2-2019

Multiagent decision making for maritime traffic management

ARAMBAM JAMES SINGH
Singapore Management University, arambamjs.2016@phdis.smu.edu.sg

Duc Thien NGUYEN
Singapore Management University, dtnguyen.2014@phdis.smu.edu.sg

Akshat KUMAR
Singapore Management University, akshatkumar@smu.edu.sg

Hoong Chuin LAU
Singapore Management University, hclau@smu.edu.sg

Citation

ARAMBAM JAMES SINGH; NGUYEN, Duc Thien; KUMAR, Akshat; and LAU, Hoong Chuin. Multiagent decision making for maritime traffic management. (2019). *Proceedings of the Thirty-Third Conference on Artificial Intelligence 2019: Honolulu, HI, January 27 – February 1*. 6171-6178.
Available at: https://ink.library.smu.edu.sg/sis_research/4887

The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)

Multiagent Decision Making For Maritime Traffic Management

Arambam James Singh, Duc Thien Nguyen, Akshat Kumar, Hoong Chuin Lau
School of Information Systems
Singapore Management University
{arambamjs.2016,dtnguyen.2014,akshatkumar,hclau}@smu.edu.sg

Abstract

We address the problem of maritime traffic management in busy waterways to increase the safety of navigation by reducing congestion. We model maritime traffic as a large multiagent systems with individual vessels as agents, and VTS authority as the regulatory agent. We develop a maritime traffic simulator based on historical traffic data that incorporates realistic domain constraints such as uncertain and asynchronous movement of vessels. We also develop a traffic coordination approach that provides speed recommendation to vessels in different zones. We exploit the nature of collective interactions among agents to develop a scalable policy gradient approach that can scale up to real world problems. Empirical results on synthetic and real world problems show that our approach can significantly reduce congestion while keeping the traffic throughput high.

1 Introduction

We address the problem of maritime traffic management in busy waterways such as the Singapore Strait and Tokyo Bay. The Singapore Strait is one of the busiest shipping lane in the world connecting the Indian ocean and the South China Sea. Vessel traffic in Strait has been consistently increasing (Hand 2017) with approximately 2000 merchant vessels (such as oil and gas tankers) crossing it daily (Liang and Maye-E 2017). Increased traffic affects safety of navigation as well as impacts the maritime ecosystem with oil and gas spills (Lim 2017; Tan 2017). Congestion in narrow waterways critically affects the safety of navigation as it leads to frequent evading maneuverer from vessels and increases the cross traffic (Segar 2015). Therefore, the key research question we address is how to coordinate maritime traffic in heavily trafficked narrow waterways, such as the Singapore Strait and Tokyo Bay, to increase the safety of navigation by reducing traffic hotspots.

Figure 1 shows the e-navigation chart (ENC) of a strait. The ENC is composed of several features such as anchorages where vessels anchor and wait for services, berths, pilot boarding grounds, and the traffic separation scheme or *TSS*. The TSS (figure 1) is the set of mandatory unidirectional routes designed to reduce collision risk among vessels transitioning through or entering the Strait. The TSS is responsible for carrying the bulk of the maritime traffic. Therefore, we focus on traffic coordination in the TSS.

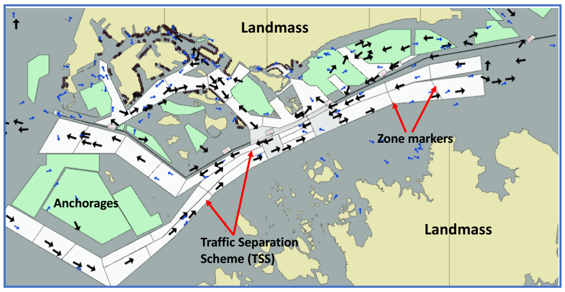


Figure 1: Electronic navigation chart (ENC) of strait near a large asian city with color-coded features (best viewed electronically)

Based on geographical features, the TSS can be further divided into smaller *zones* (as shown in figure 1). Our goal is to compute the recommended time taken to cross each zone based on the current traffic in other zones such that (a) traffic intensity is within some pre-defined limit (which increases safety of navigation), and (b) maximize traffic throughput while maintaining the safety of navigation. E.g., if the path leading to berths is crowded (or the count of vessels is high), we may slow down vessels entering TSS to regulate the traffic. We develop both the maritime traffic simulator and traffic control approaches.

Domain constraints: Our work is motivated by road traffic light control using multiagent RL (Wiering 2000; Bakker et al. 2010). However there are several differences in the maritime domain that necessitate the development of new methods for the maritime traffic control. First, vessels, unlike cars, can never fully stop in the TSS. They always have to maintain a *minimum* cruising speed, and have a *maximum* speed limit. Without the minimum speed, they risk overturning due to their sheer size, and the effects of water movement. Similarly, the TSS, although looks similar to a road lane, there are major differences. Unlike road segments, vessel traffic in a TSS zone cannot be neatly arranged in a single queue. Vessels move dynamically, often sail in parallel to each other and overtake each other. Finally, vessel movement is highly dynamic in the water, and is affected by the ship condition, and weather conditions such as wind, rain and tides. Therefore, even if we provide a recommendation to a vessel to cross a particular zone in T minutes, the actual time taken to cross is stochastic. It is not clear apriori how should we parameterize such uncertainty in the vessel movement. Hence, a micro-level simulation of the maritime traffic that simulates the precise position of each vessel is very challenging. In our work, we accommodate such unique maritime domain constraints in our traffic simulator and traffic coordination techniques.

Related work in maritime traffic management: For maritime traffic optimization, most current works either involve high fidelity commercial simulation tools to model micro-level navigation characteristics of vessels (Marin 2018), and expert systems and rule-based approaches to model the macro-level behavior of the traffic (Hasegawa et al. 2001; Hasegawa 1993; Ince and Topuz 2004). However, enhancing safety of navigation in a geographically constrained heavy traffic area, requires statistical modeling and learning from large amounts of historical data. Rule-based expert systems are not sufficient to resolve every possible close quarter situation in the heavily trafficked Strait.

