity space. More importantly, the algorithm leveraged the pivot point control technique and manage to allow the vessel to keep pivoting around a preset point for a certain angle. The method of the algorithm includes the aspects of control space discretization, trajectory prediction, and optimal control selection. More specifically, the pivot point control is incorporated into cost functions, which is an essential part of the optimal control selection process. The simulation result shows that the algorithm is able to pivot the vessel around preset points and form an arc motion trajectory which can release more maneuverability for USV on constrained surfaces.

The remainder of the paper is structured as follows. Unmanned surface vehicle's kinematic and dynamic models are given in Section II. The design of the RDWA algorithm integrated with the pivot point control is presented in Section III.

Section IV validates the performance of the proposed method with three simulation scenarios. Section V concludes this paper.

II. Inland Vessel Model

A. The Maneuvering Model

The vessel model used in this study is a scale model named Cogge which was designed after a barge [12]. As illustrated in Fig. 1, The 3-DOF kinematic model of Cogge in planar motion is given as follows,

$$\dot{\boldsymbol{\eta}} = \Re(\psi)\boldsymbol{v} \tag{1}$$

where $\boldsymbol{v} = [u, v, r]^T$ is the velocity vector with u denoting the surge velocity, v denoting the sway velocity and r denoting the yaw rate. Notice that the waterway current is assumed to be irrotational, and thus $\boldsymbol{v}$ is the relative velocity compared to the velocity of the waterway current. $\boldsymbol{\eta} = [x, y, \psi]^T$ is the position vector, and $\Re(\psi)$ is the rotation matrix.

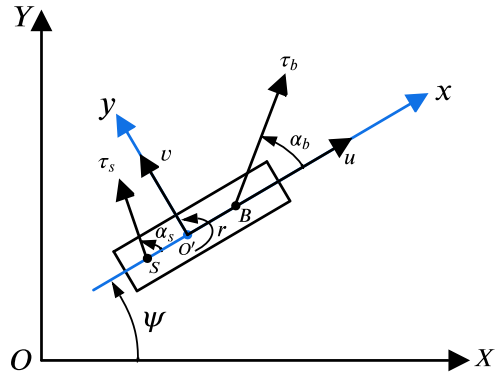

Fig. 1. The 3-DOF kinematic model.

As shown in Fig. 1, the thruster system of Cogge has two thrusters, namely, a four-channel stern thruster located at point S and a steering-grid bow thruster located at point B. The distance from the vessel's reference point O' to B is denoted as X_b, while for S is X_s. To have a clean expression, hereafter a subscript 'b/s' is used to indicate 'bow or stern'. The state of the thruster system is represented by $\boldsymbol{u} = [n_b, \alpha_b, n_s, \alpha_s]^T$, where $n_{b/s}$ and $\alpha_{b/s}$ denote the propeller speed and propulsion angle for bow or stern thrusters. According to [12], the steering mechanism of bow and stern thrusters can both rotate 360°. Moreover, based on the experimental analysis in and [13], the theoretical forces of bow and stern thrusters are computed as follows,

$$\tau_b = 7 \cdot 10^{-6} {n_b}^2 - 0.0076 n_b \sqrt{u^2 + (v + X_B r)^2} \tag{2}$$

$$\tau_s = 2.7 \cdot 10^{-5} {n_s}^2 - 0.028 n_s \sqrt{u^2 + (v + X_S r)^2} \tag{3}$$

With (2) and (3), the thruster system force vector $\boldsymbol{\tau}_{thrust}$ is computed as,

$$\boldsymbol{\tau}_{thrust} = \begin{bmatrix} \tau_b \cos(\alpha_b) + \tau_s \cos(\alpha_s) \\ -\tau_b \sin(\alpha_b) - \tau_s \sin(\alpha_s) \\ -\tau_b X_B \sin(\alpha_b) + \tau_s X_S \sin(\alpha_s) \end{bmatrix} \tag{4}$$

Considering the assumptions made in [13], the 3-DOF dynamic model is given as follows,

$$(\boldsymbol{M}_{RB} + \boldsymbol{M}_A)\dot{\boldsymbol{v}} + \boldsymbol{D}(\boldsymbol{v})\boldsymbol{v} = \boldsymbol{\tau}_{thrust} \tag{5}$$

where $\boldsymbol{M}_A$ is the added mass matrix, and $\boldsymbol{D}(\boldsymbol{v})$ is the hydrodynamic damping matrix. $\boldsymbol{M}_{RB}$ is the rigid-body inertia matrix associated with m, I_{zz}, x_g where m denotes the mass, I_{zz} denotes the moment of inertia, x_g is the center of the gravity. The form of $\boldsymbol{M}_{RB}, \boldsymbol{M}_A, \boldsymbol{D}(\boldsymbol{v})$ are detailed in [14].

