Social Trajectory Planning for Urban Autonomous Surface Vessels

Park, Shinkyu; Cap, Michal; Alonso-Mora, Javier; Ratti, Carlo; Rus, Daniela

DOI
10.1109/TRO.2020.3031250

Publication date
2021

Document Version
Final published version

Published in
IEEE Transactions on Robotics

Social Trajectory Planning for Urban Autonomous Surface Vessels

Shinkyu Park, *Member, IEEE*, Michal Cap, Javier Alonso-Mora, *Senior Member, IEEE*, Carlo Ratti, and Daniela Rus, *Fellow, IEEE*

***Abstract*—In this article, we propose a trajectory planning algorithm that enables autonomous surface vessels to perform *socially compliant* navigation in a city's canal. The key idea behind the proposed algorithm is to adopt an optimal control formulation in which the deviation of movements of the autonomous vessel from nominal movements of human-operated vessels is penalized. Consequently, given a pair of origin and destination points, it finds vessel trajectories that resemble those of human-operated vessels. To formulate this, we adopt kernel density estimation (KDE) to build a nominal movement model of human-operated vessels from a prerecorded trajectory dataset, and use a Kullback–Leibler control cost to measure the deviation of the autonomous vessel's movements from the model. We establish an analogy between our trajectory planning approach and the maximum entropy inverse reinforcement learning (MaxEntIRL) approach to explain how our approach can *learn* the navigation behavior of human-operated vessels. On the other hand, we distinguish our approach from the MaxEntIRL approach in that it does not require well-defined bases, often referred to as features, to construct its cost function as required in many of inverse reinforcement learning approaches in the trajectory planning context. Through experiments using a dataset of vessel trajectories collected from the automatic identification system, we demonstrate that the trajectories generated by our approach resemble those of human-operated vessels and that using them for canal navigation is beneficial in reducing head-on encounters between vessels and improving navigation safety.**

***Index Terms*—Autonomous vehicle navigation, learning from demonstration, marine robotics, motion and path planning.**

I. Introduction

MANY cities are seeking sustainable ways to transform their waterways to improve the public transportation capacity [1]. One example is the Roboat project [2], which aims at developing a fleet of autonomous surface vessels that can transport people and provide delivery and trash removal services through the canal network in the city of Amsterdam. Fig. 1 illustrates use cases of the Roboat project.

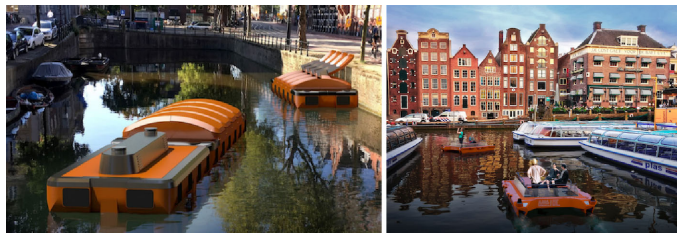


Fig. 1. Roboat: Robotic boats designed to serve as a platform for trash removal and as a new means of transportation in Amsterdam canal.

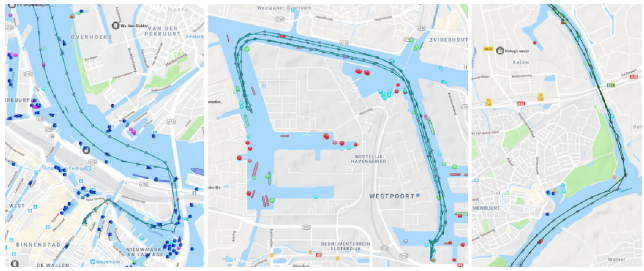


Fig. 2. Examples of vessel trajectories in Dutch canals. We observe that each trajectory typically lies on the right-hand side of the waterway. However, its exact distance from the canal boundary or from opposite trajectories varies from location to location, which is a potential cause of difficulties in designing a planning algorithm that can generate similar trajectories.

Autonomous vessels operating in the city's waterway need to perform *socially compliant* navigation to safely interact among themselves and also with human-operated vessels. Unlike road vehicles whose operations are, in general, governed by well-structured and universal guidelines/regulations for road safety, the vessels are expected to follow region-specific regulations and norms that are hard to be formalized in unstructured space where lanes are not well defined. Using illustrative examples, Fig. 2 explains difficulties in formalizing norms for safe canal navigation.

To address this issue, in this article, we investigate the problem of designing a trajectory planning algorithm for finding vessel trajectories that resemble those of human-operated vessels and that allow autonomous vessels to perform socially compliant navigation. For this reason, we refer to the algorithm as *social trajectory planning* and to outcomes of it as *social trajectories*.

Manuscript received October 27, 2019; revised June 26, 2020; accepted September 4, 2020. Date of publication October 29, 2020; date of current version April 2, 2021. This work was supported in part by the Amsterdam Institute for Advanced Metropolitan Solutions in The Netherlands, in part by the Netherlands Organization for Scientific Research, Domain Applied Sciences, and in part by the National Science Foundation under Grant 1723943. This article was recommended for publication by Associate Editor F. Stulp and Editor A. Billard upon evaluation of the reviewers' comments. *(Corresponding author: Shinkyu Park.)*

Shinkyu Park and Daniela Rus are with the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: shinkyu7275@gmail.com; rus@csail.mit.edu).

Michal Cap and Javier Alonso-Mora are with the Department of Cognitive Robotics, Delft University of Technology, 2628 Delft, The Netherlands (e-mail: michal.cap@agents.fel.cvut.cz; j.alonsomora@tudelft.nl).

Carlo Ratti is with the Senseable City Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: ratti@mit.edu).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2020.3031250

Of relevance to our social trajectory planning problem are learning-from-demonstration approaches in which the main goal is for robots to learn socially compliant navigation behavior of human demonstrators (also called experts) [3]–[14]. Except a few [12]–[14], many of the works in the literature focus on learning local interaction rules from human demonstrations and use the learned rules to guarantee safety when robots are navigating in human-populated environments. However, as such approaches are designed for collision-free multirobot motion planning, which is more suitable for shorter term planning scenarios, the interaction rules are not rich enough to explain the emergence of long-term and global trajectory patterns (as illustrated in Fig. 2) and, hence, are not sufficient to model navigation behavior of human-operated vessels for larger scale trajectory planning.

Different from existing ones, our approach aims at synthesizing long-term and global trajectories. The key idea behind our approach is to design the trajectory planning algorithm based on an optimal control formulation in which the vessel's movements—speed and heading direction—that deviate from nominal movements of human-operated vessels are penalized. To implement the idea, we use a prerecorded vessel trajectory dataset to learn nominal movements of human-operated vessels and adopt a Kullback–Leibler (KL) control cost approach to impose such penalty. Interestingly, the analysis presented in Section V shows that our social trajectory planning can be viewed as a featureless equivalent of an inverse reinforcement learning (IRL) approach.

The main contributions of this article are summarized as follows.

1) We propose the trajectory planning algorithm, referred to as *social trajectory planning*. The algorithm leverages a KL control cost approach to define its cost function in which we derive the cost using a kernel density estimation (KDE) scheme and a dataset of prerecorded vessel trajectories.
2) We provide analytical results that support our claim that the social trajectory planning is capable of learning navigation behavior of human-operated vessels. In particular, by establishing its relation to maximum entropy IRL (MaxEntIRL) trajectory planning, we show that our approach is qualified as a learning-from-demonstration approach. On the other hand, we distinguish our approach from existing learning-from-demonstration approaches in that it does not require a set of features to define its formulation. Through experiments, we empirically verify that our approach outperforms MaxEntIRL trajectory planning, which is a feature-based IRL approach.
3) We demonstrate through data analysis that, in comparison with minimum travel-time (MinTravelTime) trajectory planning—a nonlearning-from-demonstration approach—and MaxEntIRL trajectory planning, the autonomous vessels adopting our social trajectory planning experience less head-on encounters[1] against both other autonomous vessels (also adopting the social trajectory planning) and human-operated vessels. This suggests that the social trajectory planning essentially reduces the amount of effort autonomous vessels should make to avoid collisions against other vessels in the canal. In addition, head-on encounters on social trajectories occur further away from the canal boundary, which suggests that in case the vessels undergo the encounters, they would have more room to avoid vessel-to-vessel collisions. From these analyses, we conclude that the social trajectory planning reduces the likelihood of vessel-to-vessel collisions stemming from head-on encounters.

[1]A precise definition of the head-on encounter and the motivation for adopting such encounter as a performance metric are provided in Section VI-A3.

The rest of this article is organized as follows. In Section II, we provide a comparative review of work related to our problem. In Section III, we formally introduce the social trajectory planning problem that aims at finding trajectories resembling those of human-operated vessels. The key idea behind our social trajectory planning approach is to adopt an optimal control formulation in which a KL control cost is used to penalize the deviation of the velocity of the autonomous vessel from those of human-operated vessels. In Section IV, we explain how to construct the KL control cost using a vessel trajectory dataset and to numerically compute a solution to the social trajectory planning problem. In Section V, we examine our trajectory planning approach from a perspective of the IRL. To analyze benefits of our approach, we perform extensive experiments and data analysis using a dataset recorded from the automatic identification system (AIS), which we report in Section VI. Finally, Section VII concludes this article.

II. Related Work

Trajectory planning for safe autonomous vehicle navigation has been an active research area whose topics range from classical formulations of cost-minimizing trajectory planning to recent developments on pedestrian-inspired and learning-based planning for navigation in human-populated environments. In this section, we review some of recent work in literature focusing on autonomous vehicle navigation that are relevant to our social trajectory planning problem. We refer the interested reader to a review article on autonomous vehicle planning for more in-depth discussions [15].

A fundamental aspect of safe robot navigation is to require robots to avoid collision against objects, such as other robots or pedestrians, while navigating toward their destinations. The work of van den Berg *et al.* [4] presents the idea of reciprocal collision avoidance for collision-free multirobot navigation. To allow robots to perform more human-like navigation and obstacle avoidance, Guzzi *et al.* [5] and Luo *et al.* [6] propose algorithms that incorporate observed pedestrian models. Simulation results of Guzzi *et al.* [5] demonstrate a performance improvement of the algorithm over the reciprocal collision avoidance. Experimental results of Luo *et al.* [6] validate the efficacy of the algorithm in robot navigation in pedestrian-populated environments.

The work of Knepper and Rus [16] proposes a collision avoidance method that, based on the idea of *equivalence relation*

on local paths [17], encodes the avoidance behavior inherent to pedestrians. In addition, using the formalism of topological braids, recent work [18] proposes a framework that is capable of generating *legible* collision-free paths for multirobot navigation.

Furthermore, Chen *et al.* [11] introduce a deep reinforcement learning framework to design a robot motion planning algorithm. The framework, which builds on simple yet explicit local interaction rules observed in pedestrian navigation, enables robots to execute socially aware collision avoidance in pedestrian-populated environments.