Related work in multiagent planning and learning: Our work can be cast as a decentralized partially observable MDP (Dec-POMDP) (Bernstein et al. 2002) which is a rich framework for sequential multiagent decision making. However, solving even 2-agent Dec-POMDP is computationally challenging, being NEXP-Hard (Bernstein et al. 2002). To address scalability, previous works have explored several restricted variations of Dec-POMDPs (Becker et al. 2004; Spaan and Melo 2008; Witwicki and Durfee 2010). Recent works have focused on models where agent interactions are primarily dependent on agents' "collective influence" on each other rather than their identities (Varakantham, Adulyasak, and Jaillet 2014; Sonu, Chen, and Doshi 2015; Robbel, Oliehoek, and Kochenderfer 2016; Nguyen, Kumar, and Lau 2017a; 2017b). In our work, we also explore this direction as vessels in maritime traffic can be considered homogenous affecting each other only via their collective presence (such as congestion). Collective decentralized POMDPs ($\mathbb{C}$Dec-POMDPs) have been proposed to model such collective multiagent planning models (Nguyen, Kumar, and Lau 2017a). Existing works in $\mathbb{C}$Dec-POMDPs assume that all agents act in synchronous manner with fixed duration actions (Nguyen, Kumar, and Lau 2017a; 2018). We extend the $\mathbb{C}$Dec-POMDP model to handle asynchronous agent behavior with variable duration actions which helps to model real word settings (e.g., navigation to another zone has stochastic duration). There are multiagent planning models with variable durations actions (Amato, Konidaris, and Kaelbling 2014). However they are limited in scalability to a few agents as opposed to thousands of vessels or agents in the maritime domain. A deterministic scheduling approach exists for maritime traffic managment (Agussurja, Kumar, and Lau 2018). However this approach addresses a deterministic setting where each vessel follows the computed schedule exactly without any deviation. In our model, we address a more realistic model of vessel navigation based on stochastic duration actions.

In our work, we contribute the design and development of a maritime traffic simulator, and traffic control approaches. Our simulator addresses several maritime domain constraints as highlighted earlier, and also allows for variable duration actions. We have access to 4 month historical AIS (Automatic Identification System) data containing timestamped position, speed over ground, direction, and navigation status (e.g., at anchor, not under command) of each vessel roughly every 10 seconds. The total dataset contains more than 9 million unique records. We process this dataset and use it to learn and validate several parameters of our simulation model. We also develop maritime traffic control approaches using a policy gradient approach that exploits the collective nature of interactions among vessels. We test on synthetic domains and using our historical data based simulator to show that our approaches provide significant improvement over baseline approaches.

2 Model Definition

We next describe our model. The navigable sea space where traffic needs to be regulated (denoted as *port waters*) is divided into multiple zones $z \in Z$. We consider set of zones $\tilde{\mathcal{Z}} = Z \cup \{z_d\}$ including navigable zones $z \in Z$, and a dummy zone z_d. The dummy zone indicates that the vessel is outside the port waters. There are a total of M vessel agents. An agent m can be present in one of these zones. Zones can be arranged in the form of a directed acyclic graph in which each zone is a node, and edges correspond to traffic flow among zones. Such graph structure is typically determined by a VTS authority to separate outbound, inbound and transiting traffic. E.g., in figure 1, east-to-west and west-to-east are separated in the TSS.

Traffic enters the port waters via a set of specified source zones $Z_{\text{src}} \subset Z$ (e.g., extreme east or extreme west zones in figure 1). Agents terminate their journey at a set of terminal zones $Z_{\text{ter}} \subset Z$, which for example may represent berths, anchorages or transiting out of port water boundary. Vessels arrive in port waters over time; a vessel outside the port water boundary is assumed to be in a dummy zone z_d. We have discrete time, finite plan horizon H.

Traffic control: At each time step t, to regulate the congestion, a *traffic control agent* advises a speed $v_t^{zz'}$ for vessels moving from zone z to zone z'. We consider the speed recommendation as the output of a policy function parameterized using θ, $\pi_\theta^{zz'}(\bullet)$, taking input as joint-state of vessels currently in port water. The objective of the traffic control agent is to optimize the parameters θ of the policy function to optimize a congestion and delay based utility function (defined later).

Vessel Model: We model the behavior of each vessel as follows. Let s_t^m denote the state of vessel m at time t. Consider a vessel m currently inside the port waters. As the navigation action has a variable duration, this vessel can be categorized as *newly arrived* at some zone z at time t or *in-transit* through z at t.

- **In-transit:** $s_t^m = \langle z_t^m, z_t'^m, \tau^m \rangle$ where $z_t^m \in Z$ denotes vessel's current zone at time t, $z_t'^m \in Z$ is the next zone agent is heading to, and τ^m is the future time at which the vessel reaches the next zone $z_t'^m$.

- **Newly arrived:** When the vessel is newly arrived at zone z_t^m, its next zone z' and next arrival time τ are not yet determined, and its state s_t^m is denoted as $\langle z_t^m, \emptyset, \emptyset\rangle$.

Direction decision: When a vessel is *newly arrived* at zone z at time t ($s_t^m = \langle z, \phi, \phi\rangle$), it will decide the next zone z' from the distribution $\alpha(z'|z)$, and its action $a_t^m = z'$. In several ports, often the number of destinations vessels are heading to are small (e.g., berths, anchorages, transiting through). Often, there are only very few navigation routes to reach such destinations which are decided by factors such as the TSS, and hydrological features such as deep water routes for deep draft vessels. Therefore, unlike routing in road networks, we model the *average* navigation behavior of vessels. We learn the distribution $\alpha(z'|z)$ from historical data, and consider it as a fixed input model parameter.

When a vessel is *in-transit*, its action is *null* or $a_t^m = \emptyset$.

Arrival distribution: To model the arrival time τ of vessels into the source zones $z_{\text{src}} \in Z_{\text{src}}$ from outside the port waters, we assume a distribution $\{P(\langle z_d, z_{\text{src}}, \tau\rangle)\}_{z_{\text{src}} \in Z_{\text{src}}, \tau \in [1:H]}$. We estimate this distribution from historical data. The starting state $\langle z_d, z_{\text{src}}, \tau\rangle$ implies that the vessel moves to the source zone z_{src} at time τ. Before time τ, the vessel is outside the port waters. We assume that this arrival probability cannot be controlled as it is often determined by exogenous factors such as the schedule of the shipping line.

We have made the design choice to *pre-sample* the arrival time τ at the vessel's next destination because vessels can take variable amount of time to cross a zone. Therefore, at any instant in time, we have to record how many vessels are currently transiting through a zone to accurately compute the reward, and determine future adjustments of vessel speeds depending on the current traffic.