Considering (1), (4) and (5), $\mathbf{X} = [\boldsymbol{\eta}, \boldsymbol{v}]^T$ is the state vector, the state update model is derived and briefly represented as a function $\boldsymbol{f}_X$ as shown below,

$$\dot{\mathbf{X}} = \begin{bmatrix} \dot{\boldsymbol{\eta}} \\ \dot{\boldsymbol{v}} \end{bmatrix} = \begin{bmatrix} \Re(\psi)\boldsymbol{v} \\ (\boldsymbol{M}_{RB} + \boldsymbol{M}_A)^{-1}(\boldsymbol{\tau}_{thrust} - \boldsymbol{D}(\boldsymbol{v})\boldsymbol{v}) \end{bmatrix} = \boldsymbol{f}_X(\mathbf{X}, \boldsymbol{u}) \tag{6}$$

III. RDWA Algorithm

For maneuvering vessels on constrained water surfaces, a robust, real-time path planning scheme is required. Integrated with the pivot point control technique, the RDWA is a real-time local path planning algorithm that satisfies the control behavior constraints. A concise illustration of the RDWA is presented in Fig. 2. First, besides restriction in terms of the thruster's structural limitations, the control space is further restricted by the dynamic limitations concerning the current state of the USV. This restricted control space is then discretized which gives control samples. For each sample, a trajectory prediction will be made based on the kinematic and dynamic models. Finally, all the predicted trajectories will be assigned a cost. The control command with the lowest cost is selected and will be passed to the thruster system.

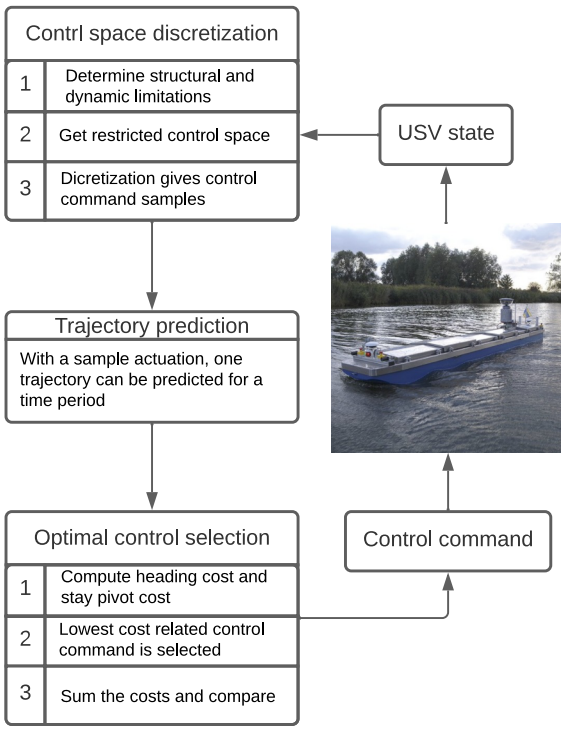

Fig. 2. The flowchart of the RDWA.

A. Control Space Discretization

The control space is restricted concerning the thruster's structural and dynamic limitations. The control domain of the Cogge is limited by its intrinsic structure and this gives the possible control space denoted as V_s. To be noticed, the thrusters are both able to rotate 360 degrees, thus the propulsion angle is not constrained by any structural limitation. The possible control space is shown below,

$$V_s = \left\{ (\mathrm{n_b}, \alpha_\mathrm{b}, \mathrm{n_s}, \alpha_\mathrm{s}) \middle| \begin{matrix} n_\mathrm{b} \in [-n_b^{max}, n_b^{max}], \alpha_\mathrm{b} \in \mathbb{R}, \\ \mathrm{n_s} \in [-n_s^{max}, n_s^{max}], \alpha_\mathrm{s} \in \mathbb{R} \end{matrix} \right\} \quad (7)$$

where $\alpha_{b/s}^{max}$ and $n_{b/s}^{max}$ are the maximum propulsion angles and the maximum propeller speeds for the bow or stern thrusters.