Different from the aforementioned references, the learning-based navigation approaches adopt the principle of *learning-from-demonstration*. These approaches aim at learning navigation behavior of experts from their demonstrations by designing data-driven frameworks for trajectory planning that allow robots to perform socially compliant navigation, e.g., avoiding other obstacles via a certain side of corridors while maintaining a safety distance to walls. Motivated by the so-called "freezing robot" problem, the work of Trautman *et al.* [19] develops an effective robot navigation framework that considers joint collision avoidance between robots and pedestrians, where a pedestrian movement dataset is used to train their Gaussian process model.

Other important developments in this research area build on the IRL approaches. Notably the work in [8]–[10] adopt different types of IRL approaches to enable robots to learn experts' local interaction rules from their demonstrations and use them for safe navigation in human-populated areas. Both simulation and experimental results reported in the work demonstrate that IRL approaches find trajectories resembling those of pedestrians and allowing robots to safely navigate in crowded environments.

More specific to the robot planning context related to our problem, applications of an IRL approach to route and trajectory planning are discussed in [12] and [13]. In particular, Ziebart *et al.* [12] investigate a route planning problem based on the MaxEntIRL approach to learn the route preference of cab drivers. The work demonstrated the potential of MaxEntIRL as a learning-based approach that allows robots to navigate like human drivers. The work of Ziebart *et al.* [13] describes an application of MaxEntIRL in modeling goal-directed trajectory planning of pedestrians and proposes a robot planning framework that, using the pedestrian trajectory planning model, enables robots to minimize the chances of hindering pedestrian movements while executing their assigned tasks.

The work of Wulfmeier *et al.* [14] proposes a deep neural network based MaxEntIRL approach for robot path planning. In particular, the authors investigate the problem of learning a cost function for path planning from experts' vehicle driving demonstrations and devise deep neural network architectures that are capable of reconstructing a cost function using LIDAR data from the vehicle and a driving demonstration dataset.

Our approach is relevant to the learning-from-demonstration approaches in that it aims at learning safe navigation of human-operated vessels using a prerecorded dataset. However, it is substantially different from these approaches in that, first, our framework finds global trajectories (rather than local interaction rules that are more suitable for generating trajectories over shorter time periods) for safe canal navigation, and second, the framework relates to the IRL-based trajectory planning approaches, but it does not require a well-defined set of features (either manually selected or learned from sensor data) to learn experts' navigation behavior.

We remark that, different from many of the references we reviewed here, collision avoidance is not the main focus of this article. However, our approach can be used to compute, for instance, a (socially compliant) reference trajectory for any local planning algorithm, which would refine the trajectory to ensure collision avoidance against other vessels.

TABLE I
LIST OF BASIC NOTATION

$\mathbb{X}$	the set of admissible vessel positions
$\mathbb{V}$	the set of admissible vessel velocities
x_t	the position of the autonomous vessel at time instant t
v_t	the velocity of the autonomous vessel at time instant t
z	a vector $\mathrm{z} \in \mathbb{X}$ indicates a location on the canal map
u	a vector $\mathrm{u} \in \mathbb{V}$ indicates a vessel velocity
$p(\mathrm{u} \mid \mathrm{z})$	the distribution of velocities of human-operated vessels at location z

III. METHOD FOR SOCIAL TRAJECTORY PLANNING

In this section, we formally describe the social trajectory planning approach discussed in this article. We begin by describing preliminaries needed to explain our approach. For convenience, in Table I, we summarize basic notation used throughout this article.

A. Preliminaries

1) Admissible Position and Velocity: Given a map of the canal, as depicted in Fig. 3, we denote by $\mathbb{X} \subset \mathbb{R}^2$ the set of admissible vessel positions representing the locations that are within the canal map and where the vessel can navigate. We impose the constraint that the vessel position $\mathrm{x} = (x, y)$ satisfies $\mathrm{x} \in \mathbb{X}$. We denote by $\mathbb{V} \subset \mathbb{R}^2$ the set of admissible velocity vectors representing the vessel velocities that are within the speed limit in the canal. We require the vessel velocity $\mathrm{v} = (\mathrm{v}_x, \mathrm{v}_y)$ to satisfy $\mathrm{v} \in \mathbb{V}$.

2) Control Policy and Trajectory: Given the sets $\mathbb{X}$ and $\mathbb{V}$ of admissible positions and velocity vectors, respectively, we call by a *control policy* a mapping $\mu : \mathbb{X} \to \mathbb{V}$ with which a vessel trajectory $(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})$ can be computed according to

$$\mathrm{x}_t = \mathrm{x}_0 + \sum_{\tau=0}^{t-1} \mu(\mathrm{x}_\tau),\ t \in \{0, \ldots, T_\mathrm{f}\} \tag{1}$$

where x_0 is the initial position of the vessel and we require that $(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})$ is contained in $\mathbb{X}$.

B. Social Trajectory Planning

Our social trajectory planning aims for the following goal. Given an origin $\mathrm{x}_\mathrm{o} \in \mathbb{X}$ and a destination $\mathrm{x}_\mathrm{d} \in \mathbb{X}$, find a control policy $\mu : \mathbb{X} \to \mathbb{V}$ that minimizes a cost $\mathcal{J}(\mu)$ for which its resulting trajectory $(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})$, determined by (1), satisfies the following two requirements.

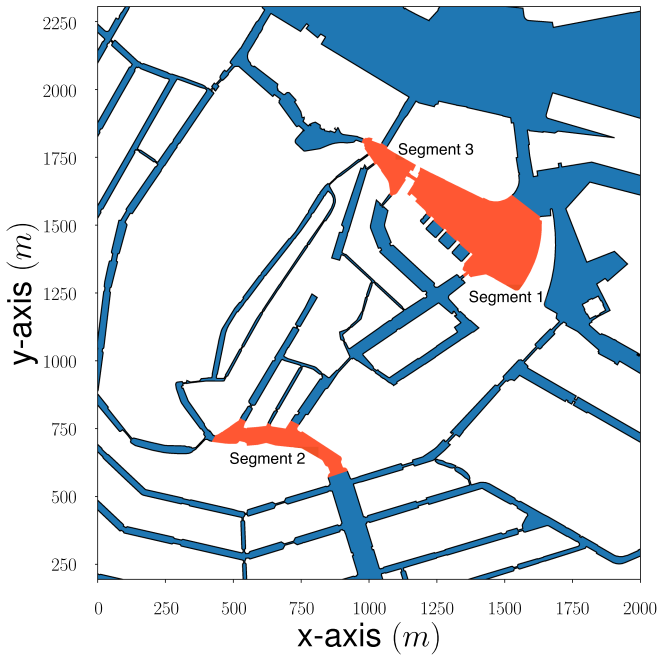


Fig. 3. Portion of Amsterdam canal map. The three orange regions on the map indicate the segments within which we perform the validation of the proposed approach using a vessel trajectory dataset collected from AIS (see Section VI).

1) *Global navigation*: It holds that $\mathrm{x}_0 = \mathrm{x}_\mathrm{o}$ and there is a finite terminal time index T_f such that $\mathrm{x}_{T_\mathrm{f}} = \mathrm{x}_\mathrm{d}$.
2) *Safety*: It holds that $\mathrm{x}_t \in \mathbb{X}$, $\mathrm{v}_t \in \mathbb{V}$ for all t in $\{0, \dots, T_\mathrm{f}\}$.

The cost $\mathcal{J}(\mu)$ should encapsulate how well the autonomous vessel performs socially compliant navigation under the control policy μ; hence, the key in the social trajectory planning is to design an appropriate cost function $\mathcal{J}$. In what follows, we explain the design of the function $\mathcal{J}$ using the KL control cost [20], which is used to impose restrictions on the movements of the vessel.

The KL control cost (2) defined below measures the deviation of the velocity v of the autonomous vessel from velocities of human-operated vessels at its position x on the canal map

$$\begin{aligned} &D_\mathrm{KL}\left(\delta\left(\mathrm{u}-\mathrm{v}\,|\,\mathrm{x}\right) \parallel p\left(\mathrm{u}\,|\,\mathrm{x}\right)\right) \\ &\quad := \int_\mathbb{V} \delta\left(\mathrm{u}-\mathrm{v}\,|\,\mathrm{x}\right) \ln \frac{\delta\left(\mathrm{u}-\mathrm{v}\,|\,\mathrm{x}\right)}{p\left(\mathrm{u}\,|\,\mathrm{x}\right)}\,\mathrm{du} \\ &\quad = -\ln p\left(\mathrm{v}\,|\,\mathrm{x}\right) + \int_\mathbb{V} \delta\left(\mathrm{u}-\mathrm{v}\,|\,\mathrm{x}\right) \ln \delta\left(\mathrm{u}-\mathrm{v}\,|\,\mathrm{x}\right)\,\mathrm{du} \end{aligned} \tag{2}$$

where $\delta(\mathrm{u}-\mathrm{v}\,|\,\mathrm{x})$ is the Dirac delta function (centered at v) that is regarded as the distribution of the autonomous vessel's velocity at its position x, and $p(\mathrm{u}\,|\,\mathrm{x})$ is the probability density function that describes the distribution of human-operated vessels' velocities at x.

Noting that the second term in (2) does not depend on v and is, in fact, a constant, the KL control cost is equivalent to $-\ln p(\mathrm{v}\,|\,\mathrm{x})$ up to the constant term. The key idea in the KL control cost approach is to construct an appropriate distribution $p(\mathrm{u}\,|\,\mathrm{z})$ at every admissible location z in $\mathbb{X}$ and impose restrictions on the velocity v of the autonomous vessel using it. In Section IV-A, based on a kernel density estimation (KDE) scheme, we describe how $p(\mathrm{u}\,|\,\mathrm{z})$ can be computed from a dataset consisting of trajectories of human-operated vessels.

Using the KL control cost (2), we define the cost $\mathcal{J}(\mu)$ by[2]

$$\mathcal{J}(\mu) := \underbrace{\lambda\, T_\mathrm{f}}_{\text{penalty due to travel time}} + \sum_{t=0}^{T_\mathrm{f}} \underbrace{-\ln p\left(\mathrm{v}_t\,|\,\mathrm{x}_t\right)}_{\text{KL control cost}}. \tag{3}$$

In light of (2), the second term, which we refer to as the KL control cost, quantifies the deviation of the velocity v_t of the autonomous vessel from velocities of human-operated vessels at its current position x_t and at time instant t. The cost (3) expresses the tradeoff between the total travel time and the KL control cost where the positive constant λ determines the importance of the travel-time penalty against the KL control cost. Once we find the control policy μ minimizing (3), using (1), we can generate a trajectory $(\mathrm{x}_0, \dots, \mathrm{x}_{T_\mathrm{f}})$ for which the requirements on the global navigation and safety are satisfied.