State transition function ϕ: We define the state transition function for a vessel as follows:

- If vessel m is currently outside the port waters or $s_t^m = \langle z_d, z_{\text{src}}, \tau\rangle$, then $\phi(s_{t+1}^m = \langle z_{\text{src}}, \phi, \phi\rangle | s_t^m) = 1$ if $t+1 = \tau$, otherwise zero. If $(t+1) < \tau$, then vessels remains in the same state $\langle z_d, z_{\text{src}}, \tau\rangle$ with probability 1.
- If vessel m is inside the port waters, and has *newly arrived* at a zone z at time t or $s_t^m = \langle z, \emptyset, \emptyset\rangle$, it would choose next zone z' from the distribution $\alpha(z'|z)$, and the arrival time τ at z' is sampled from the distribution $p^{\text{nav}}(\tau | z, z'; \beta_t^{zz'})$ where $\beta_t^{zz'}$ is the speed control parameter for moving from z to z' at time t. We will show later how β is determined, and the parametric form of distribution p^{nav}. Therefore, if $s_t^m = \langle z, \emptyset, \emptyset\rangle;\ \ s_{t+1}^m = \langle z, z', \tau\rangle$, then $\phi(s_{t+1}^m | s_t^m; \boldsymbol{\beta}_t) = \alpha(z'|z) \cdot p^{\text{nav}}(\tau | z, z'; \beta_t^{zz'})$.
- If agent m is in-transit from zone z to z' at time t or $s_t^m = \langle z, z', \tau\rangle$, then two cases can happen. At time $t+1$, the agent finally crosses zone z and reaches the starting point of zone z'. This setting occurs when $\tau = t+1$ (recall that τ denotes the arrival time at zone z'). Other case is the agent is still in-transit through zone z at time $t+1$. This occurs when $\tau > t+1$. The case $\tau < t+1$ is inconsistent as it implies the agent reaches its destination in the past.

$$\phi(s_{t+1}^m | s_t^m; \boldsymbol{\beta}_t) = \begin{cases} \mathbb{I}(s_{t+1}^m = \langle z', \emptyset, \emptyset\rangle) & \text{iff } \tau = t+1 \\ \mathbb{I}(s_{t+1}^m = \langle z, z', \tau\rangle) & \text{iff } \tau > t+1 \end{cases} \tag{1}$$

where $\mathbb{I}$ is the indicator function giving one or zero based on its input logical condition being true or false.

For ease of exposition, we have not shown the transition function for terminal zones $z_{\text{ter}} \in Z_{\text{ter}}$. When the agent first enters any z_{ter}, say at time t, its state is $\langle z_{\text{ter}}, \emptyset, \emptyset\rangle$. It is an absorbing state with zero reward, no outgoing transitions.

Count statistics: To summarize the joint activities of vessels $\boldsymbol{s}_t, \boldsymbol{a}_t = \langle s_t^m, a_t^m\ \forall m = 1:M\rangle$ at each time step t, we consider the aggregate statistics as follows:

- $\text{n}_t^{\text{txn}}(z, z', \tau) = \sum_{m=1}^M \mathbb{I}(s_t^m = \langle z, z', \tau\rangle)\ \forall z, z' \in Z, \tau > t$. It counts the vessels that are currently in-transit in zone z and will reach zone z' at time τ.
- $\text{n}_t^{\text{arr}}(z) = \sum_{m=1}^M \mathbb{I}(s_t^m = \langle z, \emptyset, \emptyset\rangle)\ \forall z \in Z$. It counts vessels that newly arrived in zone z at time t.
- $\text{n}_t^{\text{nxt}}(z, z') = \sum_{m=1}^M \mathbb{I}(s_t^m = \langle z, \emptyset, \emptyset\rangle; a_t^m = z')\ \forall z, z' \in Z$. It counts vessels that newly arrived in zone z at the current time t and decided to go to zone z'. We also have the consistency relation: $\text{n}_t^{\text{arr}}(z) = \sum_{z'} \text{n}_t^{\text{nxt}}(z, z')$
- $\tilde{\text{n}}_t(z, z', \tau) = \sum_{m=1}^M \mathbb{I}(s_t^m = \langle z, \phi, \phi\rangle, a_t^m = z', s_{t+1}^m = \langle z, z', \tau\rangle)\ \forall z, z' \in Z, \tau > t$. It counts vessels that newly arrived in zone z at the time t, decided to go to z', and reaching z' at time τ .
- From the above counts, we can compute the total number of agents present in each zone z at time t which include both newly arrived and in-transit agents as:

$$\text{n}_t^{\text{tot}}(z) = \text{n}_t^{\text{arr}}(z) + \sum_{z' \in Z, \tau = t+1:H} \text{n}_t^{\text{txn}}(z, z', \tau) \tag{2}$$

We arrange the above counts in the form of tables. E.g., $\text{n}_t^{\text{tot}} = (\text{n}_t^{\text{tot}}(z)\ \ \forall z \in Z)$. Analogously, we define the count tables $\text{n}_t^{\text{txn}}, \text{n}_t^{\text{arr}}, \text{n}_t^{\text{nxt}}, \tilde{\text{n}}_t$. We denote $\mathbf{n}_t = (\text{n}_t^{\text{txn}}, \text{n}_t^{\text{arr}}, \text{n}_t^{\text{nxt}}, \tilde{\text{n}}_t)$ to be the count table vector at each time t and $\mathbf{n}_{1:H}$ to be joint tables from time $t = 1$ to H.

Reward: The reward at time t depends on the aggregate count of agents in different zones. We treat each zone z as a limited capacity resource. The reward function balances the consumption of this resource and any potential delay caused to vessels. The capacity $cap(z)$ of this resource, for example, indicates how many maximum number of vessels can be *safely* present in the particular zone. For simplicity, we assume that each vessel consumes one unit of resource. When the capacity of the resource is violated, a penalty is imposed on each involved vessel. To also ensure that vessel reach their destination as soon as possible, there is a delay penalty per vessel for each time step. Hence, the reward r_t^m of a vessel m at zone z is computed as:

$$r_t^m = -C(z, \text{n}_t^{\text{tot}}) = -\left[w_r \cdot \max\left(\text{n}_t^{\text{tot}}(z) - cap(z),\ 0\right) + w_d\right] \tag{3}$$

where w_r and w_d are positive weights; w_r penalizes resource violation, and w_d penalizes delay for each vessel.

