
Autonomous docking and navigation using Voronoi diagrams and model predictive control

Autonomous shipping is an emerging field where it will be important to operate a ship without manual intervention.

MICHAEL LUNDH
Corporate Research
ABB
Sweden

MAO-WEI NILSSON
Electrification Products
ABB
Sweden

JONAS LINDER
Corporate Research
ABB
Sweden

KALEVI TERVO
Industrial Automation
ABB Marine & Ports
Finland

Although there are many issues yet to be solved, not the least the legal ones, it is interesting to investigate functions that already now would be possible to use in today's ship operation. One such field is autonomous navigation in narrow areas. This paper presents a study where a ship is docked autonomously while using information only from the final docking position and the harbor geometry. The Guidance, Navigation and Control system (GNC) outlined here is based on a combination of Voronoi diagrams for the waypoint generation and an integrating model predictive controller (MPC) for the path following between waypoints. Simulations demonstrate how the proposed procedure is able to autonomously dock a cruise ship in the South Harbor of Helsinki.

Introduction

The shipping business is subject to major changes in the future. Safety, costs, energy efficiency, and environmental footprint are some of the factors that drive the change. There is also a tendency to decrease the crew size on board due to cost and this type of labor is predicted to be less attractive in the future. With less people and an increased demand for safety and efficiency there is a demand in many situations to use autonomous functions for consistent and predictable behavior. One such situation is when the ship is approaching a harbor where it is going to dock.

A Guidance, Navigation and Control system (GNC) for autonomous or for automated operation could be used for this. Such systems are commonly applied for motion control of vehicles, spacecraft, aircraft, auto-mobiles and underwater vehicles, see [1], [2], [3], and [4]. Examples of such systems could be found already in the 1920s when the heading of a US navy battleship was automatically steered [5].

The guidance system aims to provide a path or trajectory which fulfills some specific requirements, such as minimum time or fuel optimization. There are various strategies for the guidance system, and the most common strategies are target tracking, trajectory tracking and path following [1].

Some traditional approaches for path planning are cell decomposition [6], the roadmap method [7], and potential fields [8]. Voronoi diagram is studied by Bhattacharya et al. [7] for finding a path for vessel navigation along the South American coastline, and the diagram is generated by a set of points which are the edges of obstacles. The advantage of the Voronoi diagram method is that the maximum clearance path can be generated. However, the path is not necessarily the shortest. The potential field method presents potential from obstacles, and high potential will be provided when a vehicle is close to an obstacle.

Once the path is defined in a list of waypoints it is the task of the control system to maneuver the ship along the desired path and finally reach the desired position. There are several approaches for this. Model Predictive Control (MPC) is one approach which has been proposed in [9]. Other methods, such as back-stepping, has also been introduced to control an autonomous ship, see [10] and [11].

A switching control strategy was proposed in [12] where switches between a linear feedforward-feedback strategy and an MPC can bring the vehicle into a desired parking spot. Instead of using waypoint tracking, a path following method was investigated in [13] in order to park a car. However, because the path was predefined, only one case, reverse parking, was studied.

Waypoint tracking is another method for navigation which is popular for autonomous vehicle, see [14], [15] and [16].

Few studies are reported for docking of ships using MPC. Docking procedures for other crafts such as spacecrafts can be found in [17]. However, path following during docking of marine vessels have been reported in [18] and [19].

A method for autonomous operation of a marine vessel is proposed here. In particular, it is a procedure for the automatic docking of a vessel in a harbor where only the final docking position (including the heading) and the geometry of the harbor are known. The method combines the use of Voronoi diagrams for generating a desired path from a set of waypoints and the use of MPC with integral action for following the desired path. The integral action is needed to handle slowly varying disturbances as wind and current. The proposed method for autonomous docking is illustrated by a simulation in the South Harbor of Helsinki where a simulated ship is navigated through a narrow passage and docked. Further details in this study can be found in [20].

Ship modeling

Ship modeling is extensively described in [1] and details of the modeling are referred to this book.

