TU Delft

Delft University of Technology

Predictive path following with arrival time awareness for waterborne AGVs

Zheng, Huarong; Negenborn, Rudy; Lodewijks, Gabri

DOI
10.1016/j.trc.2015.11.004

Publication date
2016

Document Version
Final published version

Published in
Transportation Research. Part C: Emerging Technologies

Citation (APA)
Zheng, H., Negenborn, R., & Lodewijks, G. (2016). Predictive path following with arrival time awareness for waterborne AGVs. *Transportation Research. Part C: Emerging Technologies*, *70*, 214-237. https://doi.org/10.1016/j.trc.2015.11.004

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Transportation Research Part C 70 (2016) 214–237

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Predictive path following with arrival time awareness for waterborne AGVs

Huarong Zheng *, Rudy R. Negenborn, Gabriël Lodewijks

Department of Maritime and Transport Technology, Delft University of Technology, Delft 2628 CD, The Netherlands

ARTICLE INFO

Article history:
Received 25 November 2014
Received in revised form 8 October 2015
Accepted 11 November 2015
Available online 30 November 2015

Keywords:
Waterborne AGVs
Inter terminal transport
Path following
Arrival time awareness
Model predictive control

ABSTRACT

Large ports are seeking innovative logistical ways to improve their competitiveness worldwide. This article proposes waterborne AGVs, inspired by conventional automated guided vehicles and autonomous surface vessels, for transport over water. A predictive path following with arrival time awareness controller is proposed for such waterborne AGVs. The controller is able to achieve smooth tracking and energy efficiency with arrival time awareness for transport oriented applications. Tracking errors are conveniently formulated with vessel dynamics modeled in connected reference path coordinate systems and a coordinate transformation at switching coordinate systems. Binary decision variables and logic constraints based on an along-track state are proposed for modeling switches in the framework of Model Predictive Control (MPC) so that overshoots are avoided. Moreover, timing-aware along-track references are generated by a two-level double integrator scheme. The lower level is embedded in online MPC optimizations for smooth tracking. The higher level solves a mixed-integer quadratic programming problem considering distance-to-go and time-to-go before each MPC optimization. References over the next prediction horizon are generated being aware of the requirements on arrival time. Furthermore, successive linearizations of nonlinear vessel dynamics about a shifted previous optimal system trajectory are implemented to maintain a trade-off between computational complexity and optimality. Simulation results of two industrially relevant Inter Terminal Transport case studies illustrate the effectiveness of the proposed modeling and control design for waterborne AGVs.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Cost efficient container distributions are critical for large ports to improve competitiveness in the increasingly globalized economy. The new port area in the Port of Rotterdam, Maasvlakte 2, forming a global container hub complex together with Maasvlakte 1, is expected to handle more than 30 million Twenty-foot Equivalent Unit per year towards 2035 Port of Rotterdam Authority (2011). Large throughput of containers happens both inside terminals, likely handled in automated container terminals by Automated Guided Vehicles (AGVs) Xin et al. (2014), and among various terminals by various modalities (e.g., road, rail, sea), known as Inter Terminal Transport (ITT) Duinkerken et al. (2007). Traffic flow by land has already been heavy considering the limited land in the port area; furthermore, for complex spatial layouts like Maasvlakte 2, the distances between some terminals are much longer by land than by water. Expanding the existing physical transportation

* Corresponding author.
E-mail addresses: h.zheng-1@tudelft.nl (H. Zheng), R.R.Negenborn@tudelft.nl (R.R. Negenborn), G.Lodewijks@tudelft.nl (G. Lodewijks).

http://dx.doi.org/10.1016/j.trc.2015.11.004
0968-090X/© 2015 Elsevier Ltd. All rights reserved.

infrastructure might be an option to relieve these issues, at extremely high costs nonetheless. As an alternative, innovative ways for ITT have to be investigated.

The concept of waterborne AGVs, comparable to conventional AGVs and autonomous surface vessels, is proposed for ITT. The most important criterion of ITT is "non-performance" which happens when the completion time of ITT tasks is later than the permitted latest arrival time (Duinkerken et al., 2007). Mathematical models and exact approaches to minimize ITT delay have been explored from an operational research perspective considering the entire ITT system Tierney et al. (2014). Outputs such as loads, departure and arrival times are possible to be obtained in the transportation system over water from a higher scheduling level Li et al. (2015). From a control perspective considering lower level vehicle dynamics, challenges to fulfill those ITT scheduling tasks arise from various aspects: (*a*) vessels generally have limited manoeuvrability so that they cannot respond timely when environmental information changes, which could lead to undesirable or even dangerous behaviors leading to collisions; (*b*) multiple conflicting design objectives exist including reference tracking, energy efficiency, low "non-performance" rate, etc.; (*c*) various system constraints on states, inputs and outputs due to limited engine power, mechanical maximum deflections/revolutions or spatial no-sailing zones, etc., need to be satisfied; (*d*) complex nonlinear vessel dynamics which model vessel behaviors render the options for applicable control theories very limited; techniques that use simplified representations of these dynamics, however, often result in deteriorated performance; and (*e*) research on autonomous vessels has been restricted to special purposes, e.g., military, mine countermeasure or marine rescue vessels, instead of civilian transport uses Zheng et al. (2013).

Research on applying system and control theories to vessel automation, however, has always been active. The first recognized and most widely implemented controller until now is the classical Proportional–Integral–Derivative control (Minorski, 1922), because of its simplicity both in theory and implementation. However, this simplicity is at the cost of extensive tuning work which largely depends on personal experiences. Another large family of vessel control problems in the literature are solved by combining Lyapunov stability theories and backstepping techniques (Do, 2010) Skjetne et al. (2005). Manoeuvering problems in (Skjetne et al., 2005) are divided into separate geometric tasks and dynamic tasks. Separate tracking controllers based on backstepping are then designed to accomplish these separate tasks. Nevertheless, none of the aforementioned works consider system constraints explicitly nor are they able to quantify control performance in a systematic way. Neglecting constraints in controller design might lead to instability or even damages to the system; neglecting optimizing system performance at all means the system might not have a cost-effective solution, e.g., the system might be driven to a set-point regardless of the energy taken. Reliability and economical operations are, however, critical by all means for logistics oriented waterborne AGVs.

Model Predictive Control (MPC) solves a constrained optimization problem repetitively online in a receding horizon way. It has been successfully applied in a large range of transport related scenarios, e.g., intermodal freight transport Li et al. (2015), automated container terminals Xin et al. (2015) and traffic networks Lin et al. (2012) as an advanced and effective control methodology. Its theoretical basis as well as stability, optimality, and robustness properties are well understood for certain classes of systems (Mayne et al., 2000). Besides its ability to achieve optimal system performance w.r.t. specified design criteria within system limits, MPC in principle has the advantage that it can take into account all predictable future information available and that it can therefore anticipate undesirable future situations at an early stage. This predictive feature can be used to compensate for the weak manoeuvrability of vessel dynamics and thus renders MPC particularly suitable for vessel motion control problems. A straight line path following problem is addressed using MPC in conjunction with the Line-Of-Sight (LOS) guidance law in (Oh and Sun, 2010). The idea of LOS based design is to control a vessel's heading angle in such a way that it converges to an LOS angle. The cross-track error can then be proved to converge to zero realizing path following. However, unreal assumptions on velocities and cross-track errors are made for using LOS in MPC. Moreover, LOS based design (Fossen et al., 2003,) including an improved LOS formulation in (Oh and Sun, 2010) commonly suffers from overshoots during switching waypoints. The control design based on LOS guidance might reduce complexity, however, at the same time, also flexibility of the controller. Occasions such as an imminent hazardous area to avoid and coupled desirable speed assignment for smooth tracking or fulfilling timing requirements are unlikely to be incorporated. In this article, a novel predictive path following with arrival time awareness (PPF-ATA) controller relying on MPC is proposed with which problems as overshoots, smooth tracking, timing, etc., are well handled.

In particular, we establish connected coordinate systems based on straight-line reference paths, in which system kinematics are modeled. The benefits of doing so are twofold: first, cross-track and along-track errors can be formulated more compactly; secondly, the along-track state is utilized in a reference switching logic so that overshoots are avoided. The switching logic combined with a coordinate transformation renders a continuous model in one coordinate system still applicable for successive linearizations which are important to maintain a trade-off between computational complexity and optimality. Moreover, to achieve both smooth tracking and arrival time awareness, we propose double-integrator dynamics in two levels to parameterize reference paths. The lower level is embedded in online MPC optimizations for smooth tracking. The higher level solves a mixed-integer Quadratic Programming (MIQP) problem considering distance-to-go and time-to-go each sampling step. Timing-aware but energy optimal references over the next prediction horizon are generated for the online MPC optimization. Arrival time awareness is thus achieved in a sense that the vessel will arrive at a preferable time when the transport task is feasible or with a minimal delay with respect to the preferable arrival time within a specified time-window otherwise. Case studies of two industrially relevant ITT tasks in the Port of Rotterdam are set. Simulations are run to illustrate the effectiveness of the proposed model and control framework for waterborne AGVs. The problem of path following with timing requirements, to the best of our knowledge, has not been presented in the scientific literature

to date. Therefore, this paper proposes a comprehensive solution applicable to a large category of engineering tasks including but not limited to waterborne AGVs for ITT.

The remainder of this article is organized as follows. In Section 2, connected path coordinate systems are established in which 3 degree-of-freedom (DOF) surface vessel dynamics are modeled for path following. Successive linearized prediction models for use in MPC are derived in Section 3. Algorithms for solving the problem of predictive path following with arrival time awareness are formulated in Section 4. Then in Section 5, simulation experiments and results are presented, followed by concluding remarks and future research directions in Section 6.

2. Modeling of surface vessels for path following

Models of marine crafts with different DOF for different purposes have been elaborated on in (Fossen, 2011). For tracking problems of surface vessels, models with 3 DOF in the horizontal plane are sufficient to capture the main system characteristics (Fossen, 2011). Based on a 3 DOF maneuvering model considering constant current (Skjetne, 2005), we introduce the modeling of surface vessels for path following problems. Connected path coordinate systems are established in which vessel kinematics are modeled. Benefits of doing so have been clarified in Section 1. Moreover, transformations of states between different coordinate systems for later controller design are also modeled in this section.

2.1. A marine surface vessel model

Motions of a marine surface vessel are generally described in two coordinate frames, namely the inertial frame $\{n\}$ and the body-fixed frame $\{b\}$. Following the vectorial setting in (Fossen, 2011), behaviors of a 3 DOF surface vessel considering a constant (fixed speed and angle in the inertial frame) current and neglecting other environmental disturbances are mathematically modeled as[1]:

$$\dot{\boldsymbol{\eta}} = \boldsymbol{R}(\psi)\boldsymbol{\nu}, \tag{1}$$

$$\boldsymbol{M}_{\mathrm{RB}}\dot{\boldsymbol{\nu}} + \boldsymbol{M}_{\mathrm{A}}\dot{\boldsymbol{\nu}}_{\mathrm{r}} + \boldsymbol{C}_{\mathrm{RB}}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{C}_{\mathrm{A}}(\boldsymbol{\nu}_{\mathrm{r}})\boldsymbol{\nu}_{\mathrm{r}} + \boldsymbol{D}(\boldsymbol{\nu}_{\mathrm{r}})\boldsymbol{\nu}_{\mathrm{r}} = \boldsymbol{\tau}. \tag{2}$$

The above model involves six system states and three control inputs: three pose states described in $\{n\}$, i.e., $\boldsymbol{\eta} = [x \quad y \quad \psi]^{\mathrm{T}}$ and three velocity states described in $\{b\}$, i.e., $\boldsymbol{\nu} = [u \quad v \quad r]^{\mathrm{T}}$; $\boldsymbol{\tau} = [\tau_u \quad \tau_v \quad \tau_r]^{\mathrm{T}}$ is the system input vector. See Fig. 1 and Table 1 for a further explication of the notations.

In (1), $\boldsymbol{R}(\psi)$ is a rotation matrix relating motions in coordinate systems of $\{n\}$ and $\{b\}$ and is defined as:

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

where ψ is the heading angle. $\boldsymbol{R}$ satisfies $\frac{\mathrm{d}}{\mathrm{d}t}\{\boldsymbol{R}(\psi)\} = \dot{\psi}\boldsymbol{R}(\psi)\boldsymbol{S}$ where

$$\boldsymbol{S} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

In (2), $\boldsymbol{\nu}_{\mathrm{r}} = \boldsymbol{\nu} - \boldsymbol{\nu}_{\mathrm{c}} = [u_{\mathrm{r}} \quad v_{\mathrm{r}} \quad r_{\mathrm{r}}]^{\mathrm{T}}$ is the relative velocity in the body-fixed frame between ship hull and the fluid. For a non-rotational current with fixed speed V_{c} and angle β_{c} in $\{n\}$ ($\dot{V}_{\mathrm{c}} = 0$ and $\dot{\beta}_{\mathrm{c}} = 0$), a rotation to $\{b\}$ would be

$$\boldsymbol{\nu}_{\mathrm{c}} = \boldsymbol{R}(\psi)^{\mathrm{T}} \begin{bmatrix} V_{\mathrm{c}}\cos\beta_{\mathrm{c}} \\ V_{\mathrm{c}}\sin\beta_{\mathrm{c}} \\ 0 \end{bmatrix}.$$

Therefore,

$$\dot{\nu}_{\mathrm{r}} = \dot{\nu} - r\boldsymbol{S}^{\mathrm{T}}\boldsymbol{\nu}_{\mathrm{c}}.$$

Rigid-body and added mass matrices are given as:

$$\boldsymbol{M}_{\mathrm{RB}} = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mx_{\mathrm{g}} \\ 0 & mx_{\mathrm{g}} & I_{\mathrm{z}} \end{bmatrix}, \quad \boldsymbol{M}_{\mathrm{A}} = \begin{bmatrix} -X_{\dot{u}} & 0 & 0 \\ 0 & -Y_{\dot{v}} & -Y_{\dot{r}} \\ 0 & -N_{\dot{v}} & -N_{\dot{r}} \end{bmatrix},$$

[1] For the sake of notation simplicity, time indices for system states and control inputs in vessel dynamic models are left out in this section.