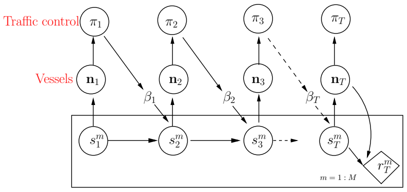


Figure 2: Dynamics of Maritime Traffic Control

The overall reward r can be computed by aggregating local rewards of vessels in different zones as follows:

$$r(\mathrm{n}_t) = -\sum_{z \in Z} \mathrm{n}_t^{\mathrm{tot}}(z) \cdot C(z, \mathrm{n}_t^{\mathrm{tot}}) \tag{4}$$

Traffic control objective: Let $\boldsymbol{s}_{1:T}, \boldsymbol{a}_{1:T} = \{s_{1:T}^m, a_{1:T}^m \; \forall m\}$ denote the joint T-step trajectory of agents resulting in counts $\mathbf{n}_{1:T}$. The objective function for the traffic control agent is to maximize:

$$V(\pi_\theta) = \sum_{T=1}^{H} \mathbb{E}_{\boldsymbol{s}_{1:T}, \boldsymbol{a}_{1:T}} [r(\mathrm{n}_T) | \boldsymbol{a}_T, \boldsymbol{s}_T; \pi_\theta] \tag{5}$$

Figure 2 shows the dynamic Bayesian net of our maritime traffic model with the traffic control agent (TCA) providing speed guidance using the policy π at each time step. The control policy π takes as input the joint counts $\mathbf{n}_t$ at each time step, and provides traffic control guidance $\boldsymbol{\beta}_t = \{\beta_t^{zz'} \forall z, z'\}$, which regulates vessel navigation using the distribution $p^{\mathrm{nav}}(\cdot | z, z'; \beta_t^{zz'}) \forall z, z'$.

Modeling vessel navigation behavior ($p^{\mathrm{nav}}, \beta^{zz'}$): A crucial aspect to address is modeling the navigation behavior of vessels. If a vessel navigating ($z \to z'$) is given an recommendation by the traffic control agent (TCA) to perform this entire navigation in μ time periods, then would this vessel finish this action in exactly μ time periods or require $\mu \pm \delta$ where δ is a random variable to take into account stochasticity of real world navigation? Furthermore, we must impose hard travel time limits $t_{\min}$ and $t_{\max}$ to model maximum and minimum speeds. As there are currently no controlled experiments with real vessels, there is no historical data to learn from. To address this issue and avoid making any unnecessary assumptions, we use the *principle of maximum entropy* (Maxent) (Jaynes 1957). The maxent principle advocates for the distribution that respects given constraints on parameters, but avoids making any other unnecessary assumption to avoid overfitting. Such maxent distributions have been used in computational sustainability to model dynamics of endangered species given that their observations are sparse and incomplete (Phillips, Anderson, and Schapire 2006).

We interpret the parameter $\beta^{zz'}$ as specifying the travel time $\Delta(\beta^{zz'})$ recommended by the TCA to move from z to z'. We *assume* that given this recommendation, the *average* travel time of vessels would be $\Delta(\beta^{zz'})$. However, the actual travel time of individual vessels can be different. Some may cross in less time than Δ and some may take more. Notice that our assumption is realistic. If on-an-average, vessels do not follow traffic guidance, then it would be virtually impossible to control the traffic. This is where the role of a VTS authority comes in—as a traffic regulatory authority, it may be possible to incentivise vessels to follow the navigation recommendations.

The travel time distribution $p^{\mathrm{nav}}(\tau | z, z'; \beta^{zz'})$ is the maximum entropy distribution with mean $\Delta(\beta^{zz'})$. It has been shown that the maxent discrete probability distribution with bounded positive support and a specified mean is the *binomial distribution* (Harremoes 2001). We incorporate hard limits $t_{\min}$ and $t_{\max}$ on the output of p^{nav} as follow.

Let current time be t. For $(z \to z')$, the arrival time at z' is $\tau = t + (t_{\min}^{zz'} + \tilde{\Delta})$. We sample $\tilde{\Delta}$ from the binomial distribution with $(t_{\max}^{zz'} - t_{\min}^{zz'})$ trials and success probability of each trial being $\beta_t^{zz'}$ (or the output of the TCA policy π). That is, $\tilde{\Delta} \sim B(t_{\max}^{zz'} - t_{\min}^{zz'}, \; \beta_t^{zz'})$. The average travel time is $\Delta(\beta^{zz'}) = t_{\min}^{zz'} + (t_{\max}^{zz'} - t_{\min}^{zz'})\beta_t^{zz'}$, a standard result for binomial distribution. In experiments, we provide empirical support by using historical data, and showing that our simulator built on the binomial distribution gives very close vessel traffic distribution to the real historical data for multiple days over peak traffic hours.

3 Generative Model for Counts

While, we can optimize objective (5) by sampling joint state-action trajectories of all the agents. This approach is not scalable as typically more than a thousand vessel cross the Strait each day. Integrating sampling of individual agent trajectories within a reinforcement learning based simulator is computationally intractable. Therefore, we reparameterize value function using counts $\mathbf{n}_{1:H}$, and show that counts are sufficient statistic for planning in our traffic control model. Sampling counts $\mathbf{n}_{1:H}$ is highly scalable as even if the number of vessels M increases, the dimensions of the count table remains fixed. Only the count of vessels in different buckets of count tables changes. As discussed in the previous section, let $\mathbf{n}_t = (\mathrm{n}_t^{\mathrm{txn}}, \mathrm{n}_t^{\mathrm{arr}}, \mathrm{n}_t^{\mathrm{nxt}}, \tilde{\mathrm{n}}_t)$ denote the count table vector. We first show that $\mathbf{n}_{1:H}$ are sufficient statistic for the joint distribution over state-actions trajectories of agents.

Theorem 1. *Count vector $\mathbf{n}_{1:H}$ is the sufficient statistics for the joint distribution $P(\boldsymbol{s}_{1:H}, \boldsymbol{a}_{1:H}; \pi)$*

Proof. Notice that a vessel takes the action to move to another zone z' and samples its travel duration from p^{nav} only when it is in a newly arrived state $\langle z, \phi, \phi \rangle$ (for some z). We can summarize this transition using the indicator function $\mathbb{I}(s_t^m = \langle z, \phi, \phi \rangle, a_t^m = z', s_{t+1}^m = \langle z, z', \tau \rangle)$. Rest of the state transitions are deterministic. E.g., when a vessel is in-transit, it moves to its destination z' at time τ with probability 1. We use this fact to aggregate vessels' states into the counts as follow:

$$\begin{aligned} P(\boldsymbol{s}_{1:H}, \boldsymbol{a}_{1:H}) = & \prod_{m=1}^{M} \prod_{t=1}^{H} \Big(\prod_{z, z', \tau} \big[\alpha(z'|z) \\ & \times p^{\mathrm{nav}}(\tau | z, z', \boldsymbol{\beta}_t = \pi_t(\mathbf{n}_t))\big]^{\mathbb{I}(s_t^m = \langle z, \phi, \phi \rangle, a_t^m = z', s_{t+1}^m = \langle z, z', \tau \rangle)} \Big) \\ = & \prod_t \Big(\prod_{z, z', \tau} \big[\alpha(z'|z) p^{\mathrm{nav}}(\tau | z, z', \boldsymbol{\beta}_t = \pi_t(\mathbf{n}_t))\big]^{\tilde{\mathrm{n}}_t(z, z', \tau)} \Big) \end{aligned} \tag{6}$$

where we used the fact that $\tilde{\mathrm{n}}_t(z, z', \tau) = \sum_{m=1}^{M} \mathbb{I}(s_t^m = \langle z, \phi, \phi \rangle, a_t^m = z', s_{t+1}^m = \langle z, z', \tau \rangle)$ from section 2. We can see that counts $\mathbf{n}$ are the sufficient statistic as (6) only depends on the counts generated by any $(\boldsymbol{s}_{1:H}, \boldsymbol{a}_{1:H})$. □