For the control purposes in this paper it is sufficient with a three degree of freedom (3-DOF)

model that describes the motions in surge, sway and yaw. The generalized velocity in the body-fixed coordinate system is defined as $\nu = [u \ v \ r]^T$ and the generalized position in an Earth-fixed coordinate is defined as $\eta = [X \ Y \ \psi]^T$. The generalized position and velocity is related by the kinematic model

$$\dot{\eta} = R(\psi)\nu \quad (1)$$

where the rotation matrix is given by

$$R(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The ship is assumed to be a rigid body and the kinetics model describing the motion induced by forces is given by

$$M_{RB}\dot{\nu} + C_{RB}(\nu)\nu = \tau_{RB} \quad (3)$$

where M_{RB} is the rigid-body inertia matrix, $C_{RB}(\nu)\nu$ represents centripetal and Coriolis force and τ_{RB} represents the generalized force acting on the rigid body. For 3-DOF motion, this forces is assumed to be given by

$$\tau_{RB} = \tau_{hyd} + \tau_{env} + \tau. \quad (4)$$

where the terms in the right hand side are the generalized forces induced by hydrodynamics, environmental disturbances, and the actuators, respectively.

The hydrodynamics forces are assumed to be given by

$$\tau_{hyd} = -M_A\dot{\nu} - C_A(\nu)\nu - D(\nu)\nu \quad (5)$$

where M_A is the added mass/inertia matrix, $C_A(\nu)\nu$ represents centripetal and Coriolis force due to added mass/inertia and the term $D(\nu)\nu$ represents hydrodynamic damping.

The ship under consideration is propelled with a combination of Azipods, [21], [22], and tunnel thrusters which gives good maneuverability. The dynamics of the thrusters can be taken into consideration to get better performance. A complete model of the thrusters was deemed too complicated and a simplified model was chosen in this

work. A suitable model was proposed in [1] and is given by the saturated first order dynamics

$$T_{thr}\dot{\tau} + \tau = \text{SAT}(\tau_{low}, \tau_c, \tau_{high}) \quad (6)$$

where $T_{thr} = \text{diag}(T_{surge}, T_{sway}, T_{yaw})$, τ_{low} and τ_{high} are upper and lower bounds in the different directions, respectively, and τ_c is the command from the control algorithm. The saturation function $\text{SAT}(\cdot)$ is acting on each element of the vector individually and each row is given by

$$\text{sat}(l, x, h) = \begin{cases} l, & \text{if } x < l \\ x, & \text{if } l \leq x \leq h \\ h, & \text{if } x > h \end{cases} \quad (7)$$

Inserting (4) and (5) into (3) and combining it with (2) and (6) give the complete model

$$\dot{\eta} = R(\psi)\nu \quad (8a)$$

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu = \tau + \tau_{env} \quad (8b)$$

$$T_{thr}\dot{\tau} + \tau = \text{SAT}(\tau_{low}, \tau_c, \tau_{high}) \quad (8c)$$

where $M = M_{RB} + M_A$ and $C = C_{RB}(\nu) + C_A(\nu)$. A linear MPC was considered sufficient to solve the control problem. To provide a linear model of the vessel, (8) is linearized about the current position, zero velocity and zero force. This results in the linear state-space model

$$\begin{bmatrix} \dot{\eta}_p \\ \dot{\nu} \\ \dot{\tau} \end{bmatrix} = \begin{bmatrix} 0 & I & 0 \\ 0 & -M^{-1}D & M^{-1} \\ 0 & 0 & -T_{thr}^{-1} \end{bmatrix} \begin{bmatrix} \eta_p \\ \nu \\ \tau \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ T_{thr}^{-1} \end{bmatrix} \tau_c + \tau_{env} \quad (9)$$

where $\eta_p = [X_p, Y_p, \psi_p]^T$ is the position vector in the linearized model.

Ship control system

The overall system for guidance, navigation, and control of the ship consist of three main components. There is a guidance system for waypoint generation and a control system for trajectory following between waypoints. There is also a navigation system that provides wave filtered observations of the ship positions and velocities transformed to the appropriate coordinate system. The structure is shown in Figure 1.

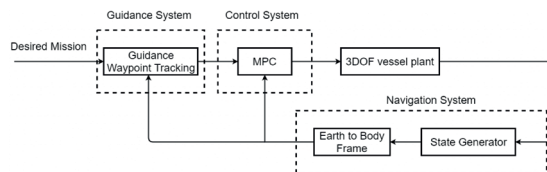


Figure 1: Structure of the simulated guidance, navigation and control system

Voronoi diagrams are used for waypoint generation in the guidance system and the control system for waypoint following is based on Model Predictive Control (MPC).

Guidance System

The potential complexity of the auto-docking problem for a general harbor geometry is a difficult problem to solve for an MPC if only the starting position and the desired final position were supplied. To simplify the task for the MPC and make it more robust to complex geometries, the guidance system will instead generate a list of waypoints which the ship could follow to safely reach its docking position.

One of the main concerns with autonomous operation is how to avoid obstacles. These obstacles can, for instance, be islands and other vessels.