In the cost function (3), a higher value of λ assigns more emphasis on the total travel time. Hence, as the value of λ becomes larger, the resulting trajectory becomes closer to a minimum travel-time (MinTravelTime) trajectory. Vessel movements along such trajectory would frequently violate what are considered as *socially acceptable*. For instance, as we will show in Section VI, the vessels traveling along MinTravelTime trajectories experience more head-on encounters with one another than those traveling along social trajectories.

On the other hand, as the value of λ becomes smaller, the resulting trajectory has longer travel time but induces smaller KL control cost, suggesting that the autonomous vessel moves more like human-operated vessels.

IV. Control Policy Synthesis for Social Trajectory Planning

Two key technical aspects in the social trajectory planning are constructing an appropriate KL control cost and computing a control policy μ minimizing the cost function (3). For this purpose, we adopt the following methods.

1) Given a dataset of vessel trajectories collected from the AIS [21], we interpolate and estimate the pose and velocity of a vessel associated with each trajectory data.
2) KDE to construct the probability density function $p(\mathrm{u}\,|\,\mathrm{z})$ using the estimated trajectories. The probability density function will be used to construct the KL control cost as in (3).
3) Value iteration to implement dynamic programming and to numerically find an optimal solution to the social trajectory planning.

Fig. 4 illustrates an overview of our social trajectory planning approach.

[2]In addition to the KL control cost, the cost function (3) penalizes the travel time to ensure that resulting trajectories terminate at given destination points.

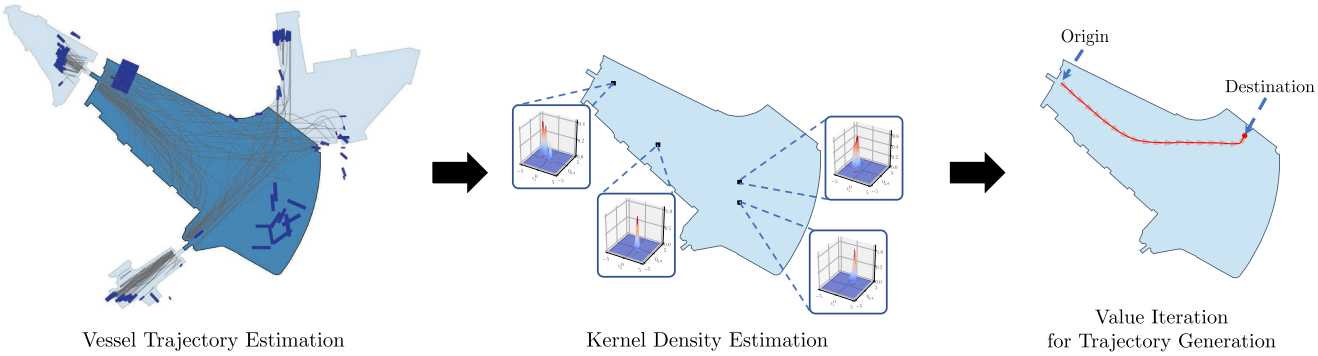


Fig. 4. Overview of the social trajectory planning: From an AIS dataset, we interpolate and estimate vessel trajectories; using KDE, from the estimated trajectories, we construct the KL control cost for (3); and by finding the optimal policy μ for (3) using the value iteration, we compute a social trajectory.

A. KDE Using AIS Dataset

We describe how to compute the probability density function $p(\mathrm{u}\,|\,\mathrm{z})$ from an AIS dataset using the KDE scheme [22]. The dataset consists of multiple sequences of time-stamped location data (GPS coordinates) where each sequence is associated with a vessel identified by a unique vessel ID. We denote the number of registered vessels and the location data (transformed in a Cartesian coordinate system), respectively, by M and $\{\mathrm{y}_{t_j}^{(i)}\}_{j=1}^{N_i}$, where N_i is the number of data points associated with vessel i and $\mathrm{y}_{t_j}^{(i)} \in \mathbb{R}^2$ represents the location data of the vessel at time instant t_j.

1) Trajectory Estimation: As a first step, given the data points $\{\mathrm{y}_{t_j}^{(i)}\}_{j=1}^{N_i}$, we interpolate the position $\mathrm{x}_t^{(i)}$ and estimate the linear velocity $\mathrm{v}_t^{(i)}$, the heading direction $\theta_t^{(i)}$, and the rotational velocity $\omega_t^{(i)}$, which will be used in the KDE step. Consider finding $\{(\mathrm{x}_t^{(i)}, \theta_t^{(i)}, \mathrm{v}_t^{(i)}, \omega_t^{(i)})\}_{t=t_1}^{t_{N_i}}$ that minimizes

$$\begin{aligned}
&\sum_{j=1}^{N_i} \underbrace{\left\|\mathrm{x}_{t_j}^{(i)} - \mathrm{y}_{t_j}^{(i)}\right\|^2}_{\text{interpolation distortion}} \\
&\quad + \sum_{\tau=t_1}^{t_{N_i}-1} \Bigg[\underbrace{\left\| Rot^T(\theta_{\tau+1}^{(i)})\,\mathrm{v}_{\tau+1}^{(i)} - Rot^T(\theta_\tau^{(i)})\,\mathrm{v}_\tau^{(i)} \right\|^2}_{\text{penalty on linear acceleration}} \\
&\quad + \underbrace{\left\|\omega_{\tau+1}^{(i)} - \omega_\tau^{(i)}\right\|^2}_{\text{penalty on angular acceleration}} + \underbrace{\left\| Rot_2^T(\theta_\tau^{(i)})\,\mathrm{v}_\tau^{(i)} \right\|^2}_{\text{penalty on lateral motion}} \Bigg] \\
&\text{subject to } \mathrm{x}_{t+1}^{(i)} = \mathrm{x}_t^{(i)} + \mathrm{v}_t^{(i)} \\
&\qquad\qquad \theta_{t+1}^{(i)} = \theta_t^{(i)} + \omega_t^{(i)}
\end{aligned} \tag{4}$$

where given the heading angle θ, the matrix $\mathrm{Rot}(\theta)$ represents the rotation matrix defined by

$$\mathrm{Rot}(\theta) = \begin{pmatrix} \mathrm{Rot}_1(\theta) & \mathrm{Rot}_2(\theta) \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}.$$

Note that $\mathrm{Rot}_2(\theta)$ used in (4) denotes the second column of $\mathrm{Rot}(\theta)$.

The second term in (4) is the regularization that penalizes the linear and angular accelerations and lateral motion of vessel i. The penalty on the acceleration is to enforce the estimated trajectory to be sufficiently smooth, and the penalty on the lateral motion is to take into account the fact that vessels in general cannot move laterally. The left plot in Fig. 4 depicts vessel trajectories estimated based on (4) using the AIS dataset in one canal segment. Each blue rectangle and gray curve, respectively, represent the initial pose and estimated trajectory of a vessel.

2) Kernel Density Estimation: After we interpolate vessel trajectories and estimate linear and angular velocities from the data, we estimate the probability density function $p(\mathrm{u}\,|\,\mathrm{z})$. We use the KDE scheme in which we adopt the Gaussian kernel to represent each vessel's velocity distribution.

We assume that the velocity of each vessel i at each time instant t is distributed according to a Gaussian distribution centered at the velocity data point $\mathrm{v}_t^{(i)}$. Based on the assumption, we adopt the Gaussian kernel $K_{\text{Gaussian}}(\mathrm{u} - \mathrm{v}_t^{(i)})$ defined as

$$\begin{aligned}
&K_{\text{Gaussian}}(\mathrm{u} - \mathrm{v}_t^{(i)}) \\
&\quad := \frac{1}{\mathcal{Z}_t^{(i)}} \exp\left[-\frac{1}{2\alpha} (\mathrm{u} - \mathrm{v}_t^{(i)})^T (\mathrm{u} - \mathrm{v}_t^{(i)}) \right]
\end{aligned} \tag{5}$$

where α is a parameter specifying the bandwidth of the Gaussian kernel, which we select to match the (estimated) variance ($= 0.2916\,\mathrm{m}^2/\mathrm{s}^2$) of $\mathrm{v}_t^{(i)}$, and $\mathcal{Z}_t^{(i)}$ is the normalizing factor given by

$$\mathcal{Z}_t^{(i)} = \int_{\mathbb{R}^2} \exp\left[-\frac{1}{2\alpha} (\mathrm{u} - \mathrm{v}_t^{(i)})^T (\mathrm{u} - \mathrm{v}_t^{(i)}) \right] \mathrm{du}.$$

On the other hand, we approximate the shape of each vessel i with a rectangle rotated by the vessel's heading angle $\theta_t^{(i)}$ and represent the points on the canal map occupied by the vessel at each time instant t using the following indicator function. For each location z in $\mathbb{X}$

$$I_t^{(i)}(\mathrm{z}) := \begin{cases} 1 & \text{if } \mathrm{Rot}^T(\theta_t^{(i)})(\mathrm{z} - \mathrm{x}_t^{(i)}) \in \mathbb{B}_{w,l} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where

$$\begin{aligned}
\mathbb{B}_{w,l} = \big\{ (\mathrm{z}_1, \mathrm{z}_2) \in \mathbb{R}^2 \,\big|\, &-l/2 \le \mathrm{z}_1 \le l/2, \\
&-w/2 \le \mathrm{z}_2 \le w/2 \big\}
\end{aligned}$$

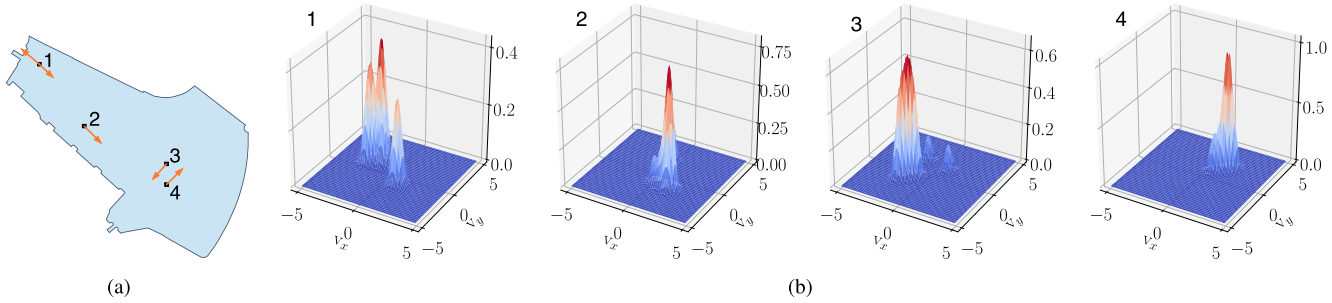


Fig. 5. Illustration of the probability density function $p(\mathrm{u}\,|\,\mathrm{z})$ at four different locations. The graphs in (b) depict $p(\mathrm{u}\,|\,\mathrm{z})$ evaluated at the four different locations indicated by the black dots in (a). In (a), the arrows represent the vessel velocities corresponding to the modes of the probability density functions, depicted in (b).

is the set of points constituting a two-dimensional rectangle with width w and length l for which we select w and l to match them with the width and length of each vessel i, respectively.