Generating counts: We next show the generative model for $\mathbf{n}_{t+1} = (\mathrm{n}^{\mathrm{arr}}_{t+1}, \mathrm{n}^{\mathrm{nxt}}_{t+1}, \tilde{\mathrm{n}}_{t+1}, \mathrm{n}^{\mathrm{txn}}_{t+1})$ given $\mathbf{n}_t$. Total vessels newly arriving in zone z' at time $t+1$, $\mathrm{n}^{\mathrm{arr}}_{t+1}(z')$, is given by the sum of vessels that were in-transit to z' at time t (or $\mathrm{n}^{\mathrm{txn}}_{t}(z, z', \tau = t+1)$), and newly arrived vessels in a zone z with next destination z' reaching z' at $t+1$ (or $\tilde{\mathrm{n}}_t(z, z', \tau = t+1)$).

$$\mathrm{n}^{\mathrm{arr}}_{t+1}(z') = \sum_{z} [\mathrm{n}^{\mathrm{txn}}_{t}(z, z', \tau = t+1) + \tilde{\mathrm{n}}_t(z, z', \tau = t+1)] \forall z' \tag{7}$$

$\mathrm{n}^{\mathrm{nxt}}_{t+1}$: Given $\mathrm{n}^{\mathrm{arr}}_{t+1}$, we can generate next zone counts $\mathrm{n}^{\mathrm{nxt}}_{t+1}(z, \cdot)$ from a multinomial distribution with parameters $p_{z'} = \alpha(z'|z) \forall z'$ as below:

$$\mathrm{n}^{\mathrm{nxt}}_{t+1}(z, \cdot) \mid \mathrm{n}^{\mathrm{arr}}_{t+1}(z) \sim \mathrm{Mul}(\mathrm{n}^{\mathrm{arr}}_{t+1}(z),\ p_{z'} \forall z') \tag{8}$$

$\tilde{\mathrm{n}}_{t+1}$: Next we sample the arrival time counts in destination zones z'. That is, for all newly arrived vessels at z moving to z', we sample the counts $\tilde{\mathrm{n}}_{t+1}(z, z', \cdot)$. Given that vessels' navigation time follows a binomial distribution (or p^{nav} is binomial), we sample from a multinomial distribution with parameters $p_\tau = p^{\mathrm{nav}}(\tau \mid z, z'; \beta^{zz'}_{t+1}) \forall \tau$

$$\tilde{\mathrm{n}}_{t+1}(z, z', \cdot) \mid \mathrm{n}^{\mathrm{nxt}}_{t+1}(z, z') \sim \mathrm{Mul}(\mathrm{n}^{\mathrm{nxt}}_{t+1}(z, z'), p_\tau \forall \tau) \tag{9}$$
$$\text{where, } \tau = t + 1 + t^{zz'}_{\min} + \tilde{\Delta}, \tilde{\Delta} \in [0, t^{zz'}_{\max} - t^{zz'}_{\min}]$$

Based on above counts, we compute all vessels that are in-transit to other zones z' at time $t+1$. It includes all vessels in-transit at time t reaching their destination at time $\tau > t+1$, and newly arrived vessels $\tilde{\mathrm{n}}_t(z, z', \tau)$ in-transit to z'.

$$\mathrm{n}^{\mathrm{txn}}_{t+1}(z, z', \tau) = \mathrm{n}^{\mathrm{txn}}_{t}(z, z', \tau) + \tilde{\mathrm{n}}_t(z, z', \tau) \forall z, z', \forall \tau > t+1 \tag{10}$$

Using above process, we can sample all counts $\mathbf{n}_{1:H}$ without sampling individual vessel trajectories. Sampling from such multinomial distributions remains efficient even if the vessel population increases. This makes such count-based sampling significantly more scalable than sampling individual agent trajectories. We show in appendix the exact distribution over counts or $P(\mathbf{n}_{1:H})$. We refer to constraints (7)-(10) which every count table must satisfy as $\Omega_{1:T}$.

4 Vessel-Based Value Function

As counts $\mathbf{n}_{1:H}$ are sufficient statistic for the distribution of joint state-action trajectories, and given the generative distribution $P(\mathbf{n}_{1:H})$ over counts, we have (proof in appendix):

Theorem 2. *The traffic control objective in* (5) *can be computed by expectation over counts*

$$V(\pi_\theta) = \sum_{t=1}^{H} \mathbb{E}_{\boldsymbol{s}_{1:t}, \boldsymbol{a}_{1:t}} [r(\mathbf{n}_t) | \boldsymbol{a}_t, \boldsymbol{s}_t; \pi_\theta] = \sum_{t=1}^{H} \mathbb{E}_{\mathbf{n}_{1:t} \in \Omega_{1:t}} [r(\mathbf{n}_t) | \pi_\theta]$$

We can directly optimize the above objective by computing gradient $\nabla_\theta V(\pi_\theta)$ using stochastic gradient ascent and moving parameters θ in the direction of the gradient. This strategy is similar to the well known REINFORCE policy gradient approach in RL (Williams 1992). However, we show empirically that this approach does not work well. The reason is the problem of multiagent credit assignment (Chang, Ho, and Kaelbling 2004; Bagnell and Ng 2006). That is, from the global reward signal r_t it is not clear which agent should get the credit or penalty for the overall traffic state. Instead, we consider a vehicle-based value function framework (Wiering 2000; Bakker et al. 2010) to train traffic control policy. Let $\pi = \langle \pi^{zz'}_\theta \forall z, z' \rangle$. Each $\pi^{zz'}_\theta$ outputs the speed control parameter $\beta^{zz'}_t$ at each time t. We assume the crossing zz' is like a traffic light, and compute the total accumulated reward (from time t till H), say $V^{zz'}_t(\pi^{zz'}_\theta)$, for those vessels that newly arrive at zone z at time t and decide to move to z'. Originally, in (Wiering 2000), vehicle-based method requires the joint state-actions of all vehicles at every time step.