Due to the limitation of the thruster's acceleration and velocity, whether a control command is reachable or not in the near future depends on the current state of the thruster system. This limitation leads to a more restricted control space called the dynamic window control space denoted as V_d and is shown as follows,

$$V_d = \left\{ (\mathrm{n_b}, \alpha_\mathrm{b}, \mathrm{n_s}, \alpha_\mathrm{s}) \middle| \begin{matrix} n_\mathrm{b} \in [n_b^a - \dot{n}_b \cdot \Delta t, n_b^a + \dot{n}_b \cdot \Delta t], \\ \alpha_\mathrm{b} \in [\alpha_b^a - \dot{\alpha}_b \cdot \Delta t, \alpha_b^a + \dot{\alpha}_b \cdot \Delta t], \\ \mathrm{n_s} \in [n_s^a - \dot{n}_s \cdot \Delta t, n_s^a + \dot{n}_s \cdot \Delta t], \\ \alpha_\mathrm{s} \in [\alpha_s^a - \dot{\alpha}_s \cdot \Delta t, \alpha_s^a + \dot{\alpha}_s \cdot \Delta t] \end{matrix} \right\} \quad (8)$$

where $n_{b/s}^a, \alpha_{b/s}^a, \dot{n}_{b/s}, \dot{\alpha}_{b/s}$ are the current propeller speeds, current propulsion angles, maximum propeller accelerations, maximum propulsion angular velocities, respectively for bow or stern thrusters, Δt is the time step.

The intersection of V_s and V_d gives the restricted actuations V_r, as shown below,

$$V_r = V_s \cap V_d \quad (9)$$

When the restricted control space V_r is determined, it can be further discretized. In this process, resolution numbers M, N, O, P are used. For bow thruster, the propeller speed n_b can be evenly discretized into N number of values, and the propulsion angle α_b can be evenly discretized into M number of values. And the stern thruster has the same process with resolution numbers O and P. The discretization process results in a discretized control space V_{rd} which has a number of $M \times N \times O \times P$ control samples of $[n_b, \alpha_b, n_s, \alpha_s]^T$.

B. Trajectory Prediction

Based on the control command samples provided by the discretized control space V_{rd}, the trajectory prediction is made. It is assumed that the control command remains constant during the prediction time duration t_n, which needs to be chosen as follows,

$$t_n = t_0 + n \cdot \Delta t (n \epsilon \mathbb{N}) \quad (10)$$

where t_0 denotes the current time stamp while making the prediction, and n is a natural number to be chosen.

The initial state vector is represented as $\mathbf{X}_{\mathrm{t}_0}$, $\boldsymbol{u}_{sp}$ denotes one control command sample from V_{rd}. According to (6), after one time step Δt, the new state is updated as below,

$$\mathbf{X}_{t_1} = \mathbf{X}_{\mathrm{t}_0} + \boldsymbol{f}_{\boldsymbol{X}}(\mathbf{X}_{\mathrm{t}_0}, \boldsymbol{u}_{sp}) \cdot \Delta t \quad (11)$$

Moreover, the end state of period t_n is computed as below,

$$\mathbf{X}_{\mathrm{t_n}} = \mathbf{X}_{\mathrm{t}_0} + \sum_{i=1}^{n} \boldsymbol{f}_{\boldsymbol{X}}(\mathbf{X}_{\mathrm{t_i}}, \boldsymbol{u}_{sp}) \cdot \Delta t \quad (12)$$

As shown in Fig. 3, regarding the current state $\boldsymbol{\eta}_0$ of USV, with a sample actuation $\boldsymbol{u}_{sp}$, one trajectory can be predicted for a time period t_n. The end position of this trajectory prediction is $\boldsymbol{\eta}_n = [x_n, y_n, \psi_n]^T$ with a global coordinate $P_n = [x_n, y_n]^T$.

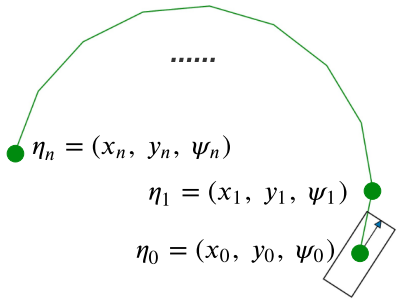

Fig. 3. Trajectory prediction for one actuation sample.