Obstacles are typically classified as static or dynamic obstacles [23]. Static obstacles can be predicted before the path is planned, while dynamic obstacles will appear during the motion. It is assumed to have knowledge of the environment and thus treat the obstacles and other ships as static obstacles.

Here, obstacles will be treated as boundaries, both for the Voronoi diagram generation and later as constraints in the MPC. In most cases, boundaries are not symmetrical and the narrowness of the path is changing. To present the obstacles with one strategy, the path is divided into several parts where each segment has their own set of geometrical constraints.

The path and segments are constructed using a Voronoi diagram which is constructed from a set of predefined points (p_{ix}, p_{iy}) in a plane. The predefined points are given by the geometry of the harbor. First, a set of cells are defined. Every point in a certain cell is closer to the predefined point in this cell than to any other predefined point, i.e. a point (q_x, q_y) in the i^{th} cell if and only if this point satisfies

$$(p_{ix} - q_x)^2 + (p_{iy} - q_y)^2 < (p_{jx} - q_x)^2 + (p_{jy} - q_y)^2$$

for all other predefined points j . These cells form the Voronoi diagram which is defined by edges and

vertices. Since each cell presents the closest area to the predefined point, the points on edges are as far away from the predefined points as possible.

The number of predefined points under consideration will affect the number of vertices. A large number of predefined points will give a large number of vertices. This can be seen in Figure 2 which shows a Voronoi diagram for the Helsinki Harbor with a rather tight grid for the Helsinki Harbor. The effect of what happens when the number of predefined points is decreased is seen in Figure 3, where only a subset of the predefined points are used. These predefined points are the corners of the harbor (marked as x_1 to x_6 in Figure 3) and were regarded as the most interesting with respect to the narrow passage and the final docking position. From these, the vertices (marked as v_1 to v_5) were obtained. Note that one vertex is not marked in Figure 3 because it is located far outside the map. Waypoints for navigation will then be defined on a set of edges in the diagram to the desired

docking position, which is located between the predefined points x_{20} and x_{21} .

The chosen subset of vertices are then used to define waypoints that has a maximum distance to the bounds [24]. The waypoints are chosen as the middle points of two vertices in the Voronoi diagram, i.e. a waypoint is given by

$$P = (v_i + v_{i+1})/2 \quad (10)$$

where it is the middle point for vertices i and $i+1$. The final docking destination is added as the last waypoint (wp4).

It is was chosen to use a Voronoi diagram with rather few points to get a limited number of waypoints. If there were many waypoints due to many predefined points, then it might be necessary to perform some kind of smoothing when the list of waypoints are used to generate the future reference for the MPC.

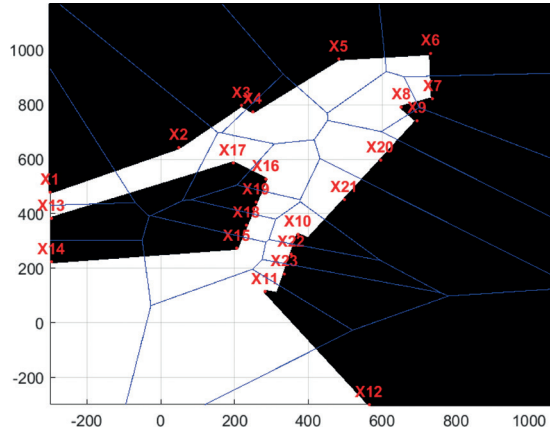


Figure 2: Voronoi diagram for the Harbor of Helsinki with tight grid

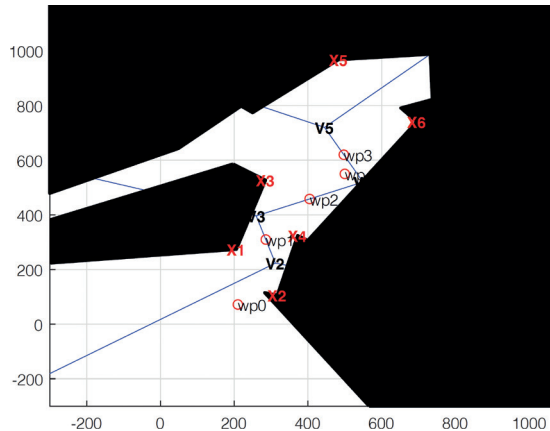


Figure 3: Voronoi diagram based on few predefined points

For each waypoint in the list, a desired yaw angle is also provided to allow the ship to navigate through narrow passages on its way to the final docking position. The yaw angle at a certain waypoint is determined from the average angle of the path between the preceding and the succeeding waypoint. The yaw angle for the docking position is given by the angle of the shore.