$$V^{zz'}_t(\pi^{zz'}_\theta) = \mathbb{E}\big[\sum_{m=1}^{M} \mathbb{I}[s^m_t = \langle z, \emptyset, \emptyset \rangle, a^m_t = z'] \sum_{t'=t}^{H} r^m_{t'} \big| \pi_\theta \big] \tag{11}$$

Under this vessel-based traffic control framework, we optimize each $V^{zz'}_t(\pi^{zz'}_\theta)$ in an iterative fashion, similar to car-based value functions in (Wiering 2000). This is an approximate solution technique, but is known to produce good road traffic control policies (Bakker et al. 2010), and empirically, we observed it works significantly better than the REINFORCE method as $V^{zz'}_t$ performs effective credit assignment computing precisely the effectiveness of policy $\pi^{zz'}_\theta$ by filtering out the contributions from other zone pairs. In our model, we work at the abstraction of counts, and thus extracting joint state-action trajectories for each vessel is expensive, and not scalable for large agent population. We therefore develop a collective vessel-based value function mechanism to compute this value using only the counts.

Theorem 3. *The vehicle-based value function can be computed by collective expectation over the counts as follows:*

$$V^{zz'}_t(\pi^{zz'}_\theta) = \mathbb{E}_{\mathbf{n}_{1:H}} \big[\sum_{\tau > t} \tilde{\mathrm{n}}_t(z, z', \tau) V^{\mathbf{n}}_t(z, z', \tau) \big| \pi_\theta \big] \tag{12}$$

in which $V^{\mathbf{n}}_t(z, z', \tau)$ is the average accumulated reward of newly arrived vessels at z at time t going to z' computed based on the realized counts $\mathbf{n}_{1:H}$ as follows:

$$R^{\mathbf{n}}_t(z, z', \tau) = \sum_{\tau" = t}^{\tau - 1} -C(z, \mathrm{n}^{\mathrm{tot}}_{\tau"}), \forall \tau \in [t + t^{zz'}_{\min},\ \ t + t^{zz'}_{\max}] \tag{13}$$

$$V^{\mathbf{n}}_t(z, z', \tau) = R_t(z, z', \tau) + \gamma \cdot V^{\mathbf{n}}_\tau(z') \tag{14}$$

$$V^{\mathbf{n}}_t(z, z') = \frac{\sum_{\tau = t + t^{zz'}_{\min}}^{t + t^{zz'}_{\max}} V^{\mathbf{n}}_t(z, z', \tau) \cdot \tilde{n}_t(z, z', \tau)}{\sum_{\tau = t + t^{zz'}_{\min}}^{t + t^{zz'}_{\max}} \tilde{n}_t(z, z', \tau)} \tag{15}$$

$$V^{\mathbf{n}}_t(z) = \frac{\sum_{z'} \mathrm{n}^{\mathrm{nxt}}_t(z, z') \cdot V^{\mathbf{n}}_t(z, z')}{\sum_{z'} \mathrm{n}^{\mathrm{nxt}}_t(z, z')}, \tag{16}$$

where $R^{\mathbf{n}}_t(z, z', \tau)$ is the reward accumulated by a vessel when it is still in zone z between time t and τ; $V^{\mathbf{n}}_\tau(z, z')$ is *the average accumulated reward of a vessel which started crossing z to z' from time t. $V_\tau^{\mathbf{n}}(z')$ is the average accumulative reward of a vessel newly arrived at z' at time τ.*

Proof is provided in the appendix. Such vessel-based value function can be computed using a dynamic programming approach. We next compute the gradient of this vessel-based value function below (derivation in appendix):

Theorem 4. *The vehicle-based policy gradient for $\pi^{zz'}$ is*

$$\nabla_\theta V_1^{zz'}(\pi_\theta^{zz'}) = \mathbb{E}_{\mathbf{n}_{1:H}}\Big[\sum_{t=1:H}\sum_{\tau=t+t_{\min}^{zz'}}^{t+t_{\max}^{zz'}} \tilde{\mathrm{n}}_t(z,z',\,\tau)\times \big[(\tau - t - t_{\min}^{zz'})\cdot\nabla_\theta\log(\pi_\theta^{zz'}(\mathbf{n}_t)) + (t_{\max}^{zz'} - (\tau-t))\cdot\nabla_\theta\log(1-\pi_\theta^{zz'}(\mathbf{n}_t))\,\big]V_t^{\mathbf{n}}(z,z',\tau)\Big] \quad (17)$$

After computing above policy gradients, we can aggregate all $\nabla_\theta V_1^{zz'}$ and update the policy parameter θ as follows:

$$\theta^{new} \leftarrow \theta^{old} + \gamma\sum_{z,z'}\nabla_\theta V_1^{zz'}(\pi_\theta^{zz'}) \quad (18)$$

where γ is the learning rate. We call this approach as *vessel-based policy gradient* (or Vessel-PG). Our results are developed for the general setting where traffic control policy $\pi^{zz'}$ takes as input all the count information $\mathbf{n}_t$. However, empirically we observed that providing only total vessel counts in zone z and z', $(\mathrm{n}_t^{\text{tot}}(z), \mathrm{n}_t^{\text{tot}}(z'))$, as input provided higher quality solutions. Another benefit of such a policy is that it is easily implementable in a decentralized setting. Vessels have radars which can provide information about count of other vessels in their current zone z, and their next destination zone z'. Thus, vessels can query the policy π based on their local observations to get their speed control input.

5 Experimental Results

We perform experiments on both synthetic and real-world instances. Synthetic instances are for comparison against different methods by varying problem sizes, while real-world instances are used to evaluate effectiveness of our approach on mitigating hotspots within the strait. A detailed description about all experimental setups (policy structure, and other settings) are provided in the appendix.