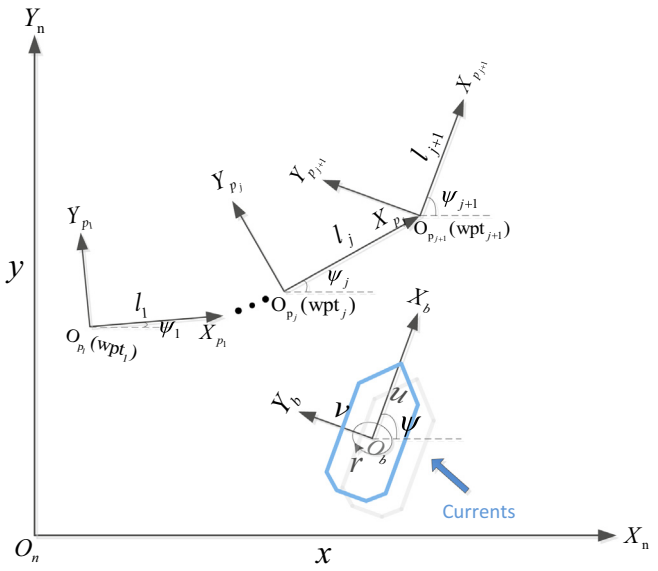


Fig. 1. Motions of a surface vessel in the horizontal plane.

Table 1
Notations in models of marine surface vessels for path following.

	Notations	Physical meanings	Units
States	x/x_{p_j}	Displacement along X_n/X_{p_j} axis	m
	y/y_{p_j}	Displacement along Y_n/Y_{p_j} axis	m
	ψ/ψ_{p_j}	Heading angle w.r.t X_n/X_{p_j} axis	rad
	u	Surge velocity along X_{b} axis	m/s
	v	Sway velocity along Y_{b} axis	m/s
	r	Angular velocity of heading angle	rad/s
Inputs	τ_u	Surge force along X_{b} axis	N
	τ_v	Sway force along Y_{b} axis	N
	τ_r	Yaw moment	Nm

where subscripts $\cdot_{\mathrm{RB}}$ and $\cdot_{\mathrm{A}}$ stand for rigid body and added force related matrices, respectively; m is the mass of the vessel; I_z is the moment of inertia in the yaw rotation; and x_g is the distance between the center of gravity of the vessel to the center of the body-fixed coordinate frame. Similarly,

$$\boldsymbol{C}_{\mathrm{RB}}(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & -m(x_g r + v) \\ 0 & 0 & mu \\ m(x_g r + v) & -mu & 0 \end{bmatrix},$$

$$\boldsymbol{C}_{\mathrm{A}}(\boldsymbol{\nu}_{\mathrm{r}}) = \begin{bmatrix} 0 & 0 & Y_{\dot{v}} v_r + (N_{\dot{v}} + Y_{\dot{r}}) r/2 \\ 0 & 0 & -X_{\dot{u}} u_r \\ -Y_{\dot{v}} v_r - (N_{\dot{v}} + Y_{\dot{r}}) r/2 & X_{\dot{u}} u_r & 0 \end{bmatrix}$$

are rigid-body and added Coriolis and centripetal matrices, respectively.

For the damping force $\boldsymbol{D}(\boldsymbol{\nu}_{\mathrm{r}}) = \boldsymbol{D}_l + \boldsymbol{D}_{\mathrm{NL}}(\boldsymbol{\nu}_{\mathrm{r}})$, which has been separated into two parts: a linear part as

$$\boldsymbol{D}_l = \begin{bmatrix} -X_u & 0 & 0 \\ 0 & -Y_v & -Y_r \\ 0 & -N_v & -N_r \end{bmatrix}$$

and a nonlinear part as

$$\boldsymbol{D}_{\mathrm{NL}}(\boldsymbol{\nu}_r) = \begin{bmatrix} -X_{|u|u}|u_r| - X_{uuu} u_r^2 & 0 & 0 \\ 0 & -Y_{|v|v}|v_r| - Y_{|r|v}|r| & -Y_{|v|r}|v_r| - Y_{|r|r}|r| \\ 0 & -N_{|v|v}|v_r| - N_{|r|v}|r| & -N_{|v|r}|v_r| - N_{|r|r}|r| \end{bmatrix}.$$

Hydrodynamic derivatives follow the notations in (SNAME, 1952). For instance, the hydrodynamic added mass force X along the x axis due to an acceleration $\dot{u}$ in the x direction is written as

$$X = -X_{\dot{u}}\dot{u}, \quad X_{\dot{u}} := \frac{\partial X}{\partial \dot{u}},$$

which implies $\{\boldsymbol{M}_{\mathrm{A}}\}_{11} = -X_{\dot{u}}$.

For controller design and system simulation, (1) and (2) are then written into the following two ordinary differential equations[2]

$$\begin{aligned} \dot{\boldsymbol{\eta}} &= \boldsymbol{R}(\psi)\boldsymbol{v}, \\ \dot{\boldsymbol{v}} &= (\boldsymbol{M}_{\mathrm{RB}} + \boldsymbol{M}_{\mathrm{A}})^{-1}\Big(\boldsymbol{\tau} - (\boldsymbol{D}_l + \boldsymbol{D}_{\mathrm{NL}}(\boldsymbol{v}_{\mathrm{r}}) + \boldsymbol{C}_{\mathrm{A}}(\boldsymbol{v}_{\mathrm{r}}))\boldsymbol{v}_{\mathrm{r}} - \boldsymbol{C}_{\mathrm{RB}}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{M}_{\mathrm{A}} r \boldsymbol{S}^{\mathrm{T}}\boldsymbol{v}_{\mathrm{c}}\Big), \end{aligned} \tag{3}$$

with $\begin{bmatrix}\boldsymbol{\eta}^{\mathrm{T}} & \boldsymbol{v}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$ as the system states and $\boldsymbol{\tau}$ as system inputs.

2.2. Modeling in path coordinate systems

For a given ITT task in this article, we assume the route information is in the form of connected straight-line segments which is the shortest reference path connecting the origin and destination. The goal is then to control the vessel to move along these reference paths while arrive at the destination as punctual as possible in an economical way. Two of the system design requirements are first distinguished here: minimizing the cross-track error which is defined as the distance from the vessel's current position to the reference line and minimizing the along-track error which is defined as the error between the orthogonal projection point of the vessel on the reference line and an reference along path position. By modeling vessel kinematics in path coordinate systems, we can then conveniently formulate cross-track and along-track errors as linear functions of system states. Apart from that, the along-track state can be taken advantage of in a switching logic to avoid overshoots which commonly exist in LOS guidance laws. Our modeling in path coordinate systems are mainly motivated by these two reasons.

As shown in Fig. 1, the path coordinate systems $\{p_j\}$ $(j = 1, 2, \ldots)$ are based on the connected reference paths. X_{p_j} is along the reference path and Y_{p_j} is vertical to the reference path pointing $\pi/2$ counterclockwise. O_{p_j} is the origin of the jth path coordinate system located at the jth waypoint connecting reference path $j-1$ and j. Lengths and angles w.r.t. X_{n} of reference path j are denoted as l_j and ψ_j, respectively. Then kinematics are modeled in path coordinate system $\{p_j\}$ as:

$$\dot{\boldsymbol{\eta}}_{\mathrm{p}_j} = \boldsymbol{R}(\psi_{\mathrm{p}_j})\boldsymbol{v}, \tag{4}$$

where, likewise, $\boldsymbol{\eta}_{\mathrm{p}_j} = \begin{bmatrix} x_{\mathrm{p}_j} & y_{\mathrm{p}_j} & \psi_{\mathrm{p}_j}\end{bmatrix}^{\mathrm{T}}$ is the pose expressed in $\{p_j\}$ with $\psi_{\mathrm{p}_j} = \psi - \psi_j$. $\boldsymbol{R}(\psi_{\mathrm{p}_j})$ is a rotation matrix relating motions between coordinate systems $\{p_j\}$ and $\{b\}$ and defined as:

$$\boldsymbol{R}(\psi_{\mathrm{p}_j}) = \begin{bmatrix} \cos(\psi_{\mathrm{p}_j}) & -\sin(\psi_{\mathrm{p}_j}) & 0 \\ \sin(\psi_{\mathrm{p}_j}) & \cos(\psi_{\mathrm{p}_j}) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Kinetics of the vessel system are still expressed in frame $\{b\}$ as (2). Since vessel heading angle ψ is involved in (2) to transform current dynamics from $\{n\}$ to $\{b\}$, a path coordinate system model with seven states is preferred and implemented for controller design as:

$$\begin{aligned} \dot{\boldsymbol{\eta}}_{\mathrm{p}_j} &= \boldsymbol{R}(\psi_{\mathrm{p}_j})\boldsymbol{v}, \\ \dot{\psi} &= r, \\ \dot{\boldsymbol{v}} &= (\boldsymbol{M}_{\mathrm{RB}} + \boldsymbol{M}_{\mathrm{A}})^{-1}\Big(\boldsymbol{\tau} - (\boldsymbol{D}_l + \boldsymbol{D}_{\mathrm{NL}}(\boldsymbol{v}_{\mathrm{r}}) + \boldsymbol{C}_{\mathrm{A}}(\boldsymbol{v}_{\mathrm{r}}))\boldsymbol{v}_{\mathrm{r}} - \boldsymbol{C}_{\mathrm{RB}}(\boldsymbol{v})\boldsymbol{v} + \boldsymbol{M}_{\mathrm{A}} r \boldsymbol{S}^{\mathrm{T}}\boldsymbol{v}_{\mathrm{c}}\Big), \end{aligned} \tag{5}$$

which is continuous within one coordinate system, however, continuity loses during a switch of coordinate systems. A transformation including rotation and translation of coordinates is then necessary.

More specifically, we consider a switch from $\{p_j\}$ to $\{p_{j+1}\}$, as shown in Fig. 1. The angle error between the new 'x' axis $X_{\mathrm{p}_{j+1}}$ and the old 'x' axis X_{p_j} is $\psi_{j+1} - \psi_j$. Since $\{p_j\}$ and $\{p_{j+1}\}$ are connected, the new origin $O_{\mathrm{p}_{j+1}}$ has coordinates $(l_j, 0)$ relative to the old coordinate system $\{p_j\}$. A transformation of coordinates from $\{p_j\}$ to $\{p_{j+1}\}$ would then be:

$$\begin{bmatrix} x_{\mathrm{p}_{j+1}} \\ y_{\mathrm{p}_{j+1}} \end{bmatrix} = \begin{bmatrix} \cos(\psi_{j+1} - \psi_j) & \sin(\psi_{j+1} - \psi_j) \\ -\sin(\psi_{j+1} - \psi_j) & \cos(\psi_{j+1} - \psi_j) \end{bmatrix} \begin{bmatrix} x_{\mathrm{p}_j} - l_j \\ y_{\mathrm{p}_j} - 0 \end{bmatrix}. \tag{6}$$

[2] $\boldsymbol{M}_{\mathrm{RB}}$ and $\boldsymbol{M}_{\mathrm{A}}$ are given as constant matrices here, and the inverse $(\boldsymbol{M}_{\mathrm{RB}} + \boldsymbol{M}_{\mathrm{A}})^{-1}$ exists for the given matrices.

Meanwhile, a transformation of the heading angle from $\{p_j\}$ to $\{p_{j+1}\}$ would be:

$$\psi_{\mathrm{p}_{j+1}} = \psi_{\mathrm{p}_j} + \psi_j - \psi_{j+1}, \tag{7}$$

or simply as:

$$\psi_{\mathrm{p}_{j+1}} = \psi - \psi_{j+1} \tag{8}$$

since ψ is directly measurable from (3) for a transformation of initial states and available in (5) for a transformation of coordinates over the prediction horizon, as to be clarified further in the following sections.

To obtain initial states for (5), the measured states from vessel system (3) need to be transformed from $\{n\}$ to $\{p_j\}$. Similar with (6), a transformation of position is:

$$\begin{bmatrix} x_{\mathrm{p}_j} \\ y_{\mathrm{p}_j} \end{bmatrix} = \begin{bmatrix} \cos(\psi_j) & \sin(\psi_j) \\ -\sin(\psi_j) & \cos(\psi_j) \end{bmatrix} \begin{bmatrix} x - x_{\mathrm{wpt}_j} \\ y - y_{\mathrm{wpt}_j} \end{bmatrix}, \tag{9}$$

where $\left(x_{\mathrm{wpt}_j}, y_{\mathrm{wpt}_j}\right)$ is the coordinate of waypoint j, or origin of $\{p_j\}$ in $\{n\}$. The transformation of heading angle is then the same with (8). So far, we have derived model (5) with coordinate transformations of (6) and (8) for transforming coordinates between $\{p_j\}$ and $\{p_{j+1}\}$, and (8) and (9) for transforming coordinates between $\{n\}$ and $\{p_j\}$, which is then readily usable for later controller design and system simulation.

3. Successively linearized prediction models for MPC

The complexity of an MPC controller with certain controller parameters mainly depends on the characteristics (order, nonlinearities, etc.) of the prediction model it uses. For highly nonlinear systems, a dilemma is usually faced by MPC: on the one hand, stringent demands on system performance generally require high accuracy system model; on the other hand, online predictions and optimizations of future system behaviors based on complex system model are too time consuming, which limits waterborne AGVs to anything but real-time transportation applications. A compromise between optimality and computational complexity has to be made for fast and nonlinear vessel dynamics. Efficient numerical algorithms such as the continuation/GMRES method Ohtsuka (2004) can achieve real-time implementations of nonlinear MPC controllers with bounded sub-optimality for highly nonlinear systems. A similar approach that also perturbs previously computed trajectories, but that uses fixed prediction horizons, is employed in this article. Successive linearizations are implemented by linearizing path coordinate system models about a shifted optimal trajectory from a previous step at each prediction step to have a trade-off between optimality and computational complexity.