The guidance system keeps track of the current waypoint and once the ship approaches it, the GNC will start to shift the MPC's reference towards the next waypoint. The switch occurs when the following criteria are satisfied:

$$\begin{aligned} \sqrt{(X - X_{di})^2 + (Y - Y_{di})^2} &< \varepsilon_{1i} \\ |\psi - \psi_{di}| &< \varepsilon_{2i} \end{aligned} \quad (11)$$

where X , Y and ψ are surge sway and yaw in earth frame, X_d , Y_d and ψ_d are desired surge, sway and yaw. Further, ε_{1i} and ε_{2i} are the allowed deviation for i th desired position. The procedure is outlined in Figure 4.

To give a smooth change in the reference to the MPC in the transition from one way point to the next, a timebased linear interpolation is used. This gives a gradually increasing position error which avoid pulses in the desired control actions.

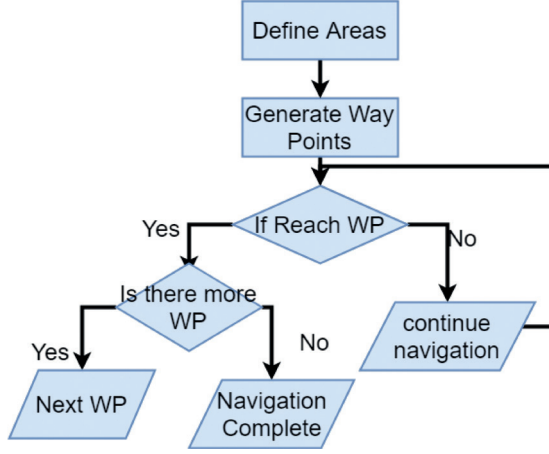


Figure 4: Guidance system function

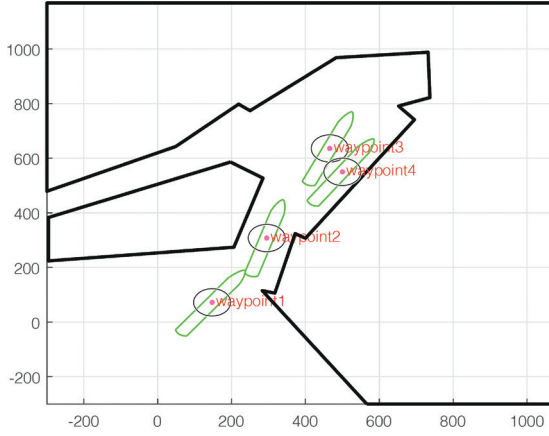


Figure 5: Waypoints for the automatic docking

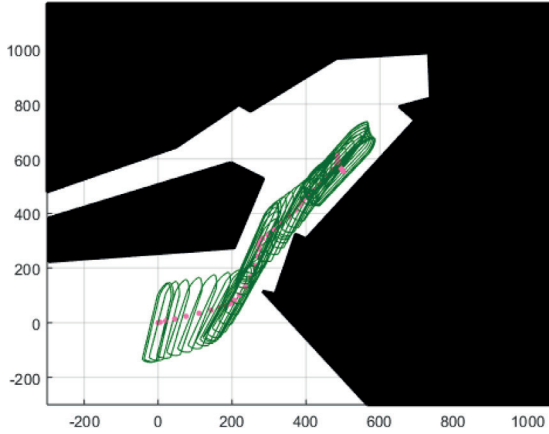


Figure 6: Positions and headings during the automatic docking

When (11) is satisfied, the new reference to the MPC is given by interpolating between the current and the next waypoint, i.e.

$$\begin{bmatrix} r_x \\ r_y \\ r_\psi \end{bmatrix} = \alpha(t) \begin{bmatrix} x_{wp(i)} \\ y_{wp(i)} \\ \psi_{wp(i)} \end{bmatrix} + (1 - \alpha(t)) \begin{bmatrix} x_{wp(i+1)} \\ y_{wp(i+1)} \\ \psi_{wp(i+1)} \end{bmatrix} \quad (12)$$

where r_x , r_y and r_ψ are the position and heading references for the MPC, $x_{wp(\cdot)}$, $y_{wp(\cdot)}$ and $\psi_{wp(\cdot)}$ are the waypoint positions and heading. The weighing factor is given by

$$\alpha(t) = \begin{cases} 1 & \text{if } t > t_b + t_0 \\ \frac{t - t_0}{t_b} & \text{if } t \leq t_b + t_0 \end{cases} \quad (13)$$

where t_0 is the time that (11) is satisfied and t_b is the interpolation interval chosen by the user.