Combining (5) and (6), we define $K^{(i)}$, which represents the velocity distribution of vessel i at location z, as follows:

$$K^{(i)}(\mathrm{u}\,|\,\mathrm{z}) = \begin{cases} \dfrac{\sum_{\tau=t_1}^{t_{N_i}} K_{\text{Gaussian}}\left(\mathrm{u}-\mathrm{v}_\tau^{(i)}\right)\cdot I_\tau^{(i)}(\mathrm{z})}{\sum_{\tau=t_1}^{t_{N_i}} I_\tau^{(i)}(\mathrm{z})} & \text{if } \sum_{\tau=t_1}^{t_{N_i}} I_\tau^{(i)}(\mathrm{z}) > 0 \\ 0 & \text{otherwise} \end{cases}.$$

Assuming that the prior distribution on the vessel velocities is uniform, based on the KDE scheme, we obtain the estimate of the probability density function $p(\mathrm{u}\,|\,\mathrm{z})$ by combining the uniform density function K_{Uniform} defined over $\mathbb{V}$ and the velocity distribution $K^{(i)}$ of every vessel as follows:

$$p(\mathrm{u}\,|\,\mathrm{z}) = \frac{1}{\mathcal{Z}(\mathrm{z})}\left[\sum_{i=1}^{M} K^{(i)}(\mathrm{u}\,|\,\mathrm{z}) + K_{\text{Uniform}}(\mathrm{u})\right] \tag{7}$$

where the normalizing factor $\mathcal{Z}$ is defined as

$$\mathcal{Z}(\mathrm{z}) = \sum_{i=1}^{M}\int_{\mathbb{R}^2} K^{(i)}(\mathrm{u}\,|\,\mathrm{z})\,\mathrm{du} + 1$$

which is essentially the number of vessels visiting the location z plus 1. Indeed, when M is substantially large, the effect of the prior $K_{\text{Uniform}}(\mathrm{u})$ on computing $p(\mathrm{u}\,|\,\mathrm{z})$ becomes negligible.

Fig. 5 illustrates the estimated probability density function $p(\mathrm{u}\,|\,\mathrm{z})$ at four different locations on a canal segment. At location 1 where we observe vessels traveling in two opposite directions to enter from and exit to a neighboring canal segment, the probability density function $p(\mathrm{u}\,|\,\mathrm{z})$ is multimodal. At other locations, on the other hand, the probability density functions tend to be unimodal with their modes appearing at different points in $\mathbb{V}$. In particular, as we can observe from locations 3 and 4, the majority of the vessels passing location 3 travel South West and those passing location 4 travel North East.

B. Value Iteration

One technical challenge in solving the social trajectory planning problem is in finding its optimal solution. Since the cost function (3) is not guaranteed to be convex, especially because the KL control cost is constructed using a real-world dataset, many of convex optimization tools are not directly applicable to find such solution. For this reason, we use the value iteration method [23] to find the optimal solution, which we briefly describe below.

The key step in the value iteration method is to find a sequence $\{\mathcal{V}_n\}_{n=1}^{\infty}$ of value functions for which each value function $\mathcal{V}_n : \mathbb{X} \to \mathbb{R}$ is iteratively updated according to

$$\mathcal{V}_{n+1}(\mathrm{x}) = \min_{\mathrm{v}\in\mathbb{V}}\left[\lambda - \ln p(\mathrm{v}\,|\,\mathrm{x}) + \mathcal{V}_n(\mathrm{x}+\mathrm{v})\right]. \tag{8}$$

When the sequence of value functions converges, say to $\mathcal{V}^*$, (8) yields

$$\mathcal{V}^*(\mathrm{x}) = \min_{\mathrm{v}\in\mathbb{V}}\left[\lambda - \ln p(\mathrm{v}\,|\,\mathrm{x}) + \mathcal{V}^*(\mathrm{x}+\mathrm{v})\right]. \tag{9}$$

According to Bellman's principle of optimality, we can derive the optimal control policy μ as

$$\mu(\mathrm{x}) = \arg\min_{\mathrm{v}\in\mathbb{V}}\left[\lambda - \ln p(\mathrm{v}\,|\,\mathrm{x}) + \mathcal{V}^*(\mathrm{x}+\mathrm{v})\right].$$

Computational methods to implement the value iteration (8) are described in machine learning literature. We adopt a simplified version of the method described in [24] for our implementation.

V. Analysis on Social Trajectory Planning

In this section, we explain key aspects of the formulation used to define our social trajectory planning in connection with IRL, one of learning-from-demonstration approaches. A key idea of IRL approaches on trajectory planning is to derive a cost function using demonstrated trajectories by experts and use it to compute optimal trajectories that presumably resemble important features of the experts' demonstrations.

In the social trajectory planning, considering that human-operated vessels are the experts and AIS trajectory data are their demonstrations, our approach is closely related to IRL approaches. In what follows, we analytically show that our formulation on the social trajectory planning, as described in Section III-B, can be cast into the form used in MaxEntIRL [25]. The analysis suggests that our social trajectory planning is related to IRL approaches, but uses different bases to construct the cost function in its optimal control formulation (3).

1) Trajectory Planning With MaxEntIRL: For our discussion, we adopt the following formulation for trajectory planning

based on MaxEntIRL: Consider the reward function $\mathcal{R}$ and constraints given by

$$\begin{aligned} &\mathcal{R}\left(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}}\right) \\ &\text{subject to } \mathrm{x}_{t+1} = \mathrm{x}_t + \mathrm{v}_t \\ &\qquad \mathrm{v}_t = \mu\left(\mathrm{x}_t\right) \\ &\qquad T_\mathrm{f} \leq T_{\max} \\ &\qquad \mathrm{x}_0 = \mathrm{x}_\mathrm{o},\ \mathrm{x}_{T_\mathrm{f}} = \mathrm{x}_\mathrm{d} \\ &\qquad \mathrm{x}_t \in \mathbb{X},\ \mathrm{v}_t \in \mathbb{V} \quad \forall t \in \{0, \ldots, T_\mathrm{f}\}. \end{aligned} \tag{10}$$

We note that the MaxEntIRL formulation seeks the policy μ that derives a trajectory $(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})$ from the origin x_o to the destination x_d based on the reward $\mathcal{R}$ and subject to the travel-time constraint $T_\mathrm{f} \leq T_{\max}$.

In the MaxEntIRL formulation, provided that a so-called *feature vector* $f : \mathcal{X} \to \mathbb{R}^k$ is given, the reward $\mathcal{R}$ is defined as $\mathcal{R}(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}}) = \theta^T f(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})$, where $\mathcal{X}$ is the set of trajectories $(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})$ between x_o and x_d and the parameter $\theta \in \mathbb{R}^k$ is referred to as the *feature weight*. According to the MaxEntIRL principle, the reward $\mathcal{R}$ defines the probability density function $p : \mathcal{X} \to \mathbb{R}_+$ as

$$\begin{aligned} &p\left(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}}\right) \\ &\quad = \frac{\exp\left(\mathcal{R}(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})\right)}{\int_{\mathcal{X}} \exp\left(\mathcal{R}(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})\right)\, \mathrm{dx}} \\ &\quad = \frac{\exp\left(\theta^T f(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})\right)}{\int_{\mathcal{X}} \exp\left(\theta^T f(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})\right)\, \mathrm{dx}}. \end{aligned} \tag{11}$$

One of key aspects in MaxEntIRL is to select the feature weight θ using a set of demonstrated trajectories, which we denote by $\mathcal{D} \subset \mathcal{X}$, based on the maximum likelihood estimation: Given the set $\mathcal{D}$, using (11), we select θ as

$$\theta = \arg\max_{\theta \in \mathbb{R}^k} \prod_{(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}}) \in \mathcal{D}} p\left(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}}\right). \tag{12}$$

2) Social Trajectory Planning as IRL: Now, we explain how our social trajectory planning formulation is related to (10). Let us begin by recalling the social trajectory planning problem. Find a control policy $\mu : \mathbb{X} \to \mathbb{V}$ that minimizes

$$\begin{aligned} &\mathcal{J}\left(\mu\right) = \lambda\, T_\mathrm{f} - \sum_{t=0}^{T_\mathrm{f}-1} \ln p\left(\mathrm{v}_t \,|\, \mathrm{x}_t\right) \\ &\text{subject to } \mathrm{x}_{t+1} = \mathrm{x}_t + \mathrm{v}_t \\ &\qquad \mathrm{v}_t = \mu\left(\mathrm{x}_t\right) \\ &\qquad \mathrm{x}_0 = \mathrm{x}_\mathrm{o},\ \mathrm{x}_{T_\mathrm{f}} = \mathrm{x}_\mathrm{d} \\ &\qquad \mathrm{x}_t \in \mathbb{X},\ \mathrm{v}_t \in \mathbb{V} \quad \forall t \in \{0, \ldots, T_\mathrm{f}\} \end{aligned} \tag{13}$$

where T_f is the total travel time.

Recall that we construct the probability density function $p(\mathrm{v}_t \,|\, \mathrm{x}_t)$ used in (13) from a dataset on human-operated vessel trajectories, as described in Section IV-A. Assume that the (human-operated) vessels adopt state-dependent control policies of the form $\mu : \mathbb{X} \to \mathbb{V}$ in which case the probability density function $p(\mathrm{v}_t \,|\, \mathrm{x}_t)$ satisfies

$$p\left(\mathrm{v}_t \,|\, \mathrm{x}_t\right) = p\left(\mathrm{v}_t \,|\, \mathrm{x}_0, \ \ldots, \ \mathrm{x}_t\right). \tag{14}$$

Using (14), we can express the second term in (13) as follows:

$$\begin{aligned} \sum_{t=0}^{T_\mathrm{f}-1} \ln p\left(\mathrm{v}_t \,|\, \mathrm{x}_t\right) &= \ln \prod_{t=0}^{T_\mathrm{f}-1} p\left(\mathrm{v}_t \,|\, \mathrm{x}_t\right) \\ &= \ln \prod_{t=0}^{T_\mathrm{f}-1} p\left(\mathrm{v}_t \,|\, \mathrm{x}_0, \ldots, \mathrm{x}_t\right) \\ &= \ln \prod_{t=0}^{T_\mathrm{f}-1} p\left(\mathrm{x}_{t+1} \,|\, \mathrm{x}_0, \ldots, \mathrm{x}_t\right) \\ &= \ln p\left(\mathrm{x}_1, \ldots, \mathrm{x}_{T_\mathrm{f}} \,|\, \mathrm{x}_0\right). \end{aligned} \tag{15}$$

Note that, as x_0 is fixed to x_o, the last equation in (15) is equivalent to $\ln p(\mathrm{x}_0, \ldots, \mathrm{x}_{T_\mathrm{f}})$ up to the constant term $\ln p(\mathrm{x}_0)$.