The nonlinear model to be linearized for prediction is (5) which models vessel motions in path coordinate system $\{p_j\}$ and is generalized as:

$$\dot{\boldsymbol{x}}_{\mathrm{p}_j}(t) = \boldsymbol{f}(\boldsymbol{x}_{\mathrm{p}_j}(t), \boldsymbol{u}(t)), \tag{10}$$

where $\boldsymbol{f} : \mathbb{R}^7 \times \mathbb{R}^3 \to \mathbb{R}^7$ is a nonlinear smooth function with system states $\boldsymbol{x}_{\mathrm{p}_j} = \begin{bmatrix} \boldsymbol{\eta}_{\mathrm{p}_j}^{\mathrm{T}} & \psi & \boldsymbol{\nu}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ and control inputs $\boldsymbol{u} = \boldsymbol{\tau}$. The subscript $\cdot_{\mathrm{p}_j}$ indicates a state modeled in $\{p_j\}$.

Remark 1. The nonlinear smooth functions $\boldsymbol{f}$ are the same in different path coordinate systems, as are the control inputs $\boldsymbol{u}$. A switch of coordinate system only affects $\boldsymbol{x}_{\mathrm{p}_j}$. Hence, the overall model of (5) in different path coordinate systems can be seen as piecewise repeated nonlinear functions but discontinuous. For the discretization and linearization framework subsequent to hold, a continuous model in one path coordinate system needs to be considered. Predictive switches of adjacent coordinate systems and coordinate transformations are to be modeled and approached in the next section.

For numerical simulations, continuous time model (10) is usually discretized with zero-order-hold assumption as:

$$\boldsymbol{x}_{\mathrm{p}_j}(k+1) = \boldsymbol{x}_{\mathrm{p}_j}(k) + \int_{kT_s}^{(k+1)T_s} \boldsymbol{f}(\boldsymbol{x}_{\mathrm{p}_j}(k), \boldsymbol{u}(k))\mathrm{d}t, \tag{11}$$

where T_s is the sampling time and k is a discrete time step standing for time instant kT_s. Consider at time step $k-1$ $(k>1)$ in MPC, the calculated optimal control input sequence is $\boldsymbol{u}(k-1+i|k-1)$ for $i = 0, 1, \ldots, N_{\mathrm{p}} - 1$, where N_{p} is the prediction horizon. Conventionally, the first element $\boldsymbol{u}(k-1|k-1)$ is applied to the system and the rest are disregarded. For linearizations at step k, however, we make extensive use of this 'tail' to build a seed trajectory (Kouvaritakis et al., 1999) $\left(\boldsymbol{x}_{\mathrm{p}_j}^0(k+i|k), \boldsymbol{u}^0(k+i|k)\right)$ for $i = 0, 1, \ldots, N_{\mathrm{p}} - 1$, where

$$\boldsymbol{u}^0(k+i|k) = \boldsymbol{u}(k+i|k-1) \tag{12}$$

for $i = 0, 1, \ldots, N_{\mathrm{p}} - 2$ and

$$\boldsymbol{u}^0(k+N_{\mathrm{p}}-1|k) = \boldsymbol{u}(k+N_{\mathrm{p}}-2|k-1). \tag{13}$$

Then, with an initial state $\boldsymbol{x}^0_{\mathrm{p}_j}(k|k) = \boldsymbol{x}_{\mathrm{p}_j}(k)$, solutions for (10) at time steps $k+i$, i.e., $\boldsymbol{x}^0_{\mathrm{p}_j}(k+i|k)$ for $i = 1, 2, \ldots, N_{\mathrm{p}}$ can be calculated according to (11).[3] Note here, path coordinate system states $\boldsymbol{x}_{\mathrm{p}_j}(k)$ are not directly measurable, but calculated by (8) and (9) given the measurable states $\boldsymbol{x}(k)$. Hereby, $k+i|k$ stands for the ith element of the predicted trajectories at step k and the superscript ^{0} denotes a seed trajectory. Small perturbations around the seed trajectory are denoted as $\left(\Delta\boldsymbol{x}_{\mathrm{p}_j}(k+i|k), \Delta\boldsymbol{u}(k+i|k)\right)$ for $i = 0, 1, \ldots, N_{\mathrm{p}} - 1$ and satisfy

$$\begin{aligned} \boldsymbol{x}_{\mathrm{p}_j}(k+i|k) &= \boldsymbol{x}^0_{\mathrm{p}_j}(k+i|k) + \Delta\boldsymbol{x}_{\mathrm{p}_j}(k+i|k), \text{ for } i = 0, 1, \ldots, N_{\mathrm{p}} \\ \boldsymbol{u}(k+i|k) &= \boldsymbol{u}^0(k+i|k) + \Delta\boldsymbol{u}(k+i|k), \text{ for } i = 0, 1, \ldots, N_{\mathrm{p}} - 1. \end{aligned} \tag{14}$$

Substituting (14) into (11), we get

$$\begin{aligned} \boldsymbol{x}^0_{\mathrm{p}_j}(k+1+i|k) + \Delta\boldsymbol{x}_{\mathrm{p}_j}(k+i+1|k) = \boldsymbol{x}^0_{\mathrm{p}_j}(k+i|k) + \Delta\boldsymbol{x}_{\mathrm{p}_j}(k+i|k) + \int_{kT_s}^{(k+1)T_s} \boldsymbol{f}(\boldsymbol{x}^0_{\mathrm{p}_j}(k+i|k) + \Delta\boldsymbol{x}_{\mathrm{p}_j}(k+i|k), \boldsymbol{u}^0(k+i|k) \\ + \Delta\boldsymbol{u}(k+i|k))\mathrm{d}t, \end{aligned} \tag{15}$$

for $i = 0, 1, \ldots, N_{\mathrm{p}} - 1$. The term in the integrator, by applying Taylor's theorem and neglecting the higher order terms than the first order, can be written as:

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}^0_{\mathrm{p}_j}(k+i|k) + \Delta\boldsymbol{x}_{\mathrm{p}_j}(k+i|k), \boldsymbol{u}^0(k+i|k) + \Delta\boldsymbol{u}(k+i|k)) = \boldsymbol{f}(\boldsymbol{x}^0_{\mathrm{p}_j}(k+i|k), \boldsymbol{u}^0(k+i|k)) + \boldsymbol{A}_{\mathrm{c}}(k+i|k)\Delta\boldsymbol{x}_{\mathrm{p}_j}(k+i|k) \\ + \boldsymbol{B}_{\mathrm{c}}(k+i|k)\Delta\boldsymbol{u}(k+i|k), \end{aligned} \tag{16}$$

where

$$\boldsymbol{A}_{\mathrm{c}}(k+i|k) = \left.\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{x}}\right|_{\left(\boldsymbol{x}^0_{\mathrm{p}_j}(k+i|k), \boldsymbol{u}^0(k+i|k)\right)}, \boldsymbol{B}_{\mathrm{c}}(k+i|k) = \left.\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{u}}\right|_{\left(\boldsymbol{x}^0_{\mathrm{p}_j}(k+i|k), \boldsymbol{u}^0(k+i|k)\right)} \tag{17}$$

are continuous Jacobian state and input matrices, respectively and can also be discretized with zero-order-hold assumption to get corresponding $\boldsymbol{A}_{\mathrm{d}}$ and $\boldsymbol{B}_{\mathrm{d}}$. Then by (15) and (16), we reach the discrete linearized incremental model

$$\Delta\boldsymbol{x}_{\mathrm{p}_j}(k+i+1|k) = \boldsymbol{A}_{\mathrm{d}}(k+i|k)\Delta\boldsymbol{x}_{\mathrm{p}_j}(k+i|k) + \boldsymbol{B}_{\mathrm{d}}(k+i|k)\Delta\boldsymbol{u}(k+i|k). \tag{18}$$

with an initial state $\Delta\boldsymbol{x}_{\mathrm{p}_j}(k|k) = 0$ since $\boldsymbol{x}^0_{\mathrm{p}_j}(k|k) = \boldsymbol{x}_{\mathrm{p}_j}(k)$. To this end, we are able to predict future system behaviors in a linear way for nonlinear system (5) by (14) and (18). Procedures to obtain the linearized prediction models to design an MPC controller for our vessel path following problem can be summarized as Fig. 2.

4. Predictive path following with arrival time awareness

Predictive vessel behaviors are approximated by the linearized models introduced in Section 3 based on path coordinate system nonlinear models described in Section 2. The issue following is to drive these predicted behaviors to desired behaviors according to system design requirements. In our case, to fulfill an ITT task in an economical way generally requires:

1. a given geometric reference path is tracked with as small as possible deviations;
2. a given arrival time requirement is met when a preferable time is feasible considering system limitations, or a minimal delay w.r.t. the preferable time within a specified time window (it is assumed the task can be in any case accomplished by the higher limit of the time window) otherwise; and
3. the aforementioned two design requirements are achieved in an energy economical way.

The first requirement can be reached by minimizing cross-track errors, as defined in Section 2, readily available in path coordinate system models w.r.t. a reference path segment. However, for the to be optimized and thus unknown predicted vessel states, which path segment is being targeted over the prediction horizon is also unknown. We propose a switching logic based on along-track states and lengths of path segments. Following this logic, we will show how predictive switching of reference path segments and coordinate systems are realized with the linearization framework as derived in Section 3 still holding in Section 4.2 by introducing binary decision variables and coordinate transformations. The second task is also related to the along-track state. To obtain smoothly trackable along-track references and at the same time satisfying the timing requirements, the geometric path is parameterized by double-integrator dynamics which are then modeled in two levels. The lower level is embedded in online MPC optimizations. The higher level solves an MIQP problem considering distance-to-go and time-to-go which are feedback from the lower level double-integrator dynamics and provides the lower level timing aware references over the next receding prediction horizon. Finally, the third task is approached by solving online optimiza-

[3] Ordinary differential equation solvers (e.g., ode45 in MATLAB (MATLAB, 2011)) are able to provide higher precision ODE solutions with specified tolerance.

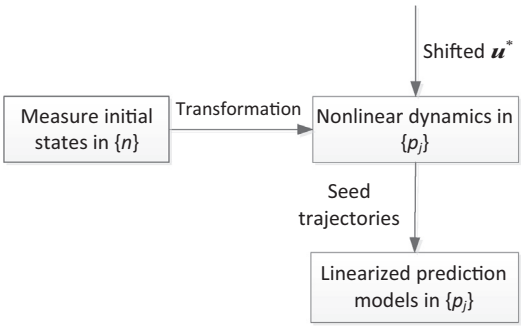


Fig. 2. Successively linearized prediction models for MPC.

tion problems within MPC. A comprehensive algorithm for the PPF-ATA problem is summarized in Section 4.3. Next, we first formulate the two-level double integrator dynamics for path parameterizations.

4.1. Two-level double integrator for path following with timing

For an ITT task, the reference path is given as time-independent straight-line segments, which means a path following rather than a trajectory tracking problem needs to be solved. Trajectory tracking requires time parameterized references while path following is usually parameterized geometrically by introducing a path parameter (Yu et al., 2015). Since no temporal constraints are posed on path following, literature on this topic generally has overlooked the timing problem. To design a PPF-ATA controller, we propose a two-level double integrator based structure, as shown in Fig. 3. In this structure, smoothly trackable along-track references which also satisfy timing requirements are generated in a receding way. To do so, the geometric path is parameterized by double integrator dynamics (henceforth called s dynamics) which are modeled as:

$$\boldsymbol{x}_{\mathrm{s}}(k+1) = \boldsymbol{A}_{\mathrm{s}}\boldsymbol{x}_{\mathrm{s}}(k) + \boldsymbol{B}_{\mathrm{s}}u_{\mathrm{s}}(k), \tag{19}$$

where $\boldsymbol{x}_{\mathrm{s}} = [\,s \quad v_{\mathrm{s}}\,]^{\mathrm{T}}$ and $u_{\mathrm{s}} = a_{\mathrm{s}}$. The scalar s is introduced for parameterizing the straight-line reference paths with v_{s} as its velocity and a_{s} as its acceleration. State and input matrices are:

$$\boldsymbol{A}_{\mathrm{s}} = \begin{bmatrix} 1 & T_{\mathrm{s}} \\ 0 & 1 \end{bmatrix}, \quad \boldsymbol{B}_{\mathrm{s}} = \begin{bmatrix} T_{\mathrm{s}}^2/2 \\ T_{\mathrm{s}} \end{bmatrix}.$$

To distinguish notations in two levels, we use subscript $\cdot_{\mathrm{s,l}}$ denoting lower level variables and $\cdot_{\mathrm{s,h}}$ indicating higher level ones. In the lower level, given an initial state $\boldsymbol{x}_{\mathrm{s,l}}(k)$, predicted trajectories over a prediction horizon are:

$$\boldsymbol{x}_{\mathrm{s,l}}(k+i+1|k) = \boldsymbol{A}_{\mathrm{s}}\boldsymbol{x}_{\mathrm{s,l}}(k+i|k) + \boldsymbol{B}_{\mathrm{s}}u_{\mathrm{s,l}}(k+i|k), \tag{20}$$

for $i = 0, 1, \ldots, N_{\mathrm{p}}$ with $\boldsymbol{x}_{\mathrm{s,l}}(k|k) = \boldsymbol{x}_{\mathrm{s,l}}(k)$. Prediction model (20) is then embedded in online MPC optimizations which are to be formulated in Section 4.3.

The higher level shares the same s dynamics (19) with the lower level. An MIQP problem is formulated aiming at generating an optimal reference trajectory for the lower level over the next receding prediction horizon, i.e., $s^{\mathrm{r}}(k+i|k)$, for $i = 1, 2, \ldots, N_{\mathrm{p}}$. For the MIQP, we specify the main objective as guaranteeing a required arrival at $s_{t_{\mathrm{f}}}$ at time t_{f}. Terminal state $s_{t_{\mathrm{f}}}$ is set as the total length of all the path segments.

Considering limitations of vessel dynamics, s dynamics cannot evolve freely as well. Vessel's maximum surge speed $u_{\max}$ is imposed as a state constraint for $v_{\mathrm{s,u}}$ in MIQP. Due to this speed limit, there is a feasibility issue for a specific ITT task: if the scheduled arrival time is too stringent, the vessel can never manage to arrive on time even if it sails at its highest power. In reality, a time window is often assigned to allow for an acceptable delay Δt in terms of a preferable arrival time t_{r}. Finite flexibility is thus set for the arrival time by $t_{\mathrm{f}} \in [\underline{t_{\mathrm{f}}}, \overline{t_{\mathrm{f}}}]$, where $\underline{t_{\mathrm{f}}} = t_{\mathrm{r}}$ and $\overline{t_{\mathrm{f}}} = t_{\mathrm{r}} + \Delta t$ and we assume by $\overline{t_{\mathrm{f}}}$, the arrival can by all means be achieved. In this case, the problem becomes a constrained optimal control problem with a fixed terminal state and a minimal arrival time (Kirk, 2012). However, the minimal arrival time should be within the time window $[\underline{t_{\mathrm{f}}}, \overline{t_{\mathrm{f}}}]$. Next, we show how this can be implemented in MIQP using binary variables.