C. Optimal Control Selection

Cost functions are used to evaluate the trajectory predictions. The trajectory which has the lowest cost will be selected as the control command to be sent to the thruster system. For each cost function, there is a corresponding weight factor. Total cost is computed as follows,

$$c_t = w_1 \cdot c_h + w_2 \cdot c_{sp} \quad (13)$$

where c_h and c_{sp} are heading cost and stay pivot cost, w_1 and w_2 are weight factors of the corresponding costs.

1) Heading cost

The purpose of using heading cost is to adjust the vessel's heading to align the vessel with the goal. The goal in the global frame is $P_g = [x_g, y_g]^T$. The goal angle is computed as follows,

$$\psi_g = arctan\left(\frac{x_g - x_n}{y_g - y_n}\right) \quad (14)$$

The heading cost is proportional to the absolute value of the gap between ψ_n and ψ_g, and is computed as follows,

$$c_h = |\psi_g - \psi_n| \quad (15)$$

2) Stay pivot cost

The pivot point is a special point on the centerline of the vessel at which the instantaneous lateral speed is zero, where the centerline is defined as the straight line running from bow to stern of the vessel, midway between the starboard and port sides. As illustrated in Fig. 4, the vessel is pivoting around a point for

different scenarios, i.e., (a) before the bow, (b) on the body, (c) after the stern. The pivot point's location is useful for maneuvering vessels to make sharp turns on constrained surfaces such as a narrow waterway. Moreover, if the pivot point's location can be controlled or preset for navigation, the future motion is therefore also under control. Hence, in this study, instead of computing the location of the pivot point to make a prediction, the pivot point is controlled directly by presetting the desired point for pivoting in the local and global frame.

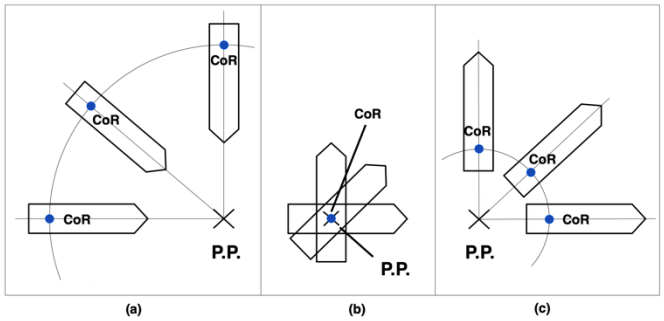

Fig. 4. Vessel pivoting scenarios regarding the pivot point and the coordinate of the vessel's reference point.

First, the concept of pivot points in the local and global frame is distinguished. The local pivot point denoted as PP_L can be seen as a point on the extension of the vessel body. While the global pivot point denoted as PP_G is a stationary point preset in the global coordinate frame. For all scenarios presented in Fig. 4, the PP_L and PP_G have coincided in the global frame. Compared to Fig. 5, the PP_L and PP_G do not coincide.

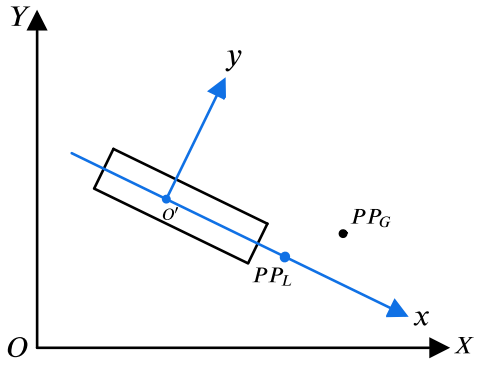

Fig. 5. Local pivot point and the global pivot point are not coincided.