By noting that the denominator in (11) is constant, comparing (11) and (15), we can rewrite the cost function (13) as follows:

$$\mathcal{J}(\mu) = \lambda\, T_\mathrm{f} - \mathcal{R}\left(\mathrm{x}_0, \ \ldots, \ \mathrm{x}_{T_\mathrm{f}}\right) + \alpha \tag{16}$$

where α is a constant that does not depend on the choice of the policy μ.

In light of (16), the optimization (13) is identical to (10), except we represent the travel time constraint in (10) as a penalty term in the cost function of (13). Therefore, our social trajectory planning finds the optimal control policy μ that maximizes a reward $\mathcal{R}$ satisfying (11) subject to the total travel time constraint $T_\mathrm{f} \leq T_{\max}$, where the value of $T_{\max}$ depends on the selection of λ in (13).

On the other hand, our social trajectory approach is different from the MaxEntIRL approach in that it does not require any predefined feature vector f as in (11), but it rather estimates the movement model $p(\mathrm{u} \,|\, \mathrm{z})$ of human-operated vessels and uses the model to build the cost function. Indeed, since we adopt the KDE method, which is a nonparametric estimation scheme, our approach would, in general, require a larger dataset than feature-based IRL approaches, such as MaxEntIRL. However, because our method accepts all vessel trajectory data regardless of their origins and destinations, we can collect vast vessel trajectory datasets through AIS.

VI. Experimental Results

We perform experiments using the AIS trajectory dataset to evaluate the benefits of the social trajectory planning in the following two aspects: First, improving the predictability of the autonomous vessel's navigation behavior and second, reducing head-on encounters against other autonomous and human-operated vessels. For the purpose of the evaluation, we compare the performance of the social trajectory planning, where we select $\lambda = 1$ for its cost function (3), with those of the MinTravelTime—a nonlearning-based approach—and MaxEntIRL—an existing learning-from-demonstration approach, which we discussed in Section V—trajectory planning. The implementation details of both approaches are explained in Section VI-A2.

1) *Predictability of autonomous vessel's navigation behavior*: In [26], the predictability of robot motion is defined as how well an (human) observer can predict the robot's trajectory given that the observer knows the destination of the robot. Adapting this concept in our context, we evaluate whether the social trajectory planning enables autonomous vessels to adopt trajectories that are more likely to be chosen by human-operated vessels and, hence, those trajectories would be more "predictable" by human operators.
   To this end, in the first experiment, we begin by evaluating how much social trajectories resemble those of human-operated vessels. Using all three trajectory planning approaches, we compute trajectories between the origin and destination of each trajectory data from the AIS dataset. We then assess the Euclidean distances between computed trajectories and associated trajectory data.[3] Outcomes of this experiment, as presented in Section VI-B, suggest that our social trajectory planning attains the smallest mean value of the Euclidean distances, and this confirms that our social trajectory planning is capable of generating trajectories that resemble those of human-operated vessels better than the two other approaches.
2) *Frequency and location of head-on encounters*: To assess the benefits of adopting the social trajectory planning, we perform extensive experiments—total 9×10^6 different cases—to evaluate the frequency and location of head-on encounters among autonomous vessels and between autonomous and human-operated vessels in which the autonomous vessels are navigating along one of MinTravelTime, MaxEntIRL, and social trajectories. Using outcomes of the experiment, we carry out statistical analysis and draw the conclusion that the vessels navigating along social trajectories experience less head-on encounters and the encounters occur further away from the canal boundary than those navigating along MinTravelTime and MaxEntIRL trajectories. These results support our claim that, compared to the two other approaches, our social trajectory planning would reduce the chances of vessel-to-vessel collisions, especially under low visibility conditions in which head-on collisions are empirically dominant [27].

A. Experiment Settings

Below, we explain the dataset we used to conduct the experiments on three canal segments, as illustrated in Fig. 3. We also provide the descriptions on the implementation of the MinTravelTime and MaxEntIRL trajectory planning—two planning approaches used as baselines—and the definitions of the head-on encounter and vessel dimension.

1) Dataset and Selection of Canal Segments for Evaluation: The AIS dataset we used for the experiments was collected during $2017/8/12\ 13:00$–$2017/8/12\ 17:00$ (CEST) and contains 182, 312, and 173 trajectories for Segments 1, 2, and 3, respectively.[4] We select $4/5$ of the data uniformly randomly to compute the probability density function $p(\mathrm{u} \mid \mathrm{z})$ using the method described in Section IV-A. We use the rest as the validation dataset to assess the performance of our method in the experiments.

TABLE II
DIMENSION OF THE CANAL SEGMENTS USED IN THE EXPERIMENTS (SEE FIG. 3 FOR THEIR LOCATIONS ON THE CANAL MAP)

segment	length	width	area
1	$526\ m$	$335\ m$	$108,198\ m^2$
2	$505\ m$	$156\ m$	$29,407\ m^2$
3	$234\ m$	$133\ m$	$16,073\ m^2$

The admissible velocity set $\mathbb{V}$ contains all the velocity vectors that are below the maximum vessel speed observed in the dataset, which is approximately 3 m/s. We select three canal segments, as indicated in Fig. 3, that are of different size and each has multiple regions where vessels enter the segment and exit to neighboring segments. The dimension of the three segments is summarized in Table II.

2) MinTravelTime and MaxEntIRL Trajectory Planning: We compare the performance of our social trajectory planning with those of the MinTravelTime and MaxEntIRL trajectory planning. We provide the descriptions of both of the approaches in the following.

1) *MinTravelTime*: Consider the cost function $\mathcal{J}_{\text{MinTravelTime}}$ and constraints given by

$$\begin{aligned} &\mathcal{J}_{\text{MinTravelTime}}(\mu) = T_{\mathrm{f}} \\ &\text{subject to } \mathrm{x}_{t+1} = \mathrm{x}_t + \mathrm{v}_t \\ &\qquad \mathrm{v}_t = \mu(\mathrm{x}_t) \\ &\qquad \mathrm{x}_0 = \mathrm{x}_{\mathrm{o}},\ \mathrm{x}_{T_{\mathrm{f}}} = \mathrm{x}_{\mathrm{d}} \\ &\qquad \mathrm{x}_t \in \mathbb{X},\ \mathrm{v}_t \in \mathbb{V} \quad \forall t \in \{0, \ldots, T_{\mathrm{f}}\} \end{aligned} \tag{17}$$

where x_{o}, x_{d} is a given pair of origin and destination points, and T_{f} denotes the total travel time. The MinTravelTime trajectory planning aims at finding a control policy $\mu : \mathbb{X} \to \mathbb{V}$ that minimizes (17). We adopt a similar standard value iteration method, as described in Section IV-B, to find the optimal policy μ.

2) *MaxEntIRL*: Consider the cost function $\mathcal{J}_{\text{MaxEntIRL}}$ and constraints given by

$$\begin{aligned} &\mathcal{J}_{\text{MaxEntIRL}}(\mu) = -\sum_{t=0}^{T_{\mathrm{f}}-1} \theta^T f(\mathrm{x}_t, \mathrm{v}_t) \\ &\text{subject to } \mathrm{x}_{t+1} = \mathrm{x}_t + \mathrm{v}_t \\ &\qquad \mathrm{v}_t = \mu(\mathrm{x}_t) \\ &\qquad \mathrm{x}_0 = \mathrm{x}_{\mathrm{o}},\ \mathrm{x}_{T_{\mathrm{f}}} = \mathrm{x}_{\mathrm{d}} \\ &\qquad \mathrm{x}_t \in \mathbb{X},\ \mathrm{v}_t \in \mathbb{V} \quad \forall t \in \{0, \ldots, T_{\mathrm{f}}\}. \end{aligned} \tag{18}$$

By making a similar selection of the feature vector f as in [13], we adopt five features $f = (f_1, \ldots, f_5)$: the

[3]The Euclidean distance between two trajectories is defined as the average distance between every point on one trajectory and its closest point on the other trajectory.

[4]We demonstrate that a single-day dataset is enough to observe the benefits of the social trajectory planning in improving canal navigation safety.

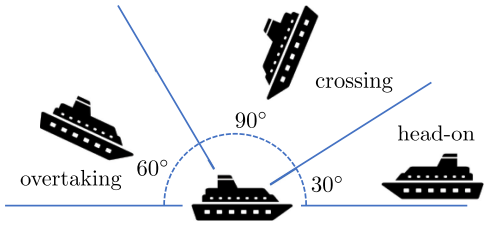


Fig. 6. Three different types of vessel-to-vessel encounters.

first feature f_1 takes a (negative) constant value across the canal map and the second feature f_2 is defined by (the negative of) the squared Euclidean norm $\|\mathrm{v}_t\|^2$ of the vessel velocity v_t. These two features represent the rewards associated with the travel time and velocity of the autonomous vessel. The rest of the features f_3, f_4, f_5 are defined using blurred images of the canal map, obtained by applying Gaussian filters with all different standard deviations. These features are used to quantify the position of the vessel with respect to the boundary of (or other obstacles on) the canal map. Applying MaxEntIRL as in [13], we compute the feature weight $\theta = (\theta_1, \ldots, \theta_5)$ using trajectory data from the AIS dataset.[5] We adopt the "softened" value iteration, as in [13], to compute a value function for (18) and using it, we find a (deterministic) policy μ that derives vessel trajectories that most likely occur under the MaxEntIRL approach.

3) Head-On Encounter and Vessel Dimension: We define the head-on encounter as an event in which two vessels contact each other at the relative degree (heading angle) between 0° and 30° (see Fig. 6 for an illustration of the head-on encounter). Based on the statistics reported in [27], this is the most frequently occurring vessel-to-vessel encounter in waterways that would potentially lead to a collision under restricted visibility conditions. Also, our analysis on the frequency of all types of encounters—overtaking, crossing, and head-on counters (see Fig. 6 for an illustration)—suggested that the head-on encounter is the most frequently occurred one.

In order to assess vessel-to-vessel encounters in the experiments, we need to define the dimension of vessels. We approximate the shape of every vessel as a rectangle and determine its width and length as follows. For the human-operated vessels recorded in the AIS dataset, their width and length are given in the dataset. For autonomous vessels, we assume that their dimension is equal to the average dimension of the human-operated vessels, which is 5 m wide and 20 m long.

B. Predictability of Autonomous Vessel's Navigation Behavior

We assess the predictability of the autonomous vessel's trajectory under the MinTravelTime, MaxEntIRL, and social trajectory planning. For this purpose, we compute vessel trajectories using all three planning approaches between the origin and destination of each trajectory data from the validation dataset, and we evaluate the Euclidean distances between the computed trajectories and associated trajectory data. Adapting the definition of the predictability of robot motion from Dragan *et al.* [26], the assessment will be used to draw the conclusion that the trajectories computed by the social trajectory planning are closest to those adopted by the human-operated vessels and, hence, would be more predictable by human operators than the other planning approaches.