In a discrete time setting, we denote $T_{\mathrm{f}}(k)$ as the calculated arrival time step at time step k, N and $N_{\max}$ corresponding to continuous time $\underline{t_{\mathrm{f}}}$ and $\overline{t_{\mathrm{f}}}$, respectively. Therefore, $T_{\mathrm{f}}(k)$, N and $N_{\max}$ satisfy $T_{\mathrm{f}}(k) \in [N, N_{\max}]$. The cost function is separated into two parts:

$$J_{\mathrm{s}}(k) = J_{\mathrm{s}}^1(k) + J_{\mathrm{s}}^2(k), \tag{21}$$

where $J_{\mathrm{s}}^1(k)$ is written as a summation from the current time step k to time step $N-1$, i.e.,

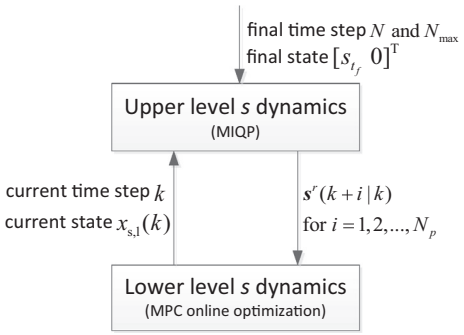


Fig. 3. Two-level double integrator structure for smooth tracking with arrival time awareness.

$$J_s^1(k) = \sum_{n=k}^{N-1}\left(\|u_{s,h}(n)\|_{w_1}^2 + \|\boldsymbol{x}_{s,h}(n)\|_{\boldsymbol{w}_2}^2\right) \tag{22}$$

subject to, for $n = k,\ k+1,\ldots,N-1$

$$\boldsymbol{x}_{s,h}(n+1) = \boldsymbol{A}_s\boldsymbol{x}_{s,h}(n) + \boldsymbol{B}_s u_{s,h}(n), \tag{23}$$

$$0 <= \boldsymbol{x}_{s,h}(n) <= u_{\max}. \tag{24}$$

Notation $\|\cdot\|_w^2$ stands for weighted vector two norms, e.g., $\|u_{s,h}(n)\|_{w_1}^2 = u_{s,h}(n)^{\mathrm{T}} w_1 u_{s,h}(n)$. A minimization of the two norms of $u_{s,h}(k)$ and $\boldsymbol{x}_{s,h}(k)$ is to guarantee an energy optimal and a smoothness of s dynamics. w_1 and $\boldsymbol{w}_2$ are weighting parameter and matrix for $u_{s,h}(k)$ and $\boldsymbol{x}_{s,h}(k)$, respectively.

The second part of $J_s(k)$, $J_s^2(k)$, is a summation over the time window $[N, N_{\max}]$, defined as

$$J_s^2(k) = \sum_{n=N}^{N_{\max}}\left(w_3 n\boldsymbol{b}(n-N+1) + \|u_{s,h}(n)\|_{w_1}^2 + \|\boldsymbol{x}_{s,h}(n)\|_{w_2}^2\right), \tag{25}$$

where $\boldsymbol{b}(n-N+1)$ for $n = N,\ N+1,\ldots,N_{\max}$ are binary decision variables satisfying

$$\boldsymbol{b}(n-N+1) = \begin{cases} 1, & \text{for } T_f(k) = n \\ 0, & \text{otherwise.} \end{cases}$$

and

$$\sum_{n=N}^{N_{\max}} \boldsymbol{b}(n-N+1) = 1 \tag{26}$$

to ensure one arrival time step is selected. This selected arrival time is then a minimal arrival time over $[N, N_{\max}]$. If at time step k, the task is feasible within the preferable arrival time N, then N will be decided as the terminal time of the MIQP. Same constraints with $J_s^1(k)$ are imposed to $J_s^2(k)$ before $T_f(k)$, however, constraints are necessary to be relaxed after the selected arrival time. In addition, terminal constraint

$$s_{s,h}(T_f(k)) = s_{t_f} \tag{27}$$

is applied upon $T_f(k)$. We define the above logics as logic constraint C_1 which is modeled for $n_i = N,\ N+1,\ldots,N_{\max}$ as:

$$C_1 = \begin{cases} (23) \text{ and } (24), & \text{for } \sum_{n=N}^{n_i} \boldsymbol{b}(n-N+1) = 0 \\ s_{s,h}(n) = s_{t_f}, & \text{for } \sum_{n=N}^{n_i} \boldsymbol{b}(n-N+1) = 1. \end{cases} \tag{28}$$

A diagram illustrating the timing involved cost of J_s^2 is shown in Fig. 4. Therefore, the final MIQP problem formulated at the higher level is:

$$\boldsymbol{u}_{s,h}^*(k), \boldsymbol{b}^*(k) = \underset{\boldsymbol{u}_{s,h}, \boldsymbol{b}}{\operatorname{argmin}} J_s(k), \tag{29}$$

subject to, for $n = k, k+1,\ldots,N-1$,

(23) and (24),

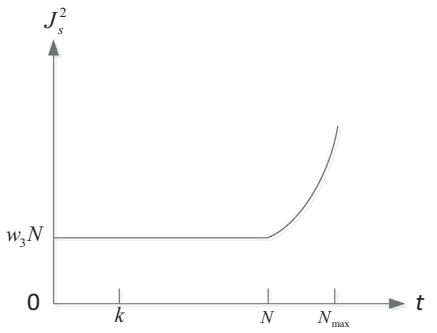


Fig. 4. Preferable arrival time and a maximum delay.

and for $n = N, N+1, \ldots, N_{\max}$

(26) and (28).

Remark 2. As can be observed from the above derivation, the length of the reference generated by the higher level is shortened by one each simulation step. But the MPC embedded low level requires an N_p-length reference $s^r(k+i|k)$, for $i = 1, 2, \ldots, N_p$ each step. When the current time is still distant with the scheduled arrival time, the generated reference might remain longer than N_p, however, upon arrival, this might cause problems. Therefore, we introduce extra N_p time step in addition to $N_{\max}$, and states $\boldsymbol{x}_s(n)$ over $n = N_{\max}+1, N_{\max}+2, \ldots, N_{\max}+N_p$ are then constrained to stay as the terminal state, i.e.,

$$s_{s,h}(n) = s_{t_f}. \tag{30}$$

Remark 3. Due to the simplicity of the double integrator dynamics, the computational complexity of the MIQP problem is still trackable even when the distance and time spans are reasonably large by solving it separately at a higher level. Directly integrating the higher level in online MPC optimizations is undesirable, because s dynamics will be coupled with predicted vessel dynamics over $k,\ k+1, \ldots, k+N_p$ so that it will always slow down to minimize tracking errors over $k,\ k+1, \ldots, k+N_p$ and lead to no forward motions of the vessel system. Besides, higher computational burdens for online MPC optimizations are also expected in this case.

4.2. Predictive switching logic

MPC can take into account situations in the future so that effective actions can be taken at an early stage to avoid undesirable system behaviors. This predictive feature of MPC is useful in our switching reference path segments to avoid overshoots which commonly exist in path following problems using LOS guidance laws. Based on path coordinate system models described in Section 2 and linearized prediction models derived in Section 3, we formulate a predictive switch logic in this subsection. Before proceeding, two relevant definitions are given first.

Definition 1. The position of a vessel is called in path coordinate system $\{p_j\}$ at time step k if $x_{p_j}(k)$, the along-track state in $\{p_j\}$, is not larger than the jth reference path length, i.e.,

$$x_{p_j}(k) \leqslant l_j. \tag{31}$$

Definition 2. For the vessel to track a geometric reference path j, three kinds of tracking errors are recognized and minimized in online MPC optimizations:

- cross-track error $y_{p_j}(k)$ the definition of which has been given in Section 2 and a vanish of $y_{p_j}(k)$ indicates a convergence to the reference path;
- along-track error $s_j(k) - s(k)$ where $s_j(k)$ is the total along-track distance the vessel has traveled, so its relationship with the along-track state $x_{p_j}(k)$ is:

$$s_j(k) = x_{p_j}(k) + \left(\sum_{jj=1}^{j} l_{jj} - l_j\right); \tag{32}$$

- and heading angle error ψ_{p_j}.

Gathering them in an error vector, we have

$$\boldsymbol{x}^e_{\mathrm{p}_j}(k) = \begin{bmatrix} s_j(k) - s_{\mathrm{s},1}(k) & y_{\mathrm{p}_j}(k) & \psi_{\mathrm{p}_j} \end{bmatrix}^{\mathrm{T}}. \tag{33}$$

The error vector $\boldsymbol{x}^e_{\mathrm{p}_j}$, which is coordinate system dependent, is minimized in MPC for tracking.

At time step k, considering the vessel is still in $\{p_j\}$, then initial states $\boldsymbol{x}_{\mathrm{p}_j}(k)$ can be obtained by a transformation of the current measured vessel states $\boldsymbol{x}(k)$ from $\{n\}$ to $\{p_j\}$ according to (8) and (9). Future system trajectories $\boldsymbol{x}_{\mathrm{p}_j}(k+i|k)$ for $i = 0, 1, \ldots, N_{\mathrm{p}}$ are then predicted in a linear way as (14) and (18). Note here that since the discretization and linearization theories are not applicable to discontinuous dynamics, all the predicted system trajectories as well as the seed trajectories for linearizations are still based on one model in $\{p_j\}$. If the to be optimized predicted trajectories are indeed all within $\{p_j\}$, e.g., when the vessel is far away from a switching waypoint, as Fig. 5a shows, minimization of $\boldsymbol{x}^e_{\mathrm{p}_j}$ in online MPC optimizations would then realize tracking path references. However, since MPC looks into the future over a prediction horizon, an initial state close to the switching waypoint would then result in predicted trajectories dispersed in both $\{p_j\}$ and $\{p_{j+1}\}$. In this case, minimizations of $\boldsymbol{x}^e_{\mathrm{p}_j}$ will result in overshoots as Fig. 5b shows. A transformation of coordinates from $\{p_j\}$ to $\{p_{j+1}\}$ is then necessary. Therefore, based on Definition 1, the following logic is introduced to realize a shift of coordinate system for predicted trajectories, and thus the tracking error $\boldsymbol{x}^e_{\mathrm{p}_j}$ so that overshoots are avoided, as Fig. 5c shows.

Define binary decision variable $\boldsymbol{b}_{\mathrm{p}}(k)$ as an $N_{\mathrm{p}} \times 1$ vector at time step k with

$$\boldsymbol{b}_{\mathrm{p}}(k+i|k) = \begin{cases} 1, & \text{for } x_{\mathrm{p}_j}(k+i|k) \leqslant l_j \\ 0, & \text{otherwise.} \end{cases} \tag{34}$$

When the vessel travels to $\{p_{j+1}\}$, i.e., when $x_{\mathrm{p}_j}(k+i|k) > l_j$, it is expected to track reference path $j+1$. This logic is expressed as logic constraint C_2 as:

$$C_2 = \begin{cases} \boldsymbol{x}^e_{\mathrm{p}_j}(k+i|k), & \text{for } \boldsymbol{b}_{\mathrm{p}}(k+i|k) = 1 \\ \boldsymbol{x}^e_{\mathrm{p}_{j\to j+1}}(k+i|k), & \text{for } \boldsymbol{b}_{\mathrm{p}}(k+i|k) = 0. \end{cases} \tag{35}$$

for $i = 1, 2, \ldots, N_{\mathrm{p}}$ and $\boldsymbol{x}^e_{\mathrm{p}_{j\to j+1}}(k+i|k)$ is the predicted tracking error w.r.t. reference path $j+1$ while the vessel is still in $\{p_j\}$, i.e., predicted states $\boldsymbol{x}_{\mathrm{p}_j}(k+i|k)$ are still derived in $\{p_j\}$. Then according to transformations from $\{p_j\}$ to $\{p_{j+1}\}$ as (6) and (7),

$$\boldsymbol{x}^e_{\mathrm{p}_{j\to j+1}}(k+i|k) = \begin{bmatrix} s_{j+1}(k+i|k) - s_{\mathrm{s},1}(k+i|k) & y_{\mathrm{p}_{j+1}}(k+i|k) & \psi_{\mathrm{p}_{j+1}}(k+i|k) \end{bmatrix}^{\mathrm{T}}, \tag{36}$$

where

$$s_{j+1}(k+i|k) = x_{\mathrm{p}_{j+1}}(k+i|k) + \left(\sum_{jj=1}^{j+1} l_{jj} - l_{j+1} \right), \tag{37}$$

and

$$\begin{bmatrix} x_{\mathrm{p}_{j+1}}(k+i|k) \\ y_{\mathrm{p}_{j+1}}(k+i|k) \end{bmatrix} = \begin{bmatrix} \cos(\psi_{j+1} - \psi_j) & \sin(\psi_{j+1} - \psi_j) \\ -\sin(\psi_{j+1} - \psi_j) & \cos(\psi_{j+1} - \psi_j) \end{bmatrix} \begin{bmatrix} x_{\mathrm{p}_j}(k+i|k) - l_j \\ y_{\mathrm{p}_j}(k+i|k) - 0 \end{bmatrix}, \tag{38}$$

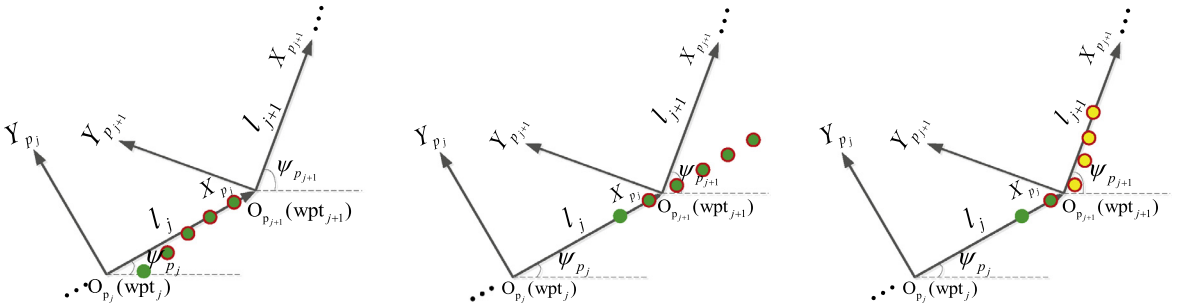
and

$$\psi_{\mathrm{p}_{j+1}}(k+i|k) = \psi(k+i|k) - \psi_{j+1}(k+i|k). \tag{39}$$

In this way, a solution to the binary variable $\boldsymbol{b}_{\mathrm{p}}(k+i|k)$ for $i = 1, 2, \ldots, N_{\mathrm{p}}$ will predictively and optimally determine the vessel's predicted position in coordinate system $\{p_j\}$ or $\{p_{j+1}\}$. Corresponding tracking errors are then minimized in online MPC optimizations and overshoots are expected to be avoided as in Fig. 5c.