To control the pivot point's position means to keep $PP_L = [x_L, y_L]^T$ overlapping with $PP_G = [x_G, y_G]^T$ in the global frame. $PP_L^G = [x_L^G, y_L^G]^T$ is the transformation result of PP_L from local frame to global frame. To be noticed, the transformation is based on the end position of one trajectory prediction which is denoted before as P_n. The transformation process is computed as below,

$$PP_L^G = \Re(\psi_n) \cdot PP_L + P_n \tag{16}$$

The stay pivot cost c_{sp} is proposed to stimulate the vessel's movement to minimize the distance between PP_G and PP_L^G while the vessel is yawing and is computed as follows,

$$c_{sp} = \sqrt{(y_G - y_L^G)^2 + (x_G - x_L^G)^2} \tag{17}$$

IV. Result and Discussion

The simulations are performed with a 2.4 GHz 8-Core Intel Core i9 CPU and 16 GB RAM, using Python 3.7.6. The USV in this study featured a rectangular rigid body and its rigid body matrix $\boldsymbol{M}_{RB}$ related parameters and other parameters of the algorithm are listed in Table I. The weight factors regarding the cost functions of RDWA are given in Table II. The hydrodynamic parameters of $\boldsymbol{M}_A, \boldsymbol{D}(\boldsymbol{v})$ are estimated by the system identification experiments in [13]. The location of the USV is represented by a black rectangle. The related local pivot point is represented by a red dot. The trajectory of the vessel's reference point is represented by the green curve. The global pivot point, the pivot goal and the position of the vessel's reference point are represented by the black crosses.

TABLE I. Parameters of the Algorithm

Parameters	Values	Units
m	590	kg
x_g	1.25	m
I_{zz}	2074	kgm^2
L (length), B (beam)	4.8, 0.63	m
X_B, X_S	2.61, 0.63	m
n_b^{max}, n_s^{max}	800, 1500	rpm
$\dot{n}_b, \dot{n}_s$	130, 60	rpm/s
$\dot{\alpha}_b, \dot{\alpha}_s$	0.70, 0.42	rad/s
$\Delta t, t_n$	1, 8	s
M, N, O, P, n	3, 3, 3, 3, 7	-

TABLE II. Weight Factors of the RDWA

Weight factor	Related cost	Values
w_1	Heading cost	0.25
w_2	Stay pivot cost	0.75

A. Scenario 1: Pivot Point on the Body

This case concerns pivot point on the body. The initial position of the vessel's reference point and its heading is $\boldsymbol{\eta}_0 = [-3,0,0]^T$ (m, m, rad). The initial speed is $\boldsymbol{v}_0 = [0,0,0]^T$ (m/s, m/s, rad/s). The initial state of the thruster is $\boldsymbol{u}_0 = [0,0,0,0]^T$. The USV is aiming at aligning the body frame with the aligning goal $P_g = [7,5]^T$ while pivoting around the global pivot point $PP_G = [-1,0]^T$. The local pivot point in the body frame is set as $PP_L = [2,0]^T$. As seen from Fig. 6: The stern thruster's propeller speed increased to the maximum to pivot the USV and turned around its propulsion angle to decelerate the vessel before aligned with the pivot goal. The vessel's surge speed was kept near zero all the time. While the yaw rate and sway speed had similar trends, as they respectively increased to near -0.1 rad/s and 0.2 m/s after 20 seconds and ceased to zero after 30 seconds. It can be seen that the USV was pivoting around a point on the body, while its centerline was gradually aligned with the pivot goal. During the motion, the local pivot point is well-coincided with the global pivot point which shows the algorithm can successfully maneuver the vessel to pivot around a point on the body for certain direction smoothly.