Fig. 7 depicts the comparison between the trajectories computed by the three planning approaches and their associated trajectory data from the validation dataset, and Fig. 8 summarizes the statistics on their Euclidean distances. We observe that the social trajectories are closer to the trajectory data than the MinTravelTime trajectories ($p = 0.0176$, one-sided t-test) and MaxEntIRL trajectories ($p = 0.0022$, one-sided t-test). Surprisingly, the MaxEntIRL trajectories recorded the largest Euclidean distance among the three approaches.

This outcome suggests that our social trajectory approach improves the predictability of the autonomous vessel's navigation behavior compared to the MinTravelTime and MaxEntIRL trajectory planning. On the other hand, despite the analogy between our social trajectory planning and the MaxEntIRL trajectory planning, as established in Section V, the latter approach does not perform similarly to the social trajectory planning. In fact, the MinTravelTime approach outperforms the MaxEntIRL approach. This observation emphasizes the importance of the feature selection in the MaxEntIRL approach.

Moreover, we notice that the social trajectories attain the mean Euclidean distance of 7.4 m from their associated trajectory data. It appears that this is because human-operated vessels, traveling between same origin and destination points, would select different trajectories based on their vessel type, dimension, traffic volume, etc., whereas our social trajectory planning assumes that the origin and destination are the only factors that affect the trajectory selection. Also, unlike the control policy μ adopted in our approach that depends only on the current vessel position, a human operator would steer his/her vessel depending on the past trajectory it traveled. Hence, the resulting human-operated vessel's trajectory would differ from one determined by the social trajectory planning.

C. Frequency and Location of Head-On Encounters

We assess the benefits of the social trajectory planning in improving canal navigation safety. For this purpose, using the three trajectory planning approaches, we compute trajectories between preselected origin and destination points in the three canal segments, as depicted in Fig. 9. In each segment, origin and destination points are randomly selected from the regions where vessels can enter the segment and exit to neighboring segments. Then, we count and record the frequency and location of head-on encounters between every pair of trajectories.

Our analysis involves the following two different scenarios: head-on encounters among autonomous vessels and head-on encounters between autonomous and human-operated vessels, where the autonomous vessels are assigned with the computed trajectories of the same type. Outcomes of both the scenarios are

[5]To allow the feature weight θ to vary depending on the destination point x_d, we categorize trajectory data based on their destinations and compute the feature weight using the data in each category. This enables the MaxEntIRL trajectory planning to use different cost functions depending on the destinations of the vessels.

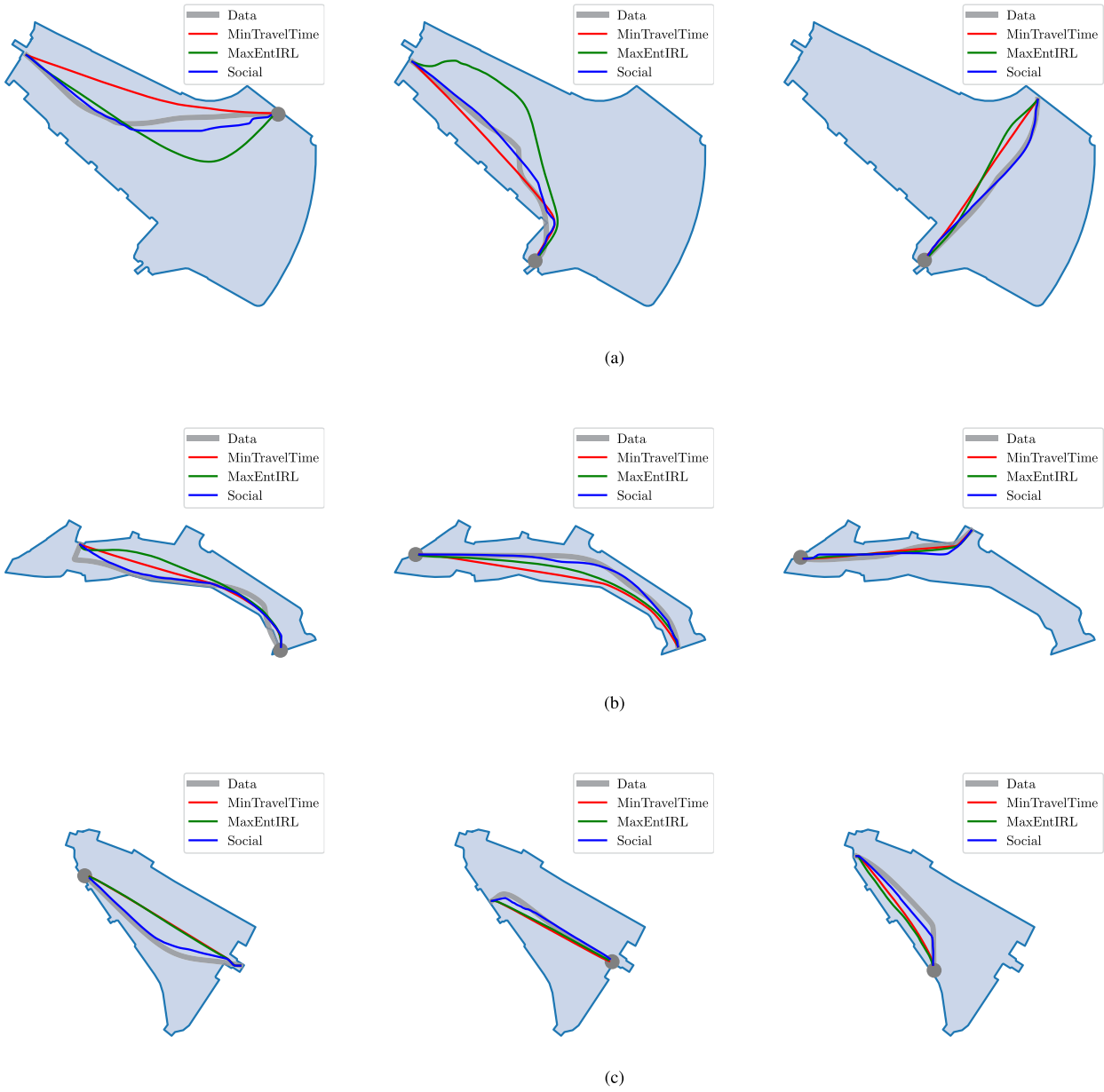


Fig. 7. Select figures depicting the comparison of MinTravelTime (red), MaxEntIRL (green), and social (blue) trajectories with respect to associated trajectory data (gray). In each figure, the four trajectories span between a same pair of origin and destination points where the destination is marked with a gray circle. (a) Segment 1. (b) Segment 2. (c) Segment 3.

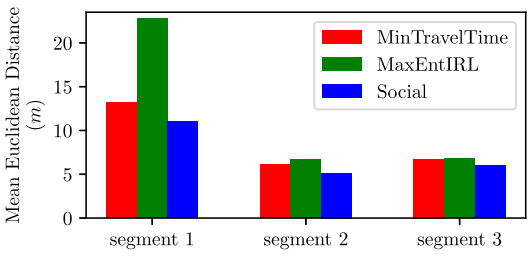


Fig. 8. Graph depicting the mean value of the Euclidean distances between the trajectories computed by the MinTravelTime (red), MaxEntIRL (green), and social (blue) trajectory planning and associated trajectory data.

used to conclude that, in comparison with the MinTravelTime and MaxEntIRL trajectory planning, the autonomous vessels navigating along social trajectories experience substantially less frequent head-on encounters with other autonomous and human-operated vessels. Furthermore, our analysis suggests that the encounters take place further away from the canal boundary under the social trajectory planning.

1) Analysis on Head-On Encounters Among Autonomous Vessels: To assess head-on encounters among autonomous vessels, we consider that two vessels are navigating along the computed trajectories of the same type where the vessels are

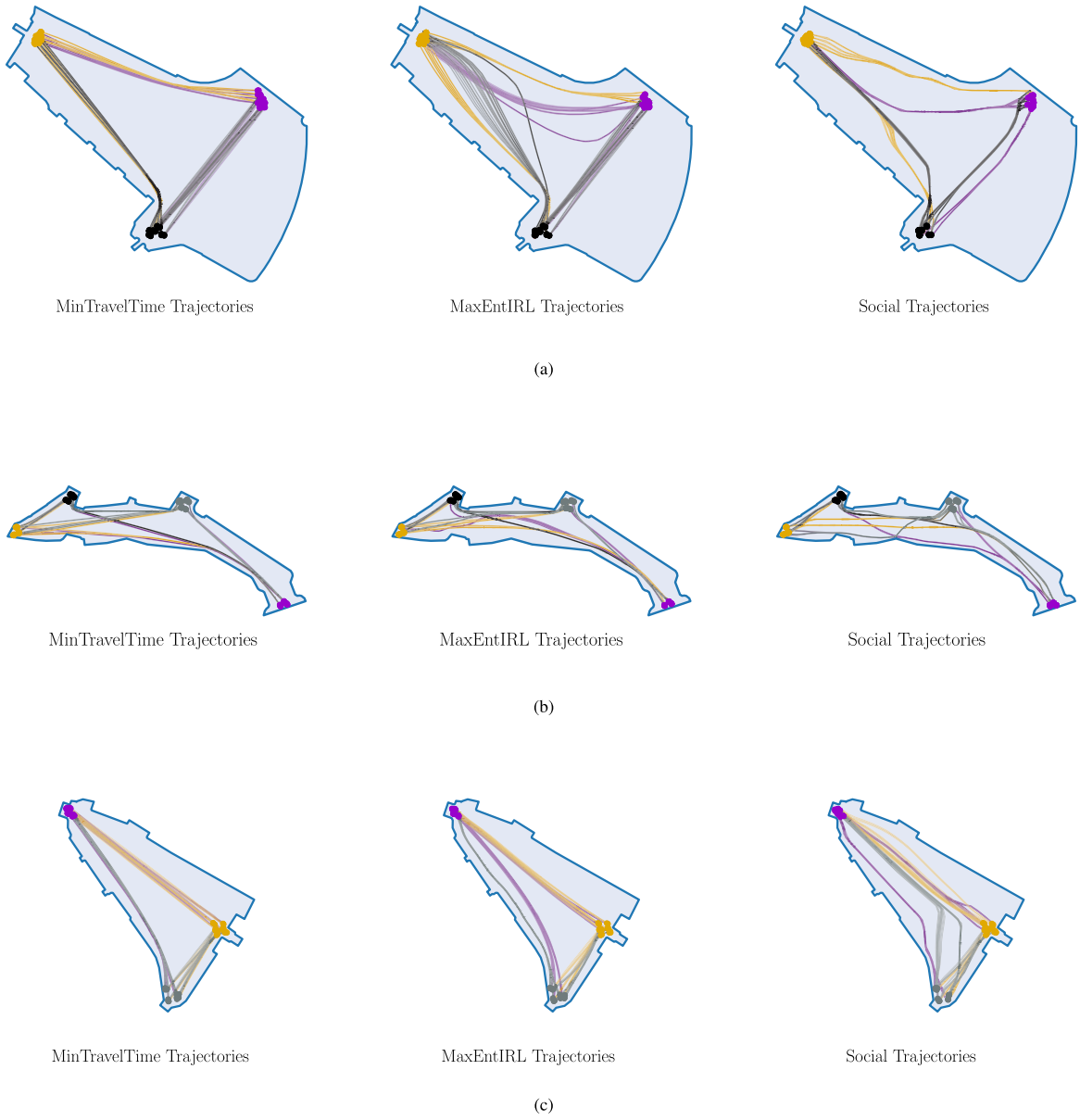


Fig. 9. Trajectories between predefined pairs of origin and destination points computed using the MinTravelTime, MaxEntIRL, and social trajectory planning in the three canal segments. We depict the destination points as colored circles and computed trajectories leading to each destination point as curves of the same color as their destination points. (a) Segment 1. (b) Segment 2. (c) Segment 3.

allowed to depart from their respective origins at different time instants, and we examine whether the vessels encounter each other at the relative orientation between 0° and 30°.