4.3. Receding horizon control for PPF-ATA

This section describes the receding horizon control law for the proposed PPF-ATA problem. MPC online optimizations compute optimal control inputs based on approximated linearized prediction models (14) and (18). Vessel behaviors are then updated based on the first element of the optimal control input sequence. This process is then repeated until the vessel arrives at the destination for an ITT task. To achieve arrival time awareness and at the same time, still have smooth tracking performances, double integrator s dynamics are introduced for path parameterizations and generating timing-aware references over the prediction horizon by solving MIQPs. Moreover, overshoots are avoided in the proposed MPC framework during switching waypoints by optimizing switching logic related binary decision variables.



(a) Predicted trajectories are all within $\{p_j\}$

(b) Predicted trajectories are dispersed in $\{p_j\}$ and $\{p_{j+1}\}$ and optimized in $\{p_j\}$

(c) Predicted trajectories are dispersed and optimized in $\{p_j\}$ and $\{p_{j+1}\}$

Fig. 5. green dot–initial states; red circled dots–states predicted in $\{p_j\}$; red circled green dots–states optimized in $\{p_j\}$; red circled yellow dots–states optimized in $\{p_{j+1}\}$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

To achieve the requirements of a PPF-ATA problem, 4 terms are to be minimized in the cost function of online MPC optimizations:

1. Path tracking errors as defined in Section 4.2 over the prediction horizon.
2. Too large changes in control inputs which could lead to actuator damages.
3. Kinetic energy consumption which is formulated as $\frac{1}{2}\boldsymbol{v}^{\mathrm{T}}\boldsymbol{M}\boldsymbol{v}$ for a surface vessel, where $\boldsymbol{M}=\boldsymbol{M}_{RB}+\boldsymbol{M}_{A}$ is the mass matrix.
4. Differences between the lower level and higher level s dynamics which cause delays w.r.t a on-time schedule.

The online optimizations are then to minimize the cost function over incremental vessel actuator inputs $\Delta\boldsymbol{u}$, acceleration of lower level s dynamics $\boldsymbol{u}_{\mathrm{s,l}}$ and binary decision variable $\boldsymbol{b}_{\mathrm{p}}$ subject to system constraints. Therefore, for a vessel in $\{p_j\}$ at time step k, the following optimization problem is solved online:

$$\Delta\boldsymbol{u}^{*}(k),\boldsymbol{u}^{*}_{\mathrm{s,l}}(k),\boldsymbol{b}^{*}_{\mathrm{p}}(k)=\underset{\Delta\boldsymbol{u},\boldsymbol{u}_{\mathrm{s}},\boldsymbol{b}_{\mathrm{p}}}{\operatorname{argmin}}J(k), \tag{40}$$

where

$$J(k)=\sum_{i=0}^{N_{\mathrm{p}}-1}\left(\|\boldsymbol{x}^{e}_{\mathrm{p}_j}(k+i+1|k)\|^{2}_{\boldsymbol{w_4}}+\|\Delta\boldsymbol{u}(k+i|k)\|^{2}_{\boldsymbol{w_5}}+\|\boldsymbol{v}(k+i+1|k)\|^{2}_{\boldsymbol{w_6M/2}}+\|\boldsymbol{s}(k+i+1|k)-\boldsymbol{s}^{\mathrm{r}}(k+i+1|k)\|^{2}_{w_7}\right), \tag{41}$$

subject to, for $i=0,1,\ldots,N_{\mathrm{p}}-1$,

$$(18)\text{ and }(14), \tag{42a}$$
$$(34)\text{ and }(35), \tag{42b}$$
$$(20), \tag{42c}$$
$$\Delta\boldsymbol{x}_{\mathrm{p}_j}(k|k)=0, \tag{42d}$$
$$|\boldsymbol{u}(k+i|k)|\leqslant\boldsymbol{u}_{\max}, \tag{42e}$$
$$\boldsymbol{x}_{\mathrm{p}_j,\min}\leqslant\boldsymbol{x}_{\mathrm{p}_j}(k+i+1|k)\leqslant\boldsymbol{x}_{\mathrm{p}_j,\max}, \tag{42f}$$
$$x_{\mathrm{p}_j}(k+i+1|k)\leqslant\left(\mathrm{obs}_{x_{p_j},\min}-d_{\mathrm{s}}\right)+Mb_{\mathrm{obs},1}, \tag{42g}$$
$$-x_{\mathrm{p}_j}(k+i+1|k)\leqslant-\left(\mathrm{obs}_{x_{p_j},\max}+d_{\mathrm{s}}\right)+Mb_{\mathrm{obs},2}, \tag{42h}$$
$$y_{\mathrm{p}_j}(k+i+1|k)\leqslant\left(\mathrm{obs}_{y_{p_j},\min}-d_{\mathrm{s}}\right)+Mb_{\mathrm{obs},3}, \tag{42i}$$
$$-y_{\mathrm{p}_j}(k+i+1|k)\leqslant-\left(\mathrm{obs}_{y_{p_j},\max}+d_{\mathrm{s}}\right)+Mb_{\mathrm{obs},4}, \tag{42j}$$
$$\sum_{n=1}^{4}b_{\mathrm{obs},n}\leqslant 3\text{ and }b_{\mathrm{obs},n}\in\{0,1\}. \tag{42k}$$

where $\Delta\boldsymbol{u}^{*}(k)$ denote the sequence of optimal incremental control inputs solved at time step k, i.e., $\Delta\boldsymbol{u}^{*}(k)=\Delta\boldsymbol{u}^{*}(k+i|k)$ for $i=0,1,\ldots,N_{\mathrm{p}}-1$. The same goes for $\boldsymbol{u}^{*}_{\mathrm{s,l}}(k)$ and $\boldsymbol{b}^{*}_{p}(k)$. In $J(k)$, references for the lower level s dynamics, $\boldsymbol{s}^{\mathrm{r}}(k+i|k)$ over the prediction horizon are calculated by solving a MIQP problem before solving the online MPC optimization problem. Generally, the length of the calculated reference $\boldsymbol{s}^{\mathrm{r}}$ is longer than N_{p}, but we only feed $\boldsymbol{s}^{\mathrm{r}}(k+i+1|k)$ for $i=0,1,\ldots,N_{\mathrm{p}}-1$ to $J(k)$. Constraints (42a) are equality constraints of the approximated linearized prediction models of nonlinear path

coordinate system model (5). (42b) are the logic constraints for formulations of tracking errors in different reference path frames, as derived in Section 4.2; initial incremental state $\Delta \boldsymbol{x}_{\mathrm{p}_j}(k|k)$ is set to 0 as (42d) because $\boldsymbol{x}_{\mathrm{p}_j}(k|k) = \boldsymbol{x}^0_{\mathrm{p}_j}(k|k)$ and both of them are equal to $\boldsymbol{x}_{\mathrm{p}_j}(k)$ which is the current "measured"[4] state; system limitations on control inputs and states due to system physical limits on maximum actuator forces/moment and maximum speed, etc., are imposed by (42e) and (42f), respectively; obstacle avoidances for static obstacles which cause delays to a schedule are formulated as (42g)–(42k) where $\left(\mathrm{obs}_{x_{\mathrm{p}_j},\min}, \mathrm{obs}_{y_{\mathrm{p}_j},\min}\right)$ and $\left(\mathrm{obs}_{x_{\mathrm{p}_j},\max}, \mathrm{obs}_{y_{\mathrm{p}_j},\max}\right)$ are the coordinates in $\{p_j\}$ for the left-low and right-up corner of a rectangular obstacle, respectively; $b_{\mathrm{obs},n}$ and M are binary variables and a big value, respectively for an convex obstacle avoidance formulation (Schouwenaars et al., 2001,). Since the avoidance constraints are only applied at discrete time steps, a safety margin $d_{\mathrm{s}} = u_{\max} T_{\mathrm{s}} \big/ 2\sqrt{2}$ is implemented to avoid crossings in corners (Kuwata, 2007). Readers are referred to (Schouwenaars et al., 2001,) (Kuwata, 2007) for more details. Note here that obstacle avoidance constraints are imposed to the center of the vessel without considering specific vessel shapes, but we assume vessel sizes have been taken into account when obstacle areas are defined. So as long as the trajectory of vessel's center is outside obstacle areas, the vessel is safe.

At each time step k, two MIQPs need to be solved: one is the upper level timing-aware reference generation problem (29) and (28) and the other is the online MPC optimization problem (40)–(42). With reasonable problem size for one waterborne AGV, the two MIQPs can be solved efficiently by standard solvers. Each time a new optimization problem is formulated given the current new measurements; a sequence of optimal control inputs $\boldsymbol{u}^*(k) = \Delta \boldsymbol{u}^*(k) + \boldsymbol{u}^0(k)$ is calculated which will drive predicted system outputs close to set references to achieve design requirements. The first element of this optimal control sequence, i.e., $\boldsymbol{u}^*(k|k)$ is applied to the real system (3). Time is then shifted one step forward and the above procedures are repeated at the new time step to formulate a receding horizon law. Convergence to the reference path and timing aware of arrival at the destination in an economical way is thus guaranteed. System constraints are also well considered in online MPC optimizations. The overall algorithm for the problem of PPF-ATA is summarized in Algorithm 1.

Algorithm 1. Predictive path following with arrival time awareness (PPF-ATA)

1: Initialization at path coordinate system $j = 1$ and time step $k = 0$: $\boldsymbol{x}(0) = \boldsymbol{x}_0$, $\boldsymbol{x}_{\mathrm{s}}(0) = \boldsymbol{x}_{\mathrm{s}0}$ and $\boldsymbol{u}(0) = \boldsymbol{0}_{3\times N_{\mathrm{p}}}$;
2: Solve MIQP to obtain $\boldsymbol{x}^{\mathrm{r}}_{\mathrm{s}}(k+i)$ for $i = 1, 2, \ldots, N_{\mathrm{p}}$;
3: **while** $\boldsymbol{x}(k) \neq \boldsymbol{x}(t_{\mathrm{f}})$ **do**
4: **while** $x_{\mathrm{p}_j}(k) \leqslant l_j$ **do**
5: Measure current states $\boldsymbol{x}(k)$ and calculate $\boldsymbol{x}_{\mathrm{s,l}}(k)$;
6: Determine initial state $\boldsymbol{x}_{\mathrm{p}_j}(k)$ in $\{p_j\}$;
7: Set $\boldsymbol{x}^0_{\mathrm{p}_j}(k|k) = \boldsymbol{x}_{\mathrm{p}_j}(k)$ and $\boldsymbol{u}^0(k+i|k) = \boldsymbol{u}(k+i|k)$ for $i = 0, 1, \ldots, N_{\mathrm{p}} - 1$ and calculate seed trajectory $\left(\boldsymbol{x}^0_{\mathrm{p}_j}(k+i|k), \boldsymbol{u}^0(k+i|k)\right)$ for $i = 0, 1, \ldots, N_{\mathrm{p}} - 1$;
8: Approximate nonlinear model (5) about the seed trajectory and predict system trajectories over N_{p} by (14) and (18);
9: Set $\boldsymbol{x}_{\mathrm{s,l}}(k|k) = \boldsymbol{x}_{\mathrm{s,l}}(k)$ and predict s dynamic trajectories over N_{p} by (20);
10: Solve optimization problem (40) to determine $\boldsymbol{u}^*(k)$, $\boldsymbol{u}^*_{\mathrm{s}}(k)$;
11: Apply the first element $\boldsymbol{u}^*(k|k)$ to vessel dynamics (3) and the first element $\boldsymbol{u}^*_{\mathrm{s,l}}(k|k)$ to s dynamics (19);
12: $k = k + 1$;
13: Shift obtained optimal control input sequences $\boldsymbol{u}^*(k)$ according to (12) and (13) to get $\boldsymbol{u}(k+1)$;
14: **end while**
15: $j = j + 1$;
16: **end while**

The condition in the outer *while* loop $\boldsymbol{x}(k) \neq \boldsymbol{x}(t_{\mathrm{f}})$ means that a vessel has not arrived at the final destination and

$$\boldsymbol{x}(t_{\mathrm{f}}) = \begin{bmatrix} \boldsymbol{\eta}^{\mathrm{T}}_{t_{\mathrm{f}}} & \boldsymbol{v}^{\mathrm{T}}_{t_{\mathrm{f}}} \end{bmatrix}^{\mathrm{T}}, \tag{43}$$

where $\boldsymbol{\eta}_{t_{\mathrm{f}}}$ is the final pose dependent on reference path information and $\boldsymbol{v}_{t_{\mathrm{f}}} = [0 \quad 0 \quad 0]^{\mathrm{T}}$. The PPF-ATA controller based on Algorithm 1 designed for waterborne AGVs is shown in Fig. 6.

5. Simulation experiments

In this section, we present results from two simulation experiments to illustrate how the PPF-ATA controller works and to demonstrate its potential for ITT. For the first simulation, the controller is given a feasible ITT task, which means the scheduled preferable arrival time can be achieved by the system. In the second simulation, however, an infeasible ITT task is set

[4] $\boldsymbol{x}_{\mathrm{p}_j}(k)$ is not directly measurable but transformed from $\boldsymbol{x}(k)$ by (8) and (9).