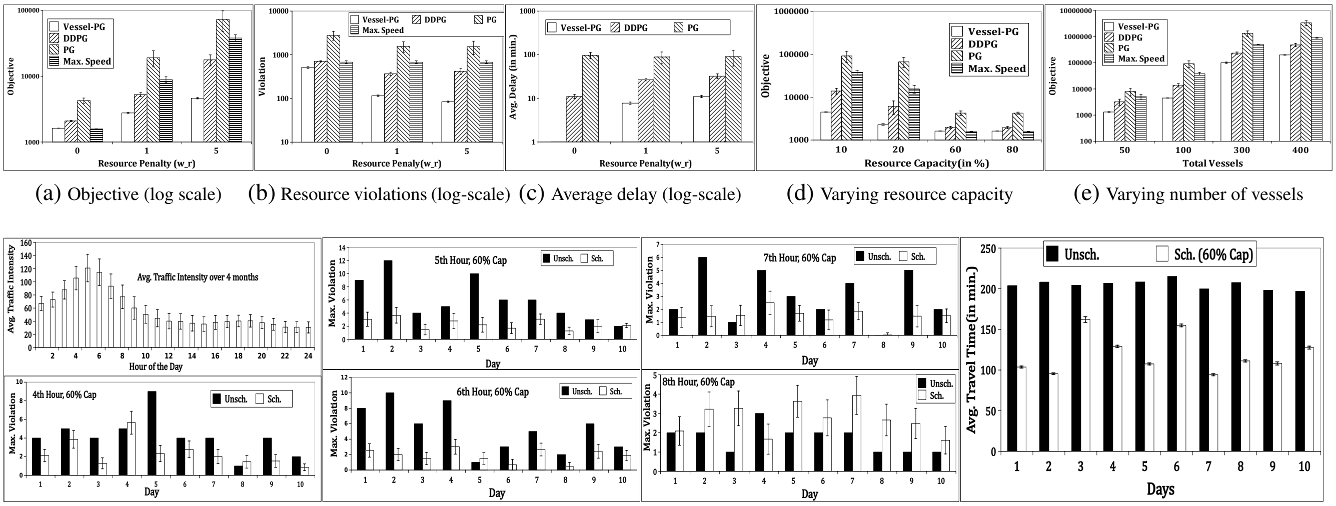
Baselines : We compare our approach Vessel-PG with three baselines—deep deterministic policy gradient (DDPG) (Lillicrap et al. 2015), policy gradient (PG) and MaxSpeed. As DDPG is for MDPs, we first extend the DDPG algorithm to our setting (details on this extension are in appendix). PG is standard policy gradient (REINFORCE) approach where we train with total empirical returns, and MaxSpeed policy is to always travel a zone with maximum uniform speed corresponding to $t_{\min}^{zz'}$ travel time for the zone pair.

Synthetic Data : For each synthetic instance, we generate a semi-random connected directed graph with edges representing zones, similar to (Agussurja, Kumar, and Lau 2018). With each edge is associated a minimum and a maximum time required to traverse the edge. Vessels arrive at traffic source edges with an arrival rate. The resources are the edge capacities (or the maximum number of vessels on an edge at any instant). Other problem settings are in the appendix.

Figures (3a-3c) show results with varying resource penalty w_r, 100 vessels and maximum capacity of each edge as 5. In all three, a lower value is better, y-axis is in log-scale. Figure 3b shows resource violations, figure 3c shows the total average delay over the MaxSpeed policy (i.e., if all vessels travel at the maximum speed $t_{\min}$, delay is zero); and figure 3a shows the overall objective we optimize in (3) (we convert rewards to costs, therefore lower cost is better). The resource violation value in figure 3b is sum of violations at each time step for the whole horizon. In figure 3c, MaxSpeed baseline is not shown as delay is 0. We also see similar behavior of close to 0 delay with Vessel-PG on w_r = 0, which is also intuitive as w_r = 0 ignores the resource violation component in the objective; DDPG also achieves a close to optimal policy, but has slightly higher resource violations than Vessel-PG for this case. Delay increases with increasing value of w_r because optimization preference shifts more towards violation. In all three settings of w_r, Vessel-PG achieves significantly better solution quality (i.e. lower delay). Similar behavior is also observed in figure 3b as violation of MaxSpeed, Vessel-PG and DDPG are similar at $w_r = 0$ and violations decreases with increasing resource penalty value. Crucially, Vessel-PG decreases resource violations faster than DDPG, and PG highlighting the effectiveness of our approach. Figure 3a results are the actual objective value that we optimize for, which subsumes both violation and delay components. We observe Vessel-PG achieves significantly better solution quality than rest of the baselines.

Figure 3d results are for 100 vessels, resource penalty $w_r = 5$ and $w_d = 1$. We set a threshold capacity as 50 for any edge (or 50% of total number of vessels), and vary actual resource capacity as a percentage of this threshold capacity. Y-axis shows the objective value (lower is better). In all the four settings of capacity% we see Vessel-PG achieving better solution quality than the rest, quality gap among all approaches reduces as the capacity% increases. This is because problem instance becomes easier as resource violations go to zero with increased capacity%. Figures 3e shows results with varying number of vessels, resource capacity as 5 for each edge and $w_r = 5, w_d = 1$. In this case also, we see Vessel-PG performing significantly better than the rest. Overall, results indicate that Vessel-PG performs best, much better than DDPG, PG does not work well, the reason being noisy empirical returns resulting in high variance gradient update and the multi-agent credit assignment problem.

Real Data: For real-world instance, we use 4 months historical AIS data of vessels voyaging in the strait of a large asian city. The AIS record contains a timestamp, vessel unique id, lat-long position, speed over ground, direction and navigation status (anchored/sailing etc). We have data for every few seconds for majority of vessel in the strait totaling about 9 million records. In this work we only consider tanker and cargo vessels which are the largest type of vessels causing hotspots. We tested our model on 20 busiest days, 10 days results are presented here, rest are in appendix. A brief description on parameters–we estimate $\alpha(z'|z)$ and zone ca-



(a) Objective (log scale) (b) Resource violations (log-scale) (c) Average delay (log-scale) (d) Varying resource capacity (e) Varying number of vessels

(f) Traffic intensity; 4th hour violations (g) 5th, 6th hour violations (h) 7th, 8th hour violations (i) Avg. travel time with varying capacity

Figure 3: (a-e) show results for synthetic instances (**lower value is better**). (f-i) show quality comparisons on historical data

pacities $cap(z)$ over 4 months data, $t_{\min}^{zz'}$ and $t_{\max}^{zz'}$ are also estimated, resource penalty $w_r = 50$ (after some trial-and-error this value worked best), delay penalty $w_d = 1$. We divide an hour period into 60 time steps (1 minute intervals); time step 0 is 12AM. Vessel's arrival rate are computed from data for each day. We experiment with 60% of historical $cap(z)$ to test how to reduce traffic intensity, and its impact on travel time.