We examine approximately 6×10^6 different cases in all three segments using the MinTravelTime, MaxEntIRL, and social trajectory planning. Table III specifies the reduction (in percentage) in the head-on encounter frequency for the MaxEntIRL and social trajectory planning with respect to the MinTravelTime trajectory planning. As can be observed in Table III, the social trajectory planning results in significantly less head-on encounters than the MinTravelTime and MaxEntIRL trajectory planning in all three segments ($p < 0.001$, chi-square test). Moreover, in Segment 2, we have observed that the MaxEntIRL trajectory planning attains higher head-on encounter frequency than the MinTravelTime trajectory planning.

Fig. 10(a) depicts the frequency of the head-on encounters with respect to their distances from the canal boundary. From the histograms in Fig. 10(a), we can see that the encounters on MinTravelTime and MaxEntIRL trajectories take place more frequently near the canal boundary. On the other hand, the

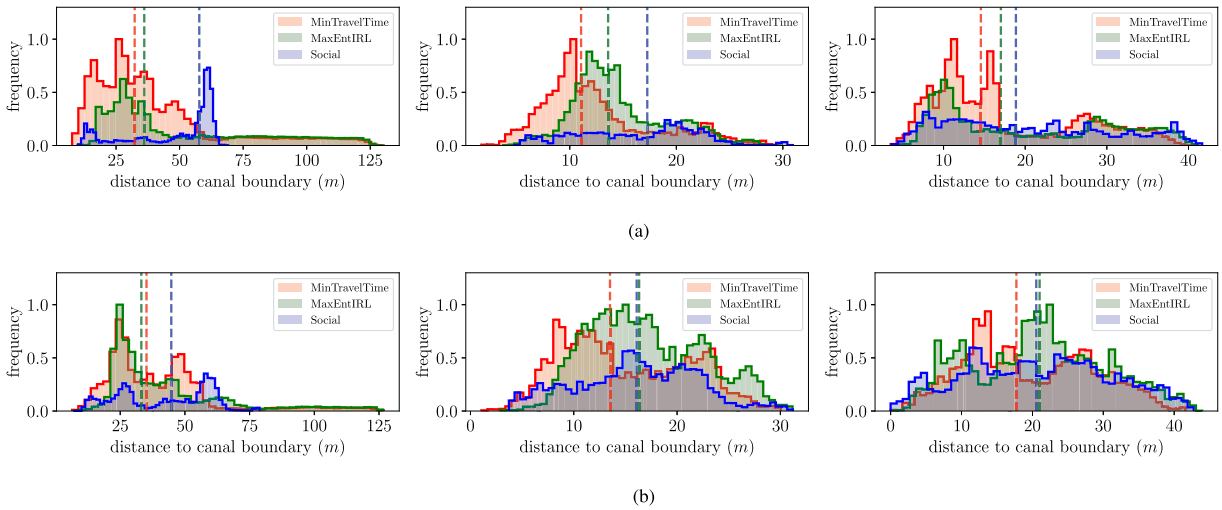


Fig. 10. Histograms depicting the head-on encounter frequency (a) among autonomous vessels and (b) between autonomous and human-operated vessels with respect to the distance to the canal boundary for the MinTravelTime (red), MaxEntIRL (green), and social (blue) trajectory planning. The dotted vertical lines represent the medians of the frequency data and the histograms are rescaled such a way that the largest value of the frequency is equal to unity. (a) Head-on encounter frequency among autonomous vessels in Segments 1–3. (b) Head-on encounter frequency between autonomous and human-operated vessels in Segments 1–3.

TABLE III
COMPARISONS ON THE HEAD-ON ENCOUNTER FREQUENCY: EACH PERCENTAGE SPECIFIES THE REDUCTION IN THE HEAD-ON ENCOUNTER FREQUENCY ATTAINED BY THE MAXENTIRL AND SOCIAL TRAJECTORY PLANNING WITH RESPECT TO THE MINTRAVELTIME TRAJECTORY PLANNING

segment	among autonomous vessels		between autonomous and human-operated vessels	
	MaxEntIRL	Social	MaxEntIRL	Social
1	46.9 %	53.1 %	13.0 %	36.1 %
2	-4.5 %	64.0 %	-31.9 %	31.1 %
3	34.5 %	41.0 %	-19.5 %	4.1 %

TABLE IV
COMPARISONS ON THE MEDIAN DISTANCES OF HEAD-ON ENCOUNTER LOCATIONS FROM THE CANAL BOUNDARY: EACH ENTRY SPECIFIES THE DIFFERENCE (SUBTRACTION) BETWEEN THE MEDIAN DISTANCES FOR THE MAXENTIRL OR SOCIAL TRAJECTORY PLANNING AND THOSE FOR THE MINTRAVELTIME TRAJECTORY PLANNING

segment	among autonomous vessels		between autonomous and human-operated vessels	
	MaxEntIRL	Social	MaxEntIRL	Social
1	3.8 m	25.4 m	-2.0 m	9.6 m
2	2.5 m	6.2 m	2.8 m	2.6 m
3	2.4 m	4.3 m	3.3 m	2.8 m

encounters on social trajectories occur more frequently away from the canal boundary (Segment 1) or approximately uniformly across the entire range of observed distances (Segments 2 and 3). Also, from the statistics on the median distance of the head-on encounter locations from the canal boundary summarized in Table IV, we observe that the social trajectory planning attains the largest median value, which suggests that the autonomous vessels adopting the social trajectory planning tend to encounter one another further away from the canal boundary.

The outcomes of the analysis suggest that the autonomous vessels navigating along social trajectories experience head-on encounters with other autonomous vessels (also adopting social trajectories) less frequently. When they occur, the encounters take place further away from the canal boundary, which implies that the vessels would have more room to avoid vessel-to-vessel collisions.

2) Analysis on Head-On Encounters Between Autonomous and Human-Operated Vessels: We repeat a similar analysis as in Section VI-C1 between autonomous and human-operated vessels to validate our assertion that the social trajectory planning incurs less head-on encounters between the two different kinds of vessels and when they occur, the encounters take place further away from the canal boundary. As a result, in conjunction with the results from Section VI-C1, we conclude that our approach improves canal navigation safety.

We examine approximately 3×10^6 different cases in all three segments to assess head-on encounters between autonomous and human-operated vessels. For the analysis, we use the same trajectories as in Section VI-C1, as depicted in Fig. 9, to assign trajectories to the autonomous vessels and trajectory data from the validation dataset to assign trajectories to the human-operated vessels. We assess the head-on encounter frequency and location by following the same method as in Section VI-C1.

From the statistics summarized in Table III, as in the first scenario, we can observe that the social trajectory planning results in significantly less head-on encounters than the two other approaches in all three segments ($p < 0.001$, chi-square test). Moreover, surprisingly in the case of the MaxEntIRL trajectory planning, the autonomous vessels undergo more frequent head-on encounters with the human-operated vessels in Segments 2 and 3 than the MinTravelTime trajectory planning.

In addition, according to the statistics summarized in Table IV and the histograms depicted in Fig. 10(b), we observe that the encounters on the social trajectories tend to occur further away from the canal boundary than those on MinTravelTime trajectories. Unlike the first scenario, the encounters on the MaxEntIRL trajectories occur at similar distances or further away from the

canal boundary than the social trajectories in Segments 2 and 3; however, they attain the highest encounter frequency among the three types of trajectories.

Based on the analysis, in conjunction with the results from Section VI-B, we assert that our approach finds trajectories that resemble those of human-operated vessels, but at the same time avoids traveling areas where human-operated vessels are navigating in opposite directions, which leads to the significant reduction on head-on encounters with them. On the other hand, the observation that the MaxEntIRL trajectory planning approach attains the highest encounter frequency, especially in Segments 2 and 3, reiterates the importance of the feature selection in this approach for improving canal navigation safety.

D. Discussions and Remarks

The experimental results for both of the scenarios, as presented in Sections VI-C1 and VI-C2, suggest that the social trajectory planning would reduce the frequency of head-on encounters. To explain a possible reason behind these observations, recall that our approach finds the trajectories that tend to cross the locations where the value of $p(\mathrm{u} \mid \mathrm{z})$ is high and avoid traversing the locations where the value of $p(\mathrm{u} \mid \mathrm{z})$ is low. Hence, the autonomous vessels navigating along social trajectories tend to align more with the movements of human-operated vessels and avoid entering areas where human-operated vessels are frequently navigating in different directions. Consequently, our approach enables the autonomous vessels to experience less head-on encounters with other vessels.

Notably, in the second scenario, the improvement on the frequency of the encounters and the median distance of the encounter locations from the canal boundary are less substantial than those attained in the first scenario. Based on the analysis given in Section VI-B, we note that some trajectory data from the validation dataset are dissimilar from the trajectories computed by the social trajectory planning, which incurred the mean Euclidean distance of 7.4 m. Such dissimilarity would contribute to the smaller improvement.

As we have discussed in Section VI-B, we hypothesize that such noticeable Euclidean distance occurred because the social trajectory planning uses state-dependent control policies to generate trajectories, which would not be the case when human operators select trajectories to maneuver their vessels toward destinations. One way to generalize the class of control policies used in the social trajectory planning and to further improve its predictability performance, which we set as a future direction of this work, is to use control policies μ that assign the vessel velocity v_t based on the vessel's past trajectory $(\mathrm{x}_{t-d+1}, \ldots, \mathrm{x}_t)$ of length d, i.e., $\mathrm{v}_t = \mu(\mathrm{x}_{t-d+1}, \ldots, \mathrm{x}_t)$. It will be straight-forward to extend the formulation (3) to accommodate such class of control policies by replacing the probability density function $p(\mathrm{v}_t \mid \mathrm{x}_t)$ used to define the KL control cost with $p(\mathrm{v}_t \mid \mathrm{x}_{t-d+1}, \ldots, \mathrm{x}_t)$. In this case, a larger vessel trajectory dataset would be required to estimate the new function $p(\mathrm{v}_t \mid \mathrm{x}_{t-d+1}, \ldots, \mathrm{x}_t)$, and the computational complexity of the value iteration will increase exponentially with respect to the parameter d. Hence, the future research direction will involve seeking a new solution that mitigates both the sample and computational complexity.