Note that the reference generation has intentionally been kept simple to focus on the concept of finding waypoint by using Voroni diagrams. This concept could be extended with an improved reference generation to get a smoother response from the system.

Control using MPC

The ship control system takes the vessel to the reference that is provided by the guidance system. MPC was chosen to explicitly handle constraints. Constraints are here mainly coming from limitations given by the harbor geometry and by obstacles. There are also limits given by the thrusters. All variables in the MPC are expressed in the linearized coordinate system.

It is assumed that the navigation system has the ability to filter out the influence of waves affecting the vessel and oscillating disturbances are thus neglected in the controller. To handle slowly varying environmental disturbances, e.g. current and wind, it is necessary to include integral action in the MPC. This also improves how the MPC is able to handle the mismatch between the model and the real world ship [25]. Integral action is introduced by using a model with the extended state vector

$$\bar{x}(k) = [\Delta x^T(k) \quad x^T(k)]^T \quad (14)$$

where $\Delta x(k)$ are increments to the original state vector

$$x = [\eta_p^T \quad \nu^T \quad \tau^T]^T \quad (15)$$

The extended discrete-time model is given by

$$\bar{x}(k+1) = \bar{A}\bar{x}(k) + \bar{B}_u\Delta u(k) \quad (16a)$$

$$y(k) = \bar{C}\bar{x}(k) \quad (16b)$$

where $\mathbf{y}(k)$ are the outputs, $\Delta \mathbf{u}(k)$ are the control signal increments, and

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{C}\mathbf{A} & \mathbf{I} \end{bmatrix}, \bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ \mathbf{C}\mathbf{B} \end{bmatrix}, \bar{\mathbf{C}} = [\mathbf{0} \quad \mathbf{I}] \quad (17)$$

where \mathbf{A} , \mathbf{B} , and \mathbf{C} are matrices in the discrete time representation of (9). Further, $\mathbf{C} = \mathbf{I}$ since all original states are assumed to be measurable.

By defining the error $\mathbf{e}(k) = \bar{\mathbf{C}}\bar{\mathbf{x}}(k) - \mathbf{r}(k)$ between the actual outputs and the desired set-point reference, the optimization in the MPC is

$$\begin{aligned} \min_{\Delta \mathbf{u}(k)} & \sum_{k=1}^{N-1} \|\mathbf{e}(k)\|_{\bar{\mathbf{Q}}_1}^2 + \|\Delta \mathbf{u}(k-1)\|_{\bar{\mathbf{Q}}_2}^2 + \|\mathbf{e}(N)\|_{\bar{\mathbf{Q}}_f}^2 \\ & \bar{\mathbf{x}}(k+1) = \bar{\mathbf{A}}\bar{\mathbf{x}}(k) + \bar{\mathbf{B}}\Delta \mathbf{u}(k) \\ & \mathbf{u}(k+1) = \mathbf{u}(k) + \Delta \mathbf{u}(k) \\ \text{s.t. } & \mathbf{y}_{\min}(k) \leq \bar{\mathbf{C}}\bar{\mathbf{x}}(k) \leq \mathbf{y}_{\max}(k) \quad k = 0, \dots, N \\ & \mathbf{u}_{\min} \leq \mathbf{u}(k) \leq \mathbf{u}_{\max} \\ & \Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u}(k) \leq \Delta \mathbf{u}_{\max} \end{aligned} \quad (18)$$

where $\|z\|_{\bar{\mathbf{Q}}}^2 = z^T \bar{\mathbf{Q}} z$. It penalizes deviations from setpoints via $\bar{\mathbf{Q}}_1$, and increments in the control variables via $\bar{\mathbf{Q}}_2$, i.e. the changed commanded forces and torque. It is also possible to have a penalty for the deviation of the final point in the horizon via $\bar{\mathbf{Q}}_f$. Note also that the limits $\mathbf{y}_{\min}(k)$ and $\mathbf{y}_{\max}(k)$ may vary over the prediction horizon. The MPC operates in the linearized coordinate system that at each sampling instant is aligned with the body-fixed system. Position measurements from the navigation system are in an Earth-fixed frame and velocity measurements are provided in a body-fixed frame. Hence, it is needed that references, geometrical limits of the harbor, and velocities are transformed into the current linearized coordinate system.