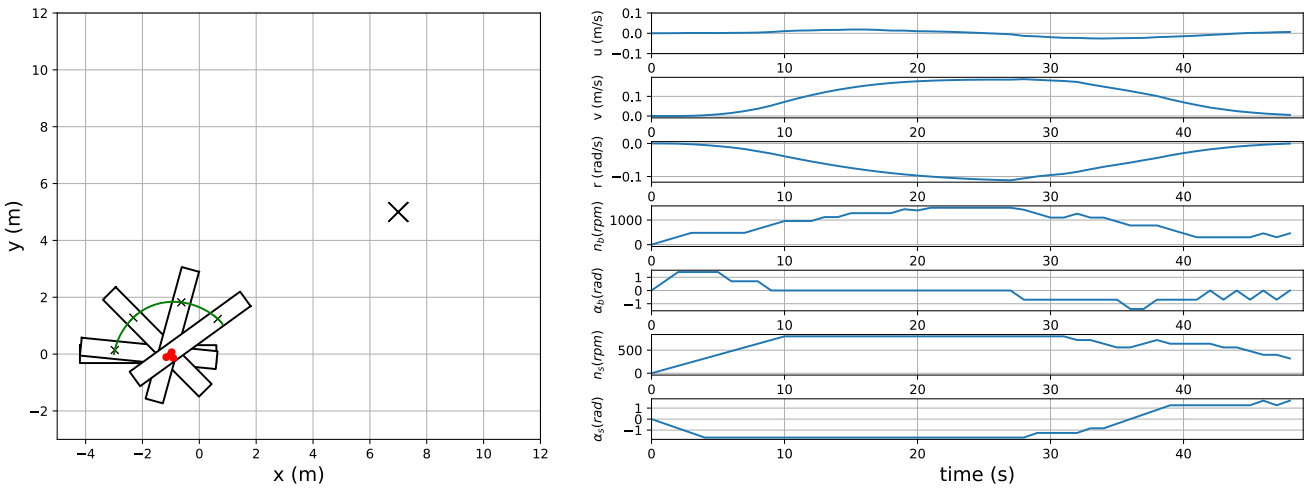

Fig. 6. States and control inputs of USV for pivot point on the body.

B. *Scenario 2: Pivot Point before the Bow*

This simulation concentrated on making the USV pivot around a point before the bow. The initial position regarding the reference point of the vessel and its heading is $\boldsymbol{\eta}_0 = [-3,0,0]^T$ (m, m, rad). Initial speed is $\boldsymbol{v}_0 = [0,0,0]^T$ (m/s, m/s, rad/s). The initial state of the thruster is $\boldsymbol{u}_0 = [0,0,0,0]^T$. The USV is aiming at aligning the body frame with the aligning goal $P_g = [10,10]^T$ while pivoting around the global pivot point $PP_G = [5,0]^T$. The local pivot point in the body frame is set as $PP_L = [8,0]^T$. As seen from Fig. 7: The bow thruster's propeller speed started to increase to the maximum and its propulsion angle was kept around 5 rad to pivot the USV. It turned around its propulsion angle after 140 seconds to decelerate before aligning with the pivot goal. The stern thruster's propeller speed was kept around 500 rpm with a propulsion angle near -2 rad. The vessel's surge speed was kept near zero for the whole maneuvering duration. While the yaw rate and sway speed had similar trends, as they respectively kept around -0.17 rad/s and 0.1 m/s during 20 to 130 seconds and ceased to zero after 150 seconds. It can be seen that the USV was pivoting around a point before the bow. While the USV's centerline was gradually aligned with the pivot goal. During the motion, the local pivot point is well-coincided with the global pivot point. In general, the algorithm can successfully maneuver the vessel to pivot for certain degrees in a smooth way and form an arc trajectory.

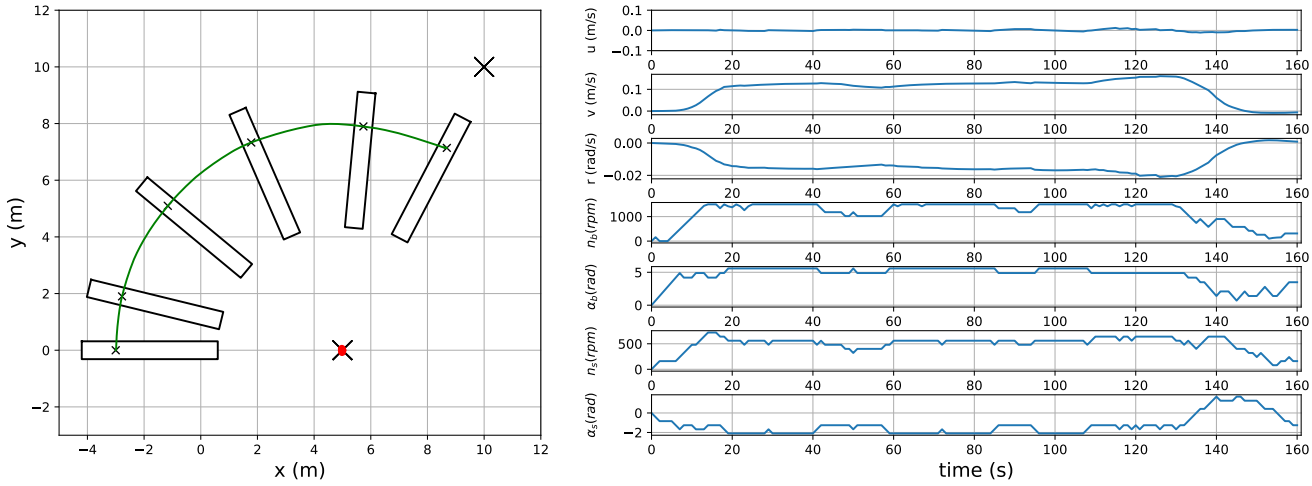

Fig. 7. States and control inputs of USV for pivot point before the bow.