In Section VI, we have partitioned the canal map into segments and computed social trajectories within individual segments. This was not only to conduct the experiments over segments of different lengths and sizes but also to reduce the computational complexity in solving the value iteration. One possible way to find a trajectory for a given pair of origin and destination points, lying in different canal segments, is first to identify all sequences of segments that connect the origin and destination, and to define intermediate points that lie at the intersection of every pair of neighboring segments. Then, by computing social trajectories between every suitable pair of the intermediate points and by connecting the computed trajectories, we can obtain a longer trajectory that traverses from the origin to the destination.

VII. Conclusion

In this article, we proposed the social trajectory planning that enables autonomous vessels to perform safe navigation in canal environments. The key idea is to adopt the optimal control formulation in which the cost function is designed to find trajectories that resemble those of human-operated vessels. Through the experiments using the AIS trajectory dataset, we validated the effectiveness of the proposed approach in improving safety in canal navigation. Important future directions of this article include improving the predictability performance of the social trajectory planning and incorporating obstacle avoidance algorithms to perform real-world canal navigation experiments.

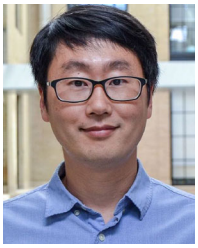
Acknowledgment

The authors would like to thank the anonymous reviewers for their constructive comments that led to substantial improvements of this article.

References

[1] "Urban waterways in global cities," Chicago Council Global Affairs, Chicago, IL, USA, Forum Rep., Jun. 2017.
[2] Roboat. [Online]. Available: http://roboat.org
[3] D. Sadigh, N. Landolfi, S. S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for cars that coordinate with people: Leveraging effects on human actions for planning and active information gathering over human internal state," *Auton. Robots*, vol. 42, no. 7, pp. 1405–1426, Oct. 2018.
[4] J. van den Berg, S. J. Guy, M. Lin, and D. Manocha, "Reciprocal n-body collision avoidance," in *Robotics Research*, C. Pradalier, R. Siegwart, and G. Hirzinger, Eds. Berlin, Germany: Springer, 2011, pp. 3–19.
[5] J. Guzzi, A. Giusti, L. M. Gambardella, and G. A. Di Caro, "Local reactive robot navigation: A comparison between reciprocal velocity obstacle variants and human-like behavior," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 2622–2629.
[6] Y. Luo, P. Cai, A. Bera, D. Hsu, W. S. Lee, and D. Manocha, "PORCA: Modeling and planning for autonomous driving among many pedestrians," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3418–3425, Oct. 2018.
[7] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robot. Auton. Syst.*, vol. 57, no. 5, pp. 469–483, May 2009.
[8] H. Kretzschmar, M. Spies, C. Sprunk, and W. Burgard, "Socially compliant mobile robot navigation via inverse reinforcement learning," *Int. J. Robot. Res.*, vol. 35, no. 11, pp. 1289–1307, Jul. 2016.
[9] B. Kim and J. Pineau, "Socially adaptive path planning in human environments using inverse reinforcement learning," *Int. J. Social Robot.*, vol. 8, no. 1, pp. 51–66, Jan. 2016.

[10] P. Henry, C. Vollmer, B. Ferris, and D. Fox, "Learning to navigate through crowded environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 981–986.
[11] Y. F. Chen, M. Everett, M. Liu, and J. P. How, "Socially aware motion planning with deep reinforcement learning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2017, pp. 1343–1350.
[12] B. D. Ziebart, A. L. Maas, A. K. Dey, and J. A. Bagnell, "Navigate like a cabbie: Probabilistic reasoning from observed context-aware behavior," in *Proc. 10th Int. Conf. Ubiquitous Comput.*, 2008, pp. 322–331.
[13] B. D. Ziebart *et al.*, "Planning-based prediction for pedestrians," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2009, pp. 3931–3936.
[14] M. Wulfmeier, D. Rao, D. Z. Wang, P. Ondruska, and I. Posner, "Large-scale cost function learning for path planning using deep inverse reinforcement learning," *Int. J. Robot. Res.*, vol. 36, no. 10, pp. 1073–1087, Aug. 2017.
[15] W. Schwarting, J. Alonso-Mora, and D. Rus, "Planning and decision-making for autonomous vehicles," *Annu. Rev. Control, Robot., Auton. Syst.*, vol. 1, no. 1, pp. 187–210, Jan. 2018.
[16] R. A. Knepper and D. Rus, "Pedestrian-inspired sampling-based multi-robot collision avoidance," in *Proc. IEEE RO-MAN: 21st IEEE Int. Symp. Robot Human Interact. Commun.*, 2012, pp. 94–100.
[17] R. A. Knepper, S. S. Srinivasa, and M. T. Mason, "Toward a deeper understanding of motion alternatives via an equivalence relation on local paths," *Int. J. Robot. Res.*, vol. 31, no. 2, pp. 167–186, Feb. 2012.
[18] C. I. Mavrogiannis and R. A. Knepper, "Multi-agent path topology in support of socially competent navigation planning," *Int. J. Robot. Res.*, vol. 38, no. 2/3, pp. 338–356, Jun. 2019.
[19] P. Trautman, J. Ma, R. M. Murray, and A. Krause, "Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation," *Int. J. Robot. Res.*, vol. 34, no. 3, pp. 335–356, Feb. 2015.
[20] P. Guan, M. Raginsky, and R. M. Willett, "Online Markov decision processes with Kullback-Leibler control cost," *IEEE Trans. Autom. Control*, vol. 59, no. 6, pp. 1423–1438, Jun. 2014.
[21] Automatic Identification System (AIS) Overview. [Online]. Available: https://www.navcen.uscg.gov/?pageName=aismain
[22] E. Alpaydin, *Introduction to Machine Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 2004.
[23] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. I, 3rd ed. Belmont, MA, USA: Athena Scientific, 2007.
[24] R. Munos and A. Moore, "Variable resolution discretization in optimal control," *Mach. Learn.*, vol. 49, no. 2, pp. 291–323, Nov. 2002.
[25] B. D. Ziebart, A. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Proc. 23rd Nat. Conf. Artif. Intell.*, Jul. 2008, vol. 3, pp. 1433–1438.
[26] A. D. Dragan, K. C. T. Lee, and S. S. Srinivasa, "Legibility and predictability of robot motion," in *Proc. 8th ACM/IEEE Int. Conf. Human-Robot Interact.*, Mar. 2013, pp. 301–308.
[27] A. N. Cockcroft, "The circumstances of sea collisions," *J. Navigat.*, vol. 35, no. 1, pp. 100–112, Jan. 1982.



Shinkyu Park (Member, IEEE) received the B.S. degree in electrical engineering from Kyungpook National University, Daegu, South Korea, in 2006, the M.S. degree in electrical engineering from Seoul National University, Seoul, South Korea, in 2008, and the Ph.D. degree in electrical engineering from the University of Maryland, College Park, MD, USA, in 2015.

In 2016, he was a Postdoctoral Researcher with the National Geographic Society, Washington, DC, USA. From 2016 to 2019, he was a Postdoctoral Associate with the Massachusetts Institute of Technology, Cambridge, MA, USA. Since 2019, he has been an Associate Research Scholar with Princeton University, Princeton, NJ, USA, working in cross-departmental robotics projects. His current research interests include robotics, multirobot control, control theory, and game theory.

Michal Cap received the M.Sc. degree in agent technology from Utrecht University, Utrecht, The Netherlands, in 2010, and the Ph.D. degree in artificial intelligence from Czech Technical University (CTU) in Prague, Prague, Czech Republic, in 2017.

Between 2015 and 2016, he held the position of a Fulbright-funded Visiting Researcher with the Massachusetts Institute of Technology, Cambridge, MA, USA. Then, in 2018, he was a Postdoctoral Fellow with the Department of Cognitive Robotics, TU Delft, Delft, The Netherlands. He is currently a Tech Lead in isee, Cambridge, MA, USA, an MIT spin-off company developing self-driving technology, and a Research Associate with CTU in Prague.

Javier Alonso-Mora (Senior Member, IEEE) received the Ph.D. degree in robotics from ETH Zurich, Zürich, Switzerland, in 2014.

He is currently an Associate Professor with the Delft University of Technology, Delft, The Netherlands, where he leads the Autonomous Multi-Robots Laboratory, and a Principal Investigator with the Amsterdam Institute for Advanced Metropolitan Solutions, Amsterdam, The Netherlands. During the Ph.D., he worked in a partnership with Disney Research Zurich, ETH Zurich. He was a Postdoctoral Associate with the Massachusetts Institute of Technology, Cambridge, MA, USA. His main research interests include motion planning and control of autonomous mobile robots and multirobot systems interacting in dynamic environments and on-demand transportation.

Dr. Alonso-Mora was the recipient of multiple prizes and grants, including the ICRA Best Paper Award on Multi-Robot Systems in 2019 and a Talent Scheme Veni Award from the Netherlands Organisation for Scientific Research in 2017.

Carlo Ratti received the Ph.D. degree in architecture from the University of Cambridge, Cambridge, U.K., in 2001.

He is currently an Architect and Engineer, who practices architecture in Turin, and teaches at the Massachusetts Institute of Technology, Cambridge, MA, USA, where he also directs the SENSEable City Laboratory. His research interests include urban design, human–computer interfaces, electronic media, and the design of public spaces.

Dr. Ratti is a member of the Ordine degli Ingegneri di Torino, the Architects Registration Board (U.K.), and the Association des Anciens Eléves de l'École Nationale des Ponts et Chaussáes.

Daniela Rus (Fellow, IEEE) received the Ph.D. degree in computer science from Cornell University, Ithaca, NY, USA, in 1992.

She is currently the Andrew and Erna Viterbi Professor of electrical engineering and computer science, the Director of the Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, and the Deputy Dean of Research with the Schwarzman College of Computing, MIT. She is also a Visiting Fellow with The MITRE Corporation, McLean, VA, USA. Her research aims toward the science and engineering of autonomy, with a focus on developing new approaches to robot design, fabrication, control, planning, and learning.

Dr. Rus is a Class of 2002 MacArthur Fellow, a Fellow of ACM and AAAI, and a member of the National Academy of Engineering and American Academy of Arts and Sciences. She was the recipient of the Engelberger Award for robotics.