Positions are transformed into the linearized system using

$$\mathbf{z}_p = \mathbf{R}^T(\psi)(\mathbf{z} - \mathbf{o}_p) \quad (19)$$

where ψ is the yaw angle, \mathbf{z}_p is the position of the vessel in the linearized frame, \mathbf{z} is the position of the vessel in the Earth frame, and \mathbf{o}_p is the position of the origin for the linearized frame in the Earth-fixed frame.

The initial state vector for the MPC in the linearized coordinate system is obtained in the following way:

$$\begin{aligned} \boldsymbol{\eta}_p(k) &= [0 \ 0 \ 0]^T \\ \Delta \boldsymbol{\eta}_p(k) &= \mathbf{R}^T(\psi(k))(\boldsymbol{\eta}(k) - \boldsymbol{\eta}(k-1)) \\ \Delta \boldsymbol{\nu}(k) &= \boldsymbol{\nu}(k) - \mathbf{R}^T(\psi(k))\mathbf{R}(\psi(k-1))\boldsymbol{\nu}(k-1) \end{aligned} \quad (20)$$

The states in the Earth frame needs to be stored from one sample to the next to be able to create the differential states. Note that linearized frame might have been rotated since previous sample, hence, the velocity $\boldsymbol{\nu}(k-1)$ has to be transformed to the current frame.

Results

The proposed method for automatic docking based on a combination of Voronoi diagrams and MPC are here demonstrated for a simulated ship.

The simulations were done in Matlab where the controller was implemented using the YALMIP toolbox [26]. A quadratic programming solver was used to solve the MPC problem. Constraints were expressed as soft constraints.

A 294 m long and 37.9 m wide cruising ship was used in the simulations. It is described by the 3DOF ship model in (8) where

$$\begin{aligned} \mathbf{D}(\boldsymbol{\nu}) = \mathbf{D} &= \text{diag}(4.32 \cdot 10^4, 2.67 \cdot 10^4, 2.2 \cdot 10^{10}) \\ \mathbf{M} &= \text{diag}(4.62 \cdot 10^7, 6.77 \cdot 10^7, 2.54 \cdot 10^{11}) \end{aligned}$$

The simplified thruster model (6) is used with the same $T_{\text{thr}} = 10$ s for all actuators. This simple model is considered to be sufficient to demonstrate the auto-docking concept.

A discrete-time linearized model (16) was used in the MPC. The weights for the MPC problem in the loss function (18) are chosen as

$$\begin{aligned} \mathbf{Q}_f &= \text{diag}([10 \ 10 \ 10^9 \ 5 \cdot 10^4 \ 5 \cdot 10^4 \ 0 \ 10^4 \ 10^4 \ 0]) \\ \mathbf{Q}_2 &= \text{diag}([1 \ 1 \ 10^7]) \\ \mathbf{Q}_1 &= 0.1\mathbf{Q}_f \end{aligned}$$

where the weights \mathbf{Q}_f and \mathbf{Q}_1 penalize deviations in surge, sway, yaw, surge speed, sway speed, yaw speed, surge force, sway force, and yaw torque. The weight \mathbf{Q}_2 penalizes changes in surge force, sway force, and yaw torque. The weights for the final point are chosen much larger than for the points for the transient behavior because the main objective is to reach the final point.

The sampling interval is 10 s and the prediction and the control horizons are both chosen to $N = 15$. The bounds for thruster forces $\boldsymbol{\tau}$ are chosen to