C. *Scenario 3: Pivot Point after the Stern*

In this scenario, the simulation focuses on making the USV pivot around a point before the bow. The initial position regarding the reference point of the vessel and its heading is $\boldsymbol{\eta}_0 = [7,0,0]^T$ (m, m, rad). Initial speed is $\boldsymbol{v}_0 = [0,0,0]^T$ (m/s, m/s, rad/s). The initial state of the thruster is $\boldsymbol{u}_0 = [0,0,0,0]^T$. The USV is aiming at aligning the body frame with the aligning goal $P_g = [-4,10]^T$ while pivoting around the global pivot point $PP_G = [5,0]^T$. The local pivot point in the body frame is set as $PP_L = [-2,0]^T$. As seen from Fig. 8: The bow and stern thruster's propeller speeds were both increased to the maximum after 15 seconds. While the yaw rate and sway speed had a similar trend, as they respectively increased to around 0.07 rad/s and 0.15 m/s during 0 to 40 seconds, started to drop evidently after 45 seconds, and ceased to near zero at 60 seconds. In general, the algorithm can successfully maneuver the vessel to pivot around a point after the stern for certain degrees.

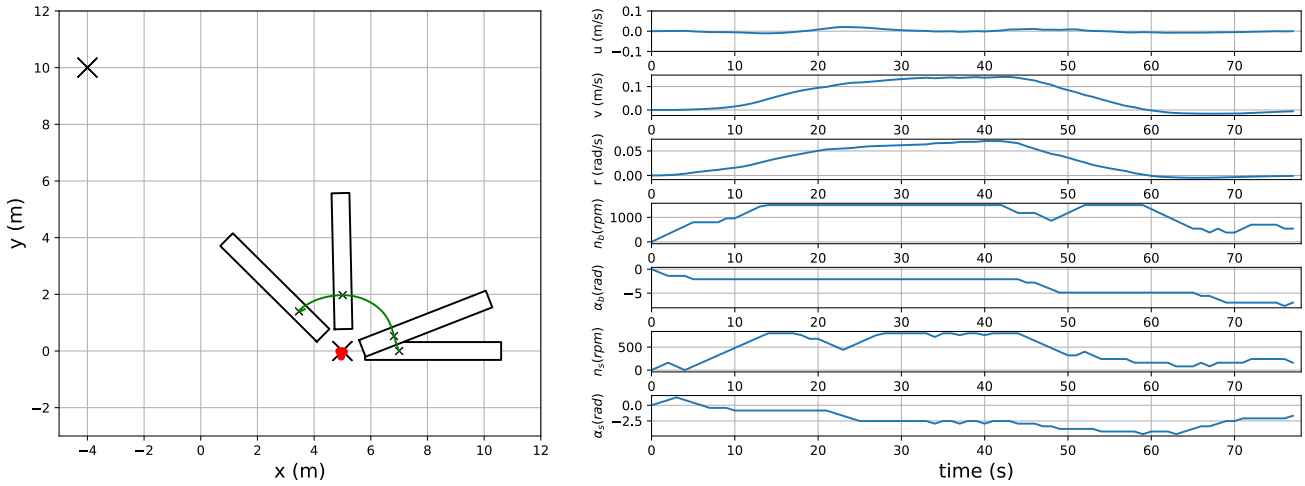

Fig. 8. States and control inputs of USV for pivot point after the stern.

V. Conclusion

A retrofitted dynamic window approach integrated with the pivot point control technique for maneuvering inland vessels in constrained environments is proposed in this paper. A 3-DOF maneuvering model of Cogge is utilized to exert the algorithm. And its control behavior constraints are satisfied by concerning the control space of the actuators instead of the velocity space. The algorithm utilized the pivot point technique and successfully manage to make the vessel keep pivoting around a preset point for a certain angle. As shown in the simulation, the USV integrated with the RDWA is able to pivot around pre-defined points which include points before the bow, on the body and after the stern for more than 90 degrees in less than three minutes. And the motion of the USV forms an arc pattern trajectory. In general, the algorithm gives more maneuverability for USV on constrained surfaces.