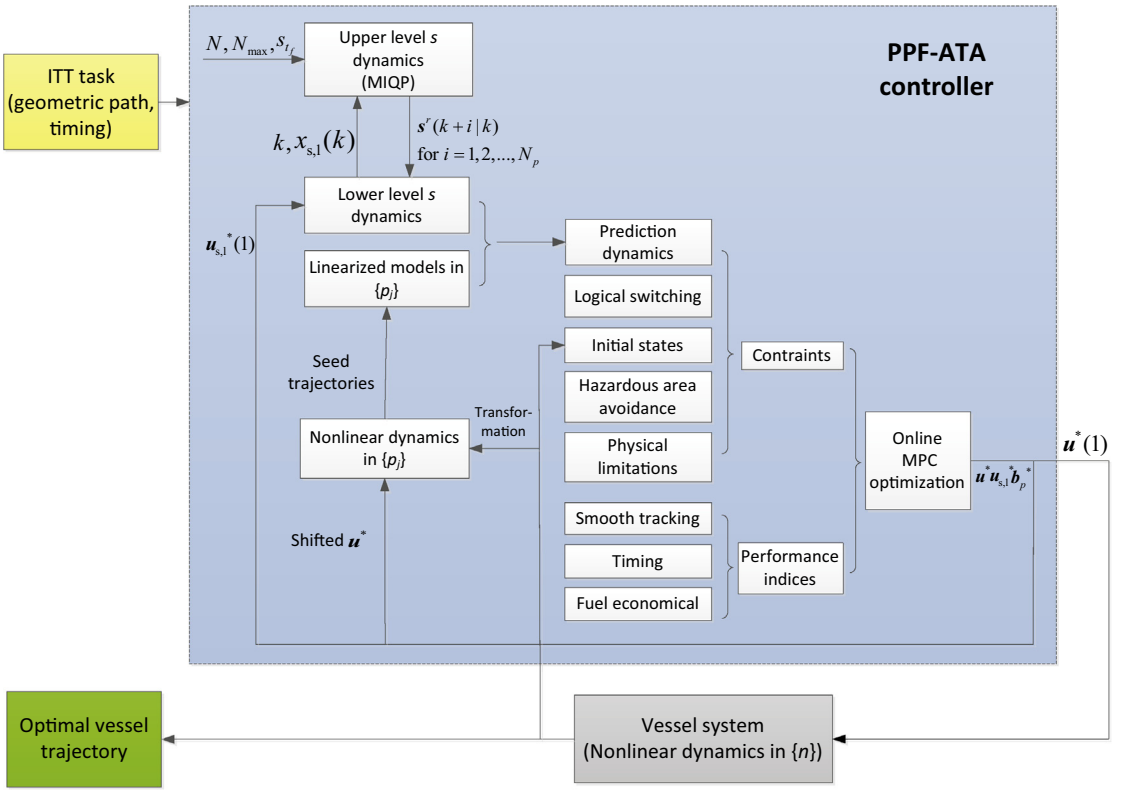


Fig. 6. PPF-ATA controller for waterborne AGVs.

where the preferable arrival time cannot be met even if the vessel would sail at the maximum speed all the time without any obstacles. In the latter case, we show how the vessel achieves the task with a minimum delay w.r.t. the preferable arrival time.

For both simulations, we set an ITT task from APM terminal to Euromax Terminal in the Port of Rotterdam, as shown in Fig. 7. It could be interesting to distribute ITT over waterborne AGVs in this scenario because these two terminals have not been connected by land so far, and even connected, the distance by land is much longer than by water. The reference path consists of several straight-line segments. Since our simulations are based on a $1:70$ small scaled vessel model, actual length and time quantities are also scaled according to Froude scaling law: $1:70$ for length (m) and $1:\sqrt{70}$ for time (s). The reference path information[5] for both of the two ITT tasks is then given as Table 2.

Reference path details including l_j and ψ_j can then be calculated from the waypoints given in Table 2. Considering that in a real situation, the vessel will not stop with a heading angle decided by Intermediate waypoint 2 and Euromax Terminal, but a heading angle required by berthing at the terminal, see the red circle in Fig. 7. Intermediate waypoint 3 is therefore introduced to produce the final reference heading angle. Therefore, final pose in (43) is given as $\boldsymbol{\eta}_{t_f} = [\,-4.7814 \quad 25.5214 \quad 3.0367\,]^{\mathrm{T}}$. We will show this berthing behavior can also be well achieved by our PPF-ATA controller. In addition, two static obstacles are placed along the path to imitate a situation where unexpected delays happen. One of them is placed half way of the first line segment, and the other half way of the third line segment.

Two simulations also share same MPC controller settings with a prediction horizon $N_{\mathrm{p}} = 20$. Weight parameters are given as:

$$w_1 = 1,\ \boldsymbol{w}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix},\ w_3 = 1000,$$
$$\boldsymbol{w}_4 = \begin{bmatrix} 1000 & 0 & 0 \\ 0 & 1000 & 0 \\ 0 & 0 & 100 \end{bmatrix},\ \boldsymbol{w}_5 = \boldsymbol{I}_{3\times 3},\ \boldsymbol{w}_6 = \boldsymbol{I}_{3\times 3},\ w_7 = 100. \tag{44}$$

The vessel is initially positioned at $(1,0)$ with $\psi = \pi$ and zero velocity, namely $\boldsymbol{x}_0 = [\,1 \quad 0 \quad \pi \quad 0 \quad 0 \quad 0\,]^{\mathrm{T}}$. System sampling time $T_s = 1$ s. System constraints are based on the actual limitations of the vessel model we use (Skjetne, 2005):

[5] The positions in latitude/longitude are obtained from Google Earth and then converted to inertial frame coordinates with APM Terminal as the origin.

Fig. 7. APM terminal and Euromax Terminal at Maasvlakte 2 in the Port of Rotterdam from Google Earth (Rotterdam, 2013).

Table 2
ITT scenario for simulation.

	Lat./Lon.	(x_n, y_n) (m)	Scaled (x_n, y_n) (m)
APM terminal	(51.957812°, 4.041716°)	(0, 0)	(0, 0)
Intermediate wpt. 1	(51.961449°, 4.053324°)	(798.0, 404.7)	(11.4000, 5.7814)
Intermediate wpt. 2	(51.965470°, 4.053777°)	(829.1, 852.1)	(11.8443, 12.1729)
Intermediate wpt. 3	(51.973374°, 4.038988°)	(−187.5, 1731.5)	(−2.6786, 24.7357)
Euromax terminal	(51.973868°, 4.036848°)	(−334.7, 1786.5)	(−4.7814, 25.5214)

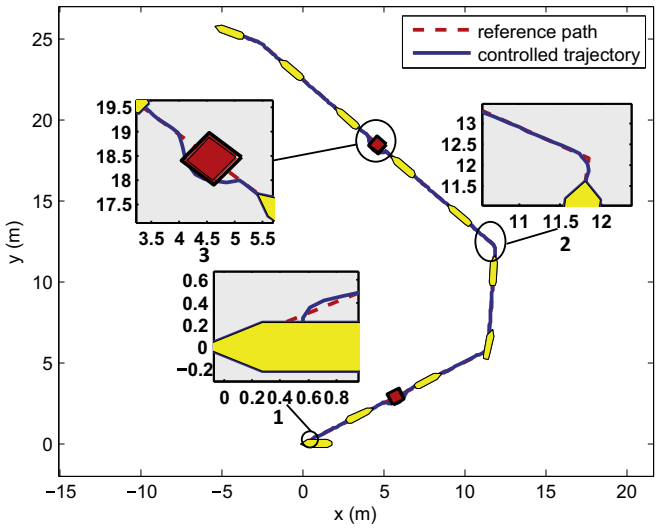


Fig. 8. Tracking performance of task 1.

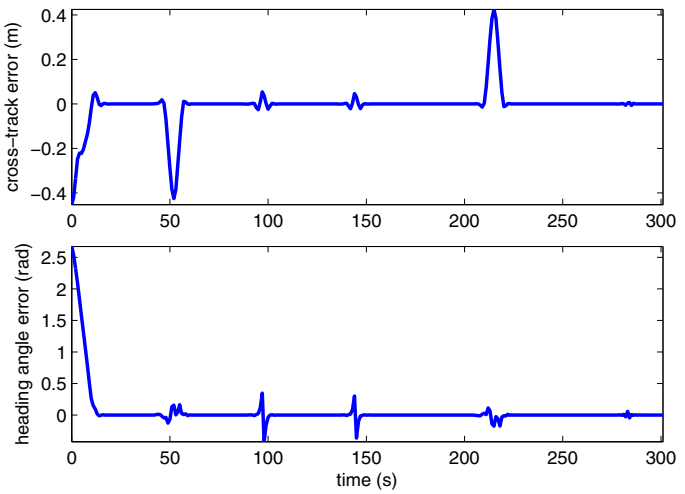


Fig. 9. Cross-track and heading angle errors of task 1.

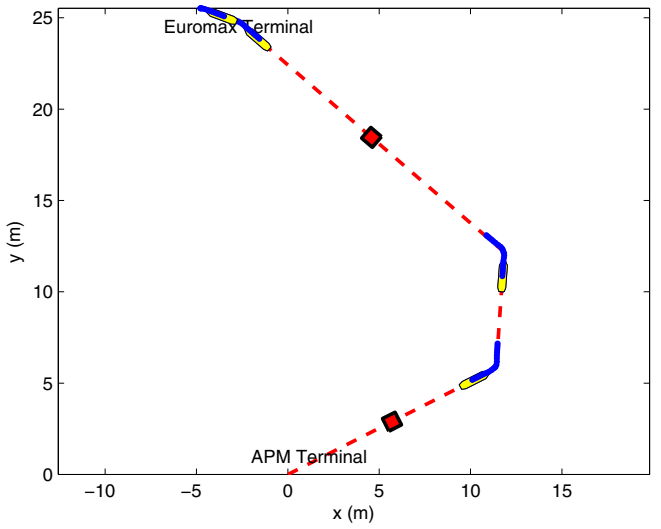


Fig. 10. Predicted vessel trajectories over the prediction horizon at one time step during switching of task 1.

$$\begin{bmatrix} 0 \\ -0.1 \\ -15\pi/180 \end{bmatrix} \leqslant \boldsymbol{v} \leqslant \begin{bmatrix} 0.2 \\ 0.1 \\ 15\pi/180 \end{bmatrix}, \text{ and } |\boldsymbol{\tau}_{\max}| = [\,2 \quad 2 \quad 1.5\,]^{\mathrm{T}},$$

which can all be incorporated in MPC in a systematic way.

Algorithms in this article are implemented using YALMIP (version 20131002) (Lofberg, 2004) in MATLAB 2011b (MATLAB, 2011). Optimization problems are solved by Gurobi (version 5.6 academic) (Gurobi Optimization, 2012). All the simulations are run on a platform with Intel (R) Core (TM) i5-3470 CPU @3.20 GHz.

5.1. Simulation experiment 1: feasible ITT task

The total reference path length after scaling is 48.6906 m and the preferable arrival time at destination is 300 s after departure from the origin. Therefore, if no unexpected events happen, an average speed of 0.1355 m/s should be attained, which is well in the maximum speed range of the vessel system, and thus is deemed as a feasible ITT task. However, considering the vessel cannot sail at this average speed all the time and delays might still happen due to unexpected events in reality, the higher level MIQP problem is solved based on an acceptable delay tolerance of 20 s. However, delay within the time-window of $[300, 320]$ are penalized in the MIQP problem. Hence, if the ITT task can be accomplished by

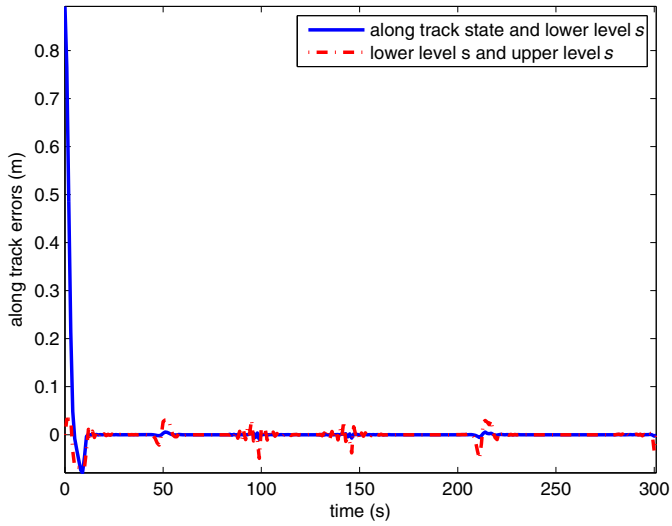


Fig. 11. Along-track errors of task 1.

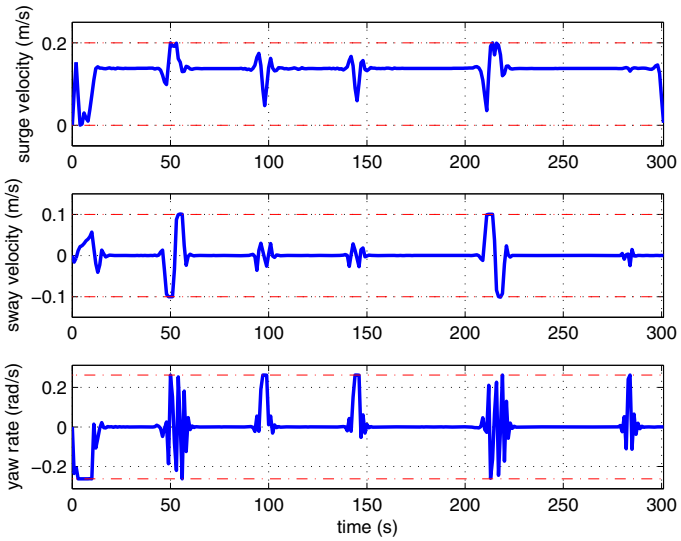


Fig. 12. Vessel surge, sway velocities and yaw rate of task 1.

the preferable arrival time, i.e., 300 s, the vessel is expected to arrive on time at 300 s. The first simulation is then run based on this task in order to achieve smooth path tracking with arrival time awareness in an economical way.