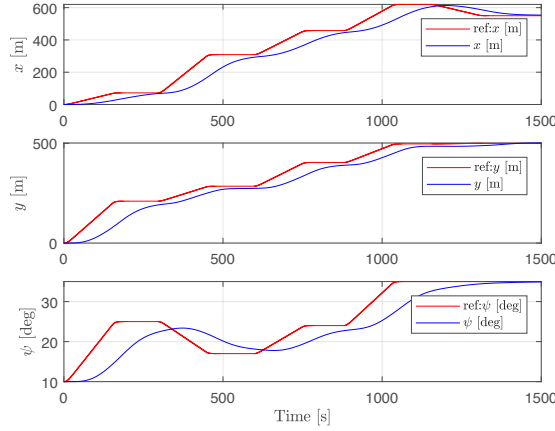


Figure 7: Positions in surge, sway and yaw in earth fixed system

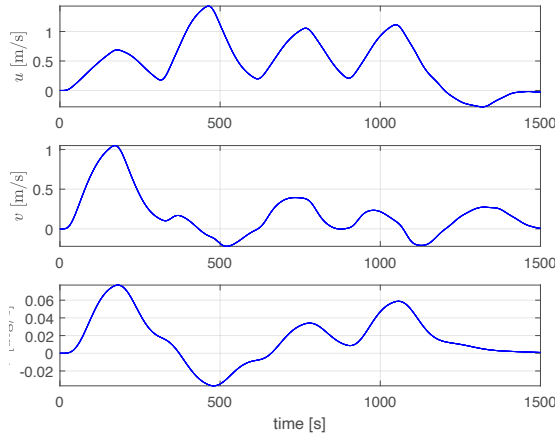


Figure 8: Velocities in surge, sway and yaw in body fixed system

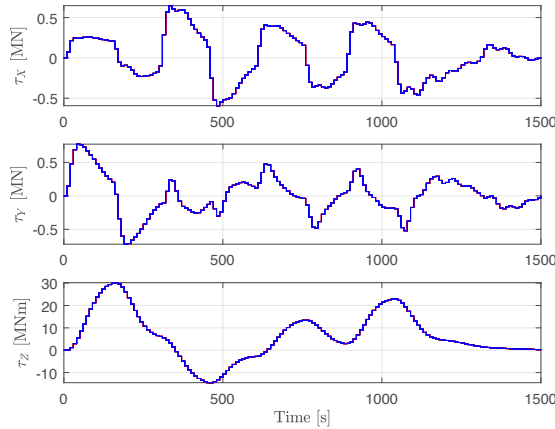


Figure 9: Forces and torque in body fixed system

be ± 4 MN, ± 0.9 MN, and ± 50 MNm, respectively. Further, for all waypoints, the deviations in (11) are $\epsilon_{1i} = 20$ m and $\epsilon_{2i} = 10$ degrees, and the interval in (13) is $t_b = 150$ s.

From the Voronoi diagram in Figure 3 waypoints are created. The defined positions and headings are seen in Figure 5.

The whole automatic docking procedure is illustrated in Figure 6 where the ships position is drawn for selected samples during the docking. Here, it can be seen that the docking procedure manages to bring the vessel to dock without having any side of vessel hit any of the bounds from the harbor.

The positions of the ship in the Earth fixed system during the docking procedure are shown in Figure 7. The corresponding velocities in the body fixed system are shown in Figure 8 and the corresponding commanded forces and torques are shown in Figure 9. In Figure 7 it can be seen that the ship's position is changed smoothly towards its docking position. It can also be noticed that the settling time for heading (yaw), ψ , is long, although it reaches the reference value at the end of the simulation. In total it takes around 1200 s for the vessel to reach its docking position with a sufficiently small error in the heading.

The main purpose of this study was to use Voronoi diagrams for the waypoint generation and use these in an MPC. Although the algorithm does its job fairly well there is room for improvements in how the transition from one waypoint to another is handled. A main reason for that is that the MPC is not using a trajectory of future references. To handle the transition as good as possible, the reference smoothing in (12) was introduced. Even with this smoothing it can be seen in Figure 8 that the surge speed of the ship decreases when the ship approaches a waypoint. The commanded forces and torques are seen in Figure 7. They show that the ship is gently docked with no excessive commands.

Conclusions

The main contribution of this paper is to propose a GNC system for autonomous docking and navigating of ships which combines Voronoi diagrams with model predictive control. A Voronoi diagram is used to generate a list of waypoints to be followed during the docking. An MPC is applied to control the vessel to follow set-points generated from the list of waypoints. The proposed ideas were demonstrated for a simulated autonomous docking in the South Harbor of Helsinki.

Future work will improve the set-point path generation for the MPC to avoid the decreased speed when intermediate waypoints are approached. Further, it is also of interest to do studies on minimizing the time or the needed fuel for docking. Another feature to include would be to dynamically regenerate the list of waypoints to avoid moving objects.

References:

- [1] Thor I Fossen. Handbook of marine craft hydrodynamics and motion control. John Wiley & Sons, 2011.
- [2] Bradley A Steinfeldt, Michael J Grant, Daniel A Matz, Robert D Braun, and Gregg H Barton. Guidance, navigation, and control system performance trades for mars pinpoint landing. *Journal of Spacecraft and Rockets*, 47(1):188–198, 2010.
- [3] Miguel San Martin, Gavin F Mendeck, Paul B Brugarolas, Gurkirpal Singh, Frederick Serricchio, Steven W Lee, Edward C Wong, and John C Essmiller. In-flight experience of the mars science laboratory guidance, navigation, and control system for entry, descent, and landing. *CEAS Space Journal*, 7(2):119–142, 2015.
- [4] Rui Huang, Yong Liu, and J Jim Zhu. Guidance, navigation, and control system design for tripropeller vertical-take-off-and-landing unmanned air vehicle. *Journal of aircraft*, 46(6):1837–1856, 2009.
- [5] Nicolas Minorsky. Directional stability of automatically steered bodies. *Journal of ASNE*, 42(2):280–309, 1922.
- [6] Marius Kloetzer, Cristian Mahulea, and Ramon Gonzalez. Optimizing cell decomposition path planning for mobile robots using different metrics. In *System Theory, Control and Computing (ICSTCC)*, 2015 19th International Conference on, pages 565–570. IEEE, 2015.
- [7] Priyadarshi Bhattacharya and Marina L Gavrilova. Roadmap-based path planning-using the voronoi diagram for a clearance-based shortest path. *IEEE Robotics & Automation Magazine*, 15(2), 2008.
- [8] Barisic, M. Guidance Of Formations Of Autonomous Underwater Vehicles By Virtual Potential Method. PhD thesis, University of Zagreb, 2012.
- [9] Perez, T. *Ship Motion Control*. Springer, 2005.
- [10] Roger Skjetne, Øyvind Smogeli, and Thor I Fossen. Modeling, identification, and adaptive maneuvering of cybership ii: A complete design with experiments. *IFAC Proceedings Volumes*, 37(10):203–208, 2004.
- [11] Thor I Fossen and Jann Peter Strand. Nonlinear passive weather optimal positioning control (wopc) system for ships and rigs: experimental results. *Automatica*, 37(5):701–715, 2001.
- [12] E Jelavic, J Gonzales, and F Borrelli. Autonomous drift parking using a switched control strategy with onboard sensors. *IFAC-PapersOnLine*, 50(1):3714–3719, 2017.
- [13] Tsutomu Tashiro. Vehicle steering control with mpc for target trajectory tracking of autonomous reverse parking. In *Control Applications (CCA)*, 2013 IEEE International Conference on, pages 247–251. IEEE, 2013.
- [14] Jenhwa Guo. A waypoint-tracking controller for a biomimetic autonomous underwater vehicle. *Ocean engineering*, 33(17-18):2369–2380, 2006.
- [15] KD Do and J Pan. Global waypoint tracking control of underactuated ships under relaxed assumptions. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 2, pages 1244–1249. IEEE, 2003.
- [16] Dong-in Han, Jong-hun Kim, Chan-oh Min, Sung-jin Jo, Jeong-ho Kim, and Dae-woo Lee. Development of unmanned aerial vehicle (uav) system with waypoint tracking and vision-based reconnaissance. *International Journal of Control, Automation and Systems*, 8(5):1091–1099, 2010.
- [17] Avishai Weiss, Morgan Baldwin, Richard Scott Erwin, and Ilya Kolmanovsky. Model predictive control for spacecraft rendezvous and docking: Strategies for handling constraints and case studies. *IEEE Transactions on Control Systems Technology*, 23(4):1638–1647, 2015.
- [18] Djouani, K. and Hamam, Y. Minimum Time-Energy Trajectory Planning for Automatic Ship Berthing. *IEEE Journal of Oceanic Engineering*, Vol 20, 1995.
- [19] Naoki Mizuno*, Yosuke Uchida*, Tadatsugi Okazaki. Quasi Real-Time Optimal Control Scheme for Automatic Berthing. *IFAC PapersOnLine* 48-16, 2015.
- [20] Mao-Wei Nilsson. Autonomous docking and navigation of ships using model predictive control. Master’s thesis, Royal Institute of Technology, KTH, Sweden, 2018.
- [21] B. McCabe. A closer look at Azipod propulsion. *SuperyachtNews*, 2016. <http://www.superyachtnews.com/technology/a-closer-look-atazipod-propulsion>.
- [22] ABB electric propulsion system saved over 700,000 tons of fuel for marine vessels so far, 2016. <https://new.abb.com/news/detail/13783/abb-electric-propulsionsystem-saved-over-700000-tons-of-fuel-for-marine-vessels-so-far>.
- [23] AT Ismail, Alaa Sheta, and Mohammed Al-Weshah. A mobile robot path planning using genetic algorithm in static environment. *Journal of Computer Science*, 4(4):341–344, 2008.
- [24] Randal W Beard, Timothy W McLain, Michael A Goodrich, and Erik P Anderson. Coordinated target assignment and intercept for unmanned air vehicles. *IEEE transactions on robotics and automation*, 18(6):911–922, 2002.
- [25] Jan Maciejowski. *Predictive control: with constraints*. Pearson Education, 2002.
- [26] J. Löfberg. Yalmip: A toolbox for modeling and optimization in matlab. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.