Future work can concentrate on exerting experiments in real life since environmental disturbances such as wind and waves can be challenging. It is also interesting to have an automatic selection system of the pivot point based on the surrounding situation of the vessel, which will directly involve the pivot point technique in collision avoidance.

Acknowledgment

I would like to express my special thanks to Dr. Paolo Pilozzi, Robrecht Louw, and Yuyu Cai. The second author Yanyun Zhang is a researcher under European Training and Research Network on Autonomous Barges for Smart Inland Shipping, funded by European Union's EU Framework Program for Research and Innovation Horizon 2020 under Grant Agreement No 955.768. The scale model was funded by the EFRO-Flanders project "Autonoom Varen in de Westhoek".

Reference

[1] W. Deling, W. Dongkui, C. Huang, and W. Changyue, "Marine Autonomous Surface Ship - A Great Challenge to Maritime Education and Training," *American Journal of Water Science and Engineering,* vol. 6, pp. 10, 01/01, 2020.

[2] C. Liu, Q. Mao, X. Chu, and S. Xie, "An Improved A-Star Algorithm Considering Water Current, Traffic Separation and Berthing for Vessel Path Planning," *Applied Sciences,* 9, 2019].

[3] M. Luo, X. Hou, and J. Yang, "Surface Optimal Path Planning Using an Extended Dijkstra Algorithm," *IEEE Access,* vol. 8, pp. 147827-147838, 2020.

[4] D. Fox, W. Burgard, and S. Thrun, "The Dynamic Window Approach to Collision Avoidance," *Robotics & Automation Magazine, IEEE,* vol. 4, pp. 23-33, 04/01, 1997.

[5] L. Zhang, Y. Han, and B. Jiang, "Research on Path Planning Method of Unmanned Boat Based on Improved DWA Algorithm," *Journal of Sensors,* vol. 2022, pp. 1-10, 09/23, 2022.

[6] D. Gao, P. Zhou, W. Shi, T. Wang, and Y. Wang, "A Dynamic Obstacle Avoidance Method for Unmanned Surface Vehicle under the International Regulations for Preventing Collisions at Sea," *Journal of Marine Science and Engineering,* 10, 2022].

[7] Z. Tan, N. Wei, and Z. Liu, "Local Path Planning for Unmanned Surface Vehicle based on the Improved DWA Algorithm." pp. 3820-3825.

[8] Z. Wang, Y. Liang, C. Gong, Y. Zhou, C. Zeng, and S. Zhu, "Improved Dynamic Window Approach for Unmanned Surface Vehicles’ Local Path Planning Considering the Impact of Environmental Factors," *Sensors,* 22, 2022].

[9] X. Sun, G. Wang, Y. Fan, and D. Mu, "Collision avoidance control for unmanned surface vehicle with COLREGs compliance," *Ocean Engineering,* vol. 267, pp. 113263, 2023/01/01/, 2023.

[10] S.-G. Seo, "Safer and More Efficient Ship Handling with the Pivot Point Concept," *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation,* vol. 10, pp. 605-612, 01/01, 2016.

[11] L. P. Perera, "Ship Maneuvering Prediction under Navigation Vector Multiplication based Pivot Point Estimation," *IFAC-PapersOnLine,* vol. 48, no. 16, pp. 1-6, 2015/01/01/, 2015.

[12] G. Peeters, M. Kotzé, M. R. Afzal, T. Catoor, S. Baelen, P. Geenen, M. Vanierschot, R. Boonen, and P. Slaets, "An unmanned inland cargo vessel: Design, build, and experiments," *Ocean Engineering,* vol. 201, pp. 107056, 04/01, 2020.

[13] G. Peeters, S. Van Baelen, G. Yayla, T. Catoor, M. R. Afzal, C. Christofakis, R. Louw, Y. Singh, M. Vanierschot, R. Boonen, and P. Slaets, "Decoupled Hydrodynamic Models and Their Outdoor Identification for an Unmanned Inland Cargo Vessel with Embedded Fully Rotatable Thrusters," *Journal of Marine Science and Engineering,* 8, 2020].

[14] T. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*, p.^pp. 135-137, 2011.