5.1.1. Path tracking performance

Path tracking performance of our PPF-ATA controller is illustrated by smooth convergence to reference paths when there are off-sets and small deviations when the vessel is on track. Besides, overshoots during switching of reference line segments are well avoided, which also demonstrates the controller's capability of path following.

Fig. 8 shows how the vessel accurately tracks the reference path. Yellow heptagons represent controlled vessels which are plotted according to vessel poses at certain time intervals. Fig. 9 further illustrates the path reference tracking performances by showing cross-track and path heading angle errors which are the second and third term in the error vector (33), respectively. Large errors are observed for both cross-track and heading at the beginning and also around obstacle areas because there is an initial offset and obstacle avoidance are implemented as hard constraints to guarantee safety. Other relatively smaller deviations in Fig. 9 are due to switches at the three intermediate waypoints.

The three boxes in Fig. 8 along the path are zoom-ins of vessel behaviors at starting point, switching at intermediate waypoint 2 and around the second obstacle, respectively. In boxes 1 and 3, it can be observed that the vessel is able to converge to the reference path smoothly with an initial offset or after a necessary offset to avoid obstacles. This is because the lower

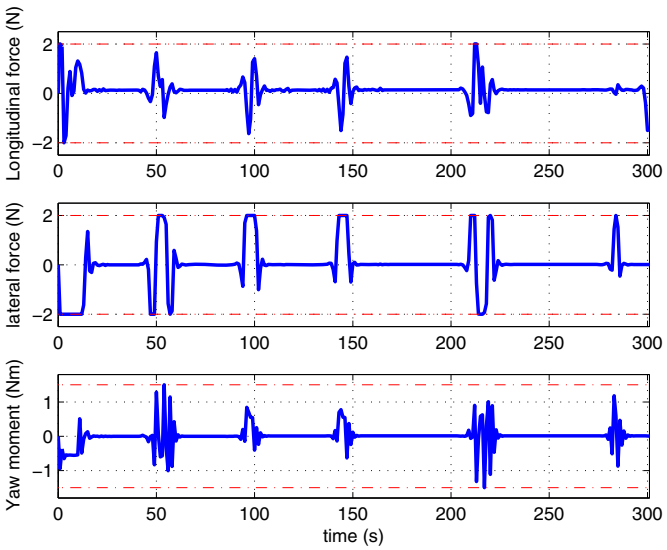


Fig. 13. Vessel surge, sway forces and yaw moment of task 1.

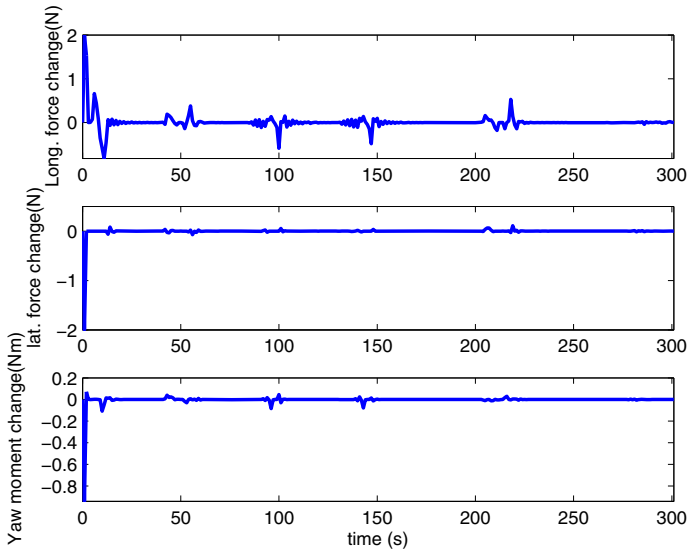


Fig. 14. Vessel surge, sway forces and yaw moment changes of task 1.

level *s* dynamics will always "slow down" to "wait for" the vessel if the vessel is in a situation with low speed, e.g., at the starting point, avoiding an obstacle, etc.

During switching of the reference line segments, as shown in the second box, the controlled vessel trajectory can also match the reference path very well with negligible deviations and almost no overshoots. This is because MPC can in principle take future available information into account and thus respond at an early stage. In our PPF-ATA controller, a coordinate transformation to the new path coordinate system when the logic switch (34) and (35) are satisfied will guarantee always right future reference information is used. In Fig. 10, predicted vessel trajectories over the prediction horizon (20 time steps into the future) at one time step near switching waypoints are plotted. If there are no model mismatches between the real system model and the model used for prediction, and if there are no external disturbances, the future system trajectories will be exactly like the one predicted at the current time step, which means the real vessel trajectory will also switch successfully. To ease online computational burden, we utilize approximated Jacobian linearized models for prediction, which inevitably will result in model mismatches. However, the successive linearization framework by conducting the linearization of the nonlinear dynamic system about a shifted optimal trajectory from a previous step has significantly reduced linearization errors. Furthermore, MPC has a closed-loop feedback control structure so that it can calculate new optimal control inputs

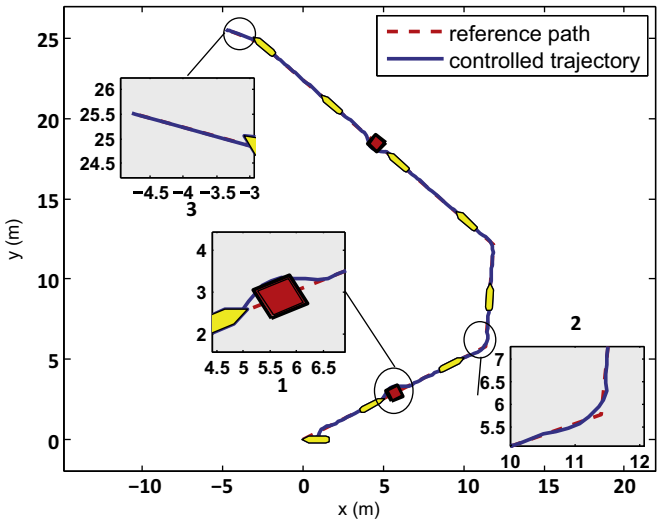


Fig. 15. Tracking performances of task 2.

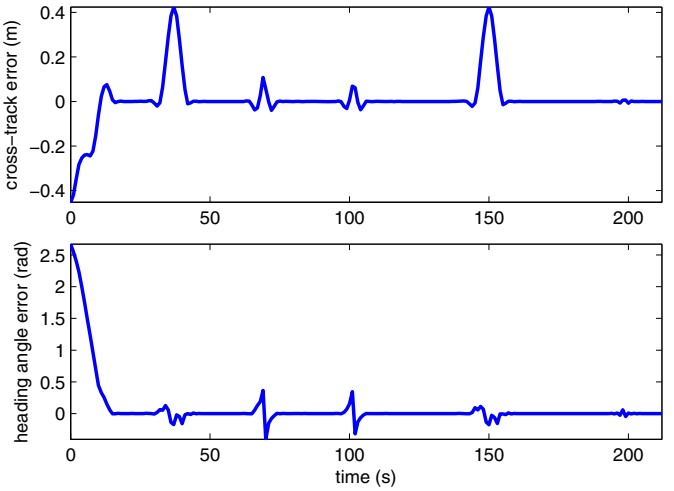


Fig. 16. Cross-track and heading angle errors of task 2.

based on new measured system states in a receding horizon way, and thus can eliminate adverse effects caused by linearization errors or other disturbances to a certain extent. Therefore, real vessel trajectories are also expected to have a successful switch like the predicted trajectory. Box 2 in Fig. 8 confirms this.

5.1.2. Arrival time awareness

The "slow down" of lower level s dynamics for smooth path tracking causes delays. However, the delays can be compensated after the vessel is not so "lagging-behind" by minimizing the error between lower and higher level s dynamics. The higher level s dynamics consider current new distance-to-go and time-to-go such that timing aware references are generated. Fig. 11 illustrates this. During starting time, time around 50 s and 210 s when the first and second obstacle avoidances happen, respectively, both along-track errors and lower-/higher level tracking errors see some fluctuations, but both of them return to an approximate 0 afterwards. Moreover, the vessel arrives at the destination at $t = 301$ s with 1 s delay which is 0.33% of the total time and corresponds to 8.37 s for a real scale vessel.

5.1.3. Energy consumption and system constraints

The objectives of good path following performance and arrival time awareness are achieved in an energy economical way within system limits. In Fig. 12, system velocities all maintain almost constant except for fluctuations at initial, obstacle and reference switching points. Since all the online MPC optimization problems are successfully solved, we believe the velocities

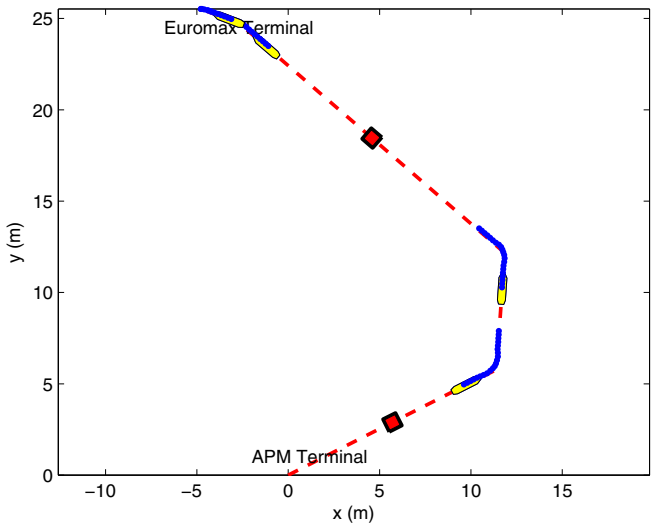


Fig. 17. Predicted vessel trajectories over the prediction horizon at one time step during switching of task 2.

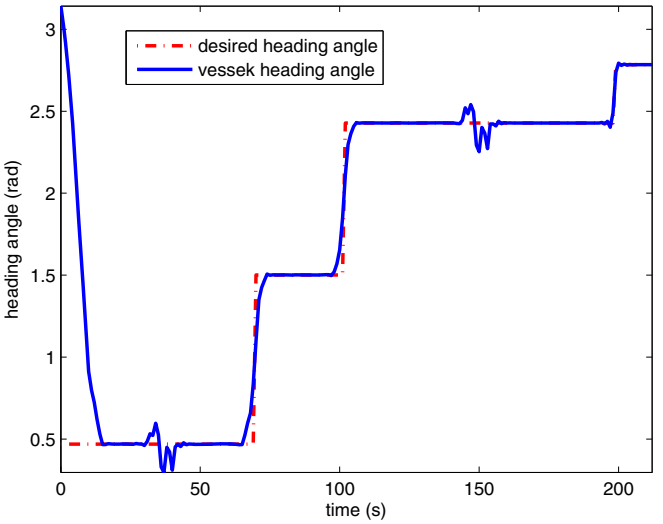


Fig. 18. Vessel heading angles of task 2.

are optimal values in the feasible region defined by system constraints. More on energy consumptions of the two experiments are presented in Section 5.2.

System physical constraints are also well satisfied in our scheme. Actuator inputs are shown in Fig. 13. Same as in Fig. 12, all the parameters are within the system limitations. To reduce actuator weariness, small control input changes are also desirable. Fig. 14 shows that except for the beginning and obstacle avoidance areas, control input changes are small.

5.2. Simulation experiment 2: infeasible ITT task

In this simulation, the vessel also needs follow the scaled reference path from APM Terminal to Euromax Terminal with a total of 40.6365 m but the scheduled preferable arrival time is set to be only 200 s after departure. Therefore, even if no unexpected events happen and the vessel is right on the path with a heading angle tangent to the path, the vessel still needs to sail at an average speed of 0.2032 m/s all the time. However, the vessel has a maximum surge speed of 0.2 m/s let alone the effects of current, off-track positions and zero velocities as initial states and unexpected events as obstacle avoidance, etc. So, this ITT task is defined as infeasible. But similarly as in Experiment 1, we append 20 s as an acceptable maximum arrival time, which results in an average speed of 0.1847 m/s. We assume despite of all the factors prone to lead to delays, the vessel is able to arrive at the destination by 220 s after departure. Simulation results below illustrate how our PPF-ATA controller also works well in this scenario to achieve smooth path tracking, arrival time awareness and energy efficiency.

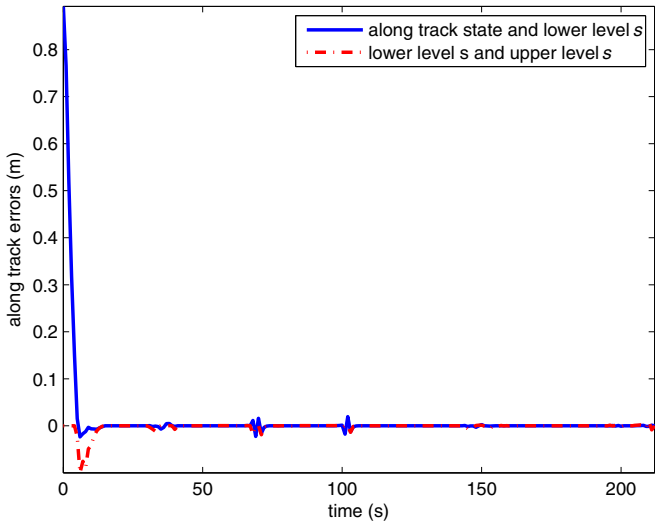


Fig. 19. Along-track errors of task 2.

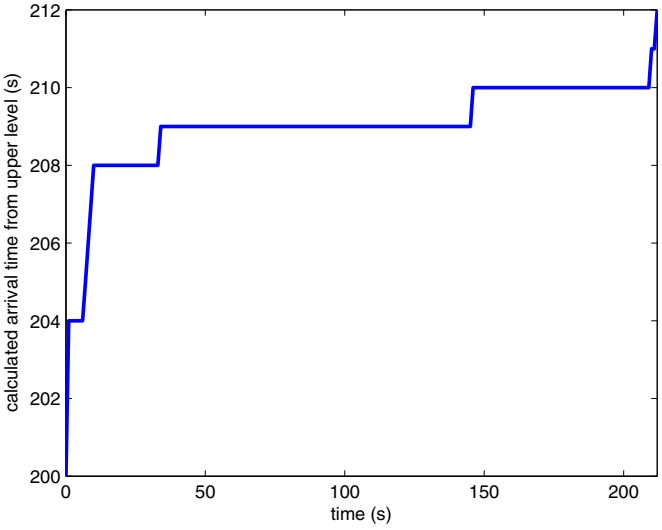


Fig. 20. Calculated arrival time by higher level of task 2.

5.2.1. Path tracking performance

Similarly as in Experiment 1, accurately tracking of the reference path is observed in this case as Fig. 15 shows. Zoom-ins of vessel behaviors around the first obstacle, switching at intermediate waypoint 1, and during the final destination area are shown as the three boxes, respectively. Again, smooth tracking and convergence to the reference path is achieved including areas around the starting point, obstacle and during switches. Fig. 16 further illustrate the path convergence performances by showing cross-track and heading angle errors along time. Relatively obvious deviations in both sub-figures of Fig. 16 are due to the initial offset, obstacles and switches at the three intermediate waypoints.

The second box demonstrate the switching of reference paths has been successful with almost no overshoots. But compared with the switch box in Fig. 8, larger deviations are observed, which can also be observed by comparing the errors caused by switches in Figs. 9 and 16. This is because when the arrival time is set shorter to 200 s, vessel needs to sail at a higher speed, which then leads to larger errors. Predicted vessel trajectories over the prediction horizon (20 time steps into the future) at one time step near switching waypoints for this ITT task are shown as Fig. 17.

In the third box, vessel trajectories can also well follow the last line segment which has been added for a berthing behavior. Fig. 18 further shows the heading angle evolutions which illustrate that the vessel stops at the destination terminal with a berthing angle.

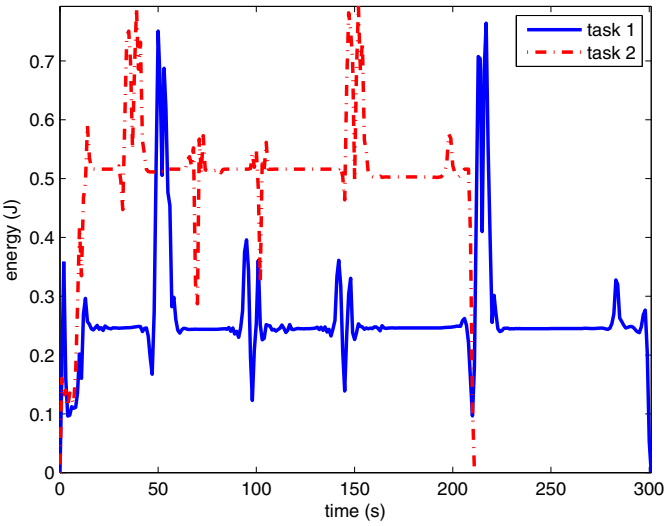


Fig. 21. Comparison of consumed energy.

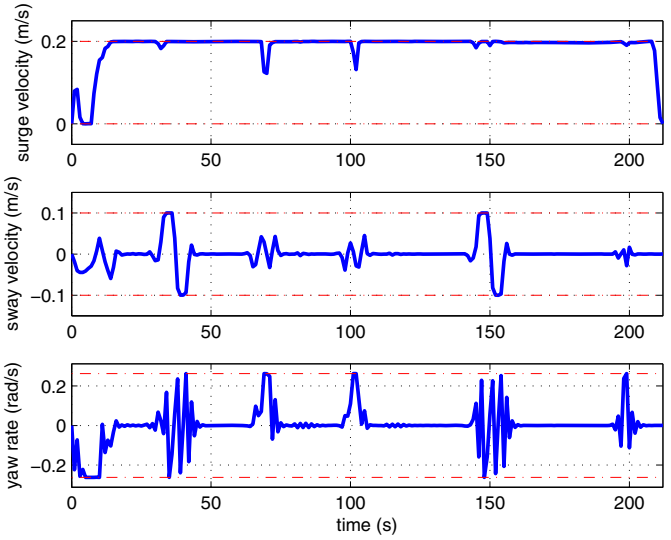


Fig. 22. Vessel surge, sway velocities and yaw rate of task 2.

5.2.2. Arrival time awareness

Fig. 19 shows along-track errors in this task. Again, both along-track errors and lower-/higher level tracking errors see some fluctuations during starting, obstacle and switching areas, but both of them return to an approximate zero afterwards. But compared to Fig. 11, the times when fluctuations happen due to obstacles and switchings are earlier. This is because in Experiment 2, the vessel is sailing at a higher speed.

However, in this experiment, the vessel is still able to meet the timing requirement and arrives at the destination at $t = 212$ s which is within the higher bound of the time window, i.e., 220 s. Fig. 20 shows how the arrival times calculated by higher level MIQP changes every time an delay event, e.g., obstacles, switches, etc., happens.

5.2.3. Energy consumption and system constraints

The total kinetic energy consumption calculated according to Section 4.3 for task 1 and task 2 is 78.4683 J and 106.7409 J, respectively. Again, since we are solving repetitive constrained optimization problems which are all successfully solved, it is sufficient to conclude that the energy consumption is optimal[6] in a sense that they are the smallest within the system

[6] Optimization problems are formulated based on successively linearized prediction models, therefore, being optimal here can be seen as being approximately optimal for the original nonlinear system.

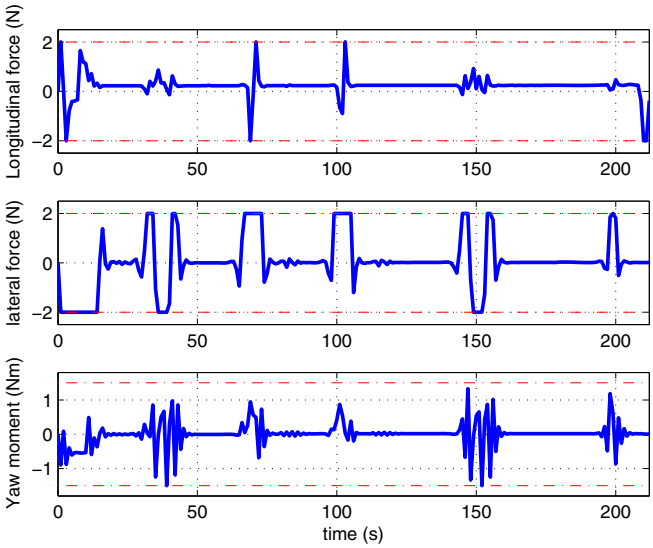


Fig. 23. Vessel surge, sway forces and yaw moment of task 2.

constraints. Although for the second ITT task, the cumulative time is shorter, it still has a much larger total energy consumption. Comparisons of the time-wise energy consumption of them are presented as Fig. 21. It is then clear that the vessel in the second task is consuming higher energies all the time so that it can fulfill the ITT task on time.

Evolutions of velocities and actuator forces for this experiment are shown as Figs. 22 and 23, respectively. Again, all the parameters are within the system limits.

6. Conclusions and future research

A predictive path following with arrival time awareness (PPF-ATA) controller for waterborne AGVs with application to Inter Terminal Transport (ITT) is proposed. Geometric reference paths are given as connected straight-line segments from which minimal derivations are expected. To minimize the "non-performance", timing requirements are also imposed. Economical operations within system constraints are important for logistics oriented applications. We show these conflicting objectives are achieved in a systematic way by our PPF-ATA controller based on modeling in connected path coordinate systems and a receding horizon control law. Vessel models in path coordinate systems facilitate the formulation of tracking errors. Overshoots are avoided during switching reference paths by taking advantage of the predictive feature of Model Predictive Control (MPC) and an along-track state involved switching logic. In this logic, coordinate transformations of predicted trajectories to a new path coordinate system also ensure a continuous dynamic model in one path coordinate system can be utilized, which further guarantees a successive linearization framework of MPC applicable. Smooth tracking and timing are then guaranteed by a two-level double integrator scheme. The lower level is embedded in online MPC optimization problems and the higher level solves varying horizon MIQP problems considering current distance-to-go and time-to-go to generate timing-aware references over the prediction horizon of MPC. In such a way, time aware of arrival, i.e., a preferable time of arrival when an ITT task is feasible and a minimal time delay w.r.t. the preferable time of arrival otherwise is accomplished. To demonstrate the effectiveness of the proposed modeling and control framework for waterborne AGVs, industrially relevant ITT scenarios with reference paths connecting APM Terminal and Euromax Terminal in the Port of Rotterdam are set. Two groups of ITT tasks are given, one being feasible and the other not. Simulation results illustrate waterborne AGVs with our PPF-ATA controller can fulfill both of the tasks successfully while meet system design criteria very well. Future research will extend and apply the controller of PPF-ATA for single waterborne AGV to multiple waterborne AGVs working cooperatively for ITT. Distributed MPC methodologies are to be explored for this scheme.

Acknowledgements

This research is supported by the China Scholarship Council under Grant 201206950021 and the VENI project "Intelligent multi-agent control for flexible coordination of transport hubs" (Project 11210) of the Dutch Technology Foundation STW, a subdivision of the Netherlands Organization for Scientific Research (NWO).

References

Do, K., 2010. Practical control of underactuated ships. Ocean Eng. 37 (13), 1111–1119.

Duinkerken, M.B., Dekker, R., Kurstjens, S.T., Ottjes, J.A., Dellaert, N.P., 2007. Comparing transportation systems for inter-terminal transport at the maasvlakte container terminals. In: Kap Hwan Kim, H.-O.G. (Ed.), Container Terminals and Cargo Systems. Springer, Berlin, Germany, pp. 37–61.

Fossen, T.I., 2011. Handbook of Marine Craft Hydrodynamics and Motion Control. John Wiley and Sons Ltd., West Sussex, UK.

Fossen, T.I., Breivik, M., Skjetne, R., 2003. Line-of-sight path following of underactuated marine craft. In: Proceedings of the 6th IFAC on Manoeuvring and Control of Marine Craft, Girona, Spain, pp. 244–249.

Gurobi Optimization, 2012. Gurobi optimizer reference manual. <http://www.gurobi.com>. Accessed: 2015-10-08.

Kirk, D.E., 2012. Optimal Control Theory: An Introduction. Courier Dover Publications, New York, United States.

Kouvaritakis, B., Cannon, M., Rossiter, J., 1999. Non-linear model based predictive control. Int. J. Control 72 (10), 919–928.

Kuwata, Y., 2007. Trajectory Planning for Unmanned Vehicles Using Robust Receding Horizon Control. Ph.D. Thesis, Massachusetts Institute of Technology, Massachusetts, United States.

Li, S., Zheng, H., Negenborn, R.R., Lodewijks, G., 2015. Coordination for efficient transport over water. In: Proceedings of IEEE 19th International Conference on Computer Supported Cooperative Work in Design. IEEE, Calabria, Italy, pp. 389–394.

Li, L., Negenborn, R.R., De Schutter, B., 2015. Intermodal freight transport planning – a receding horizon control approach. Transp. Res. C: Emerg. Technol. 60, 77–95.

Lin, S., De Schutter, B., Xi, Y., Hellendoorn, H., 2012. Efficient network-wide model-based predictive control for urban traffic networks. Transp. Res. C: Emerg. Technol. 24, 122–140.

Lofberg, J., 2004. YALMIP: a toolbox for modeling and optimization in matlab. In: Proceedings of 2004 IEEE International Symposium on Computer Aided Control Systems Design, Taipei, Taiwan, pp. 284–289.

MATLAB, 2011. Version 7.13 (R2011b), The MathWorks Inc., Natick, Massachusetts.

Mayne, D.Q., Rawlings, J.B., Rao, C.V., Scokaert, P.O.M., 2000. Constrained model predictive control: stability and optimality. Automatica 36 (6), 789–814.

Minorski, N., 1922. Directional stability of automatically steered bodies. J. Am. Soc. Nav. Eng. 42 (2), 280–309.

Oh, S., Sun, J., 2010. Path following of underactuated marine surface vessels using line-of-sight based model predictive control. Ocean Eng. 37 (2), 289–295.

Ohtsuka, T., 2004. A continuation/GMRES method for fast computation of nonlinear receding horizon control. Automatica 40 (4), 563–574.

Port of Rotterdam Authority, December 2011. Port Vision 2030. <http://www.portofrotterdam.com/en/Port/port-in-general/port-vision-2030/Pages/default.aspx>, Accessed: 2014-11-13.

Rotterdam, Maasvlakte, August 7, 2013. 51.962398° N and 4.056800° E, Google Earth, Accessed: 2014-10-17.

Schouwenaars, T., De Moor, B., Feron, E., How, J., 2001. Mixed integer programming for multi-vehicle path planning. In: Proceedings of 6th European Control Conference, Seminário de Vilar, Porto, Portugal, pp. 2603–2608.

Skjetne, R., 2005. The Maneuvering Problem. Ph.D. Thesis, Norwegian University of Science and Technology, Trondheim, Norway.

Skjetne, R., Fossen, T.I., Kokotović, P.V., 2005. Adaptive maneuvering with experiments for a model ship in a marine control laboratory. Automatica 41 (2), 289–298.

SNAME: The Society of Naval Architects and Marine Engineers, 1952. Nomenclature for treating the motion of a submerged body through a fluid. Techn. Res. Bull., 1–5.

Tierney, K., Voß, S., Stahlbock, R., 2014. A mathematical model of inter-terminal transportation. Eur. J. Oper. Res. 235 (2), 448–460.

Xin, J., Negenborn, R.R., Lodewijks, G., 2014. Energy-aware control for automated container terminals using integrated flow shop scheduling and optimal control. Transp. Res. C: Emerg. Technol. 44, 214–230.

Xin, J., Negenborn, R.R., Corman, F., Lodewijks, G., 2015. Control of interacting machines in automated container terminals using a sequential planning approach for collision avoidance. Transp. Res. C: Emerg. Technol. 60, 377–396.

Yu, S., Li, X., Chen, H., Allgöwer, F., 2015. Nonlinear model predictive control for path following problems. Int. J. Robust Nonlinear Control 25 (8), 1168–1182.

Zheng, H., Negenborn, R.R., Lodewijks, G., 2013. Survey of approaches for improving the intelligence of marine surface vehicles. In: Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems. IEEE, Den Hague, The Netherlands, pp. 1217–1223.