Simulator Accuracy: We wanted to show that our simulator provides very similar peak traffic window and trend as the real data. We have computed root mean square error (RMSE) as a more concrete accuracy measure. For 12 (out of a total of 27 zones) high traffic intensity zones for peak hour window. On average over 4 months, the RMSE value is around 1.8, which intuitively means that on average there is a difference of 1.8 vessels between predicted count and observed count at each time step. On an average, around 50 vessels cross any of these 12 zones during the peak hour; thus the RMSE of 1.8 is relatively low. Figures are provided in appendix section 3

Peak traffic intensity reduction: Figures (3f-3i) show real data experiments. Figure 3f(top) shows traffic intensity for the whole planning area averaged over 4 months period, y-axis is number of unique vessels present in the planning area for each hour period, x-axis shows hours of the day. We can see that peak hours are at 4th, 5th and 6th. Therefore, for each day we apply our method (Vessel-PG) to control for this 3 hour window. Figures 3f(bottom)–3h(bottom) show results for 4th-8th hours, y-axis shows the maximum violation for that hour, x-axis shows the day number, legend **Sch** is our method and **Unsch** is the observed values from data. As noted, we control only 4th-6th hour window; for 7th and 8th hour we use $\beta_{data}^{zz'}$ to simulate future traffic for our method. We have added these two additional hours to show if there is any shift of peak hour, which would be undesirable. Results show significant reduction in violations for all 10 days on all hours except 8th hour (figure 3h(bottom)). Even though traffic intensity for 8th hour has increased by our method, the increase is only marginal, significantly less than the reduction in peak traffic intensity reduction for 4th-7th hour. Therefore, our traffic control strategy is highly effective, and does not shift peak traffic intensity.

Next we assess how traffic throughput is impacted by our traffic control method. Figure 3i shows average travel time a vessel would take if it entered the planning zone between 4th - 8th hour window (starting of our traffic control) traveling on the longest west-east route in TSS. We observe that travel time reduces with our method for all days significantly. Notice that our results imply that vessels should move at a higher speed (within the defined thresholds implied by $t_{\min}$, $t_{\max}$) within TSS, which would lead to a reduction in resource violations (implying safer traffic), and also would reduce travel time.

6 Conclusion

We addressed the problem of maritime traffic management in busy waterways of strait near a large asian city. Based on historical data, we have developed and validated a maritime traffic simulator. Using this simulator, which models aggregate behavior of vessels, we developed a policy gradient approach that provides speed guidance to vessels. Empirically, our approach works much better than competing approaches, and shows the potential of coordinating traffic for better navigation safety with high traffic throughput.

7 Acknowledgments

This research is supported by the Agency for Science, Technology and Research (A*STAR), Fujitsu Limited and the National Research Foundation Singapore as part of the A*STAR-Fujitsu- SMU Urban Computing and Engineering Centre of Excellence.

References

Agussurja, L.; Kumar, A.; and Lau, H. C. 2018. Resource-constrained scheduling for maritime traffic management. In *AAAI Conference on Artificial Intelligence*.

Amato, C.; Konidaris, G.; and Kaelbling, L. P. 2014. Planning with macro-actions in decentralized POMDPs. In *International conference on Autonomous Agents and Multi-Agent Systems*, 1273–1280.

Bagnell, D., and Ng, A. Y. 2006. On local rewards and scaling distributed reinforcement learning. In *Advances in Neural Information Processing Systems*, 91–98.

Bakker, B.; Whiteson, S.; Kester, L.; and Groen, F. C. A. 2010. *Traffic Light Control by Multiagent Reinforcement Learning Systems*. Springer Berlin Heidelberg. 475–510.

Becker, R.; Zilberstein, S.; Lesser, V.; and Goldman, C. V. 2004. Solving transition independent decentralized Markov decision processes. *Journal of Artificial Intelligence Research* 22:423–455.

Bernstein, D. S.; Givan, R.; Immerman, N.; and Zilberstein, S. 2002. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research* 27:819–840.

Chang, Y.; Ho, T.; and Kaelbling, L. P. 2004. All learning is local: Multi-agent learning in global reward games. In *Advances in Neural Information Processing Systems*, 807–814.

Hand, M. 2017. Malacca and S'pore Straits traffic hits new high in 2016, VLCCs fastest growing segment. http://www.seatrade-maritime.com/news/asia/malacca-and-s-pore-strait-traffic-hits-new-high-in-2016-vlccs-fastest-growing-segment.html.

Harremoes, P. 2001. Binomial and poisson distributions as maximum entropy distributions. *IEEE Transactions on Information Theory* 47(5):2039–2041.

Hasegawa, K.; Tashiro, G.; Kiritani, S.; and Tachikawa, K. 2001. Intelligent marine traffic simulator for congested waterways. In *IEEE International Conference on Methods and Models in Automation and Robotics*.

Hasegawa, K. 1993. Knowledge-based automatic navigation system for harbour manoeuvring. In *Ship Control Systems Symposium*, 67–90.

Ince, A. N., and Topuz, E. 2004. Modelling and simulation for safe and efficient navigation in narrow waterways. *Journal of Navigation* 57(1):53–71.

Jaynes, E. T. 1957. Information theory and statistical mechanics. *Phys. Rev.* 106(4):620–630.

Liang, A., and Maye-E, W. 2017. Busy waters around Singapore carry a host of hazards. https://www.navytimes.com/news/your-navy/2017/08/22/busy-waters-around-singapore-carry-a-host-of-hazards/.

Lillicrap, T. P.; Hunt, J. J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; and Wierstra, D. 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Lim, V. 2017. 300 tonnes of oil spilled after Singapore-registered ship collides with vessel off Johor. http://www.channelnewsasia.com/news/singapore/300-tonnes-of-oil-spilled-after-singapore-registered-ship-collid-7537142.

Marin. 2018. MARIN: Vessel Traffic Service (VTS) Simulators. http://www.marin.nl/web/Facilities-Tools/Simulators/Simulator-Sales/Vessel-Traffic-Service-VTS-Simulators.htm.

Nguyen, D. T.; Kumar, A.; and Lau, H. C. 2017a. Collective multiagent sequential decision making under uncertainty. In *AAAI Conference on Artificial Intelligence*, 3036–3043.

Nguyen, D. T.; Kumar, A.; and Lau, H. C. 2017b. Policy gradient with value function approximation for collective multiagent planning. In *Advances in Neural Information Processing Systems*, 4322–4332.

Nguyen, D. T.; Kumar, A.; and Lau, H. C. 2018. Credit assignment for collective multiagent RL with global rewards. In *Advances in Neural Information Processing Systems*.

Phillips, S. J.; Anderson, R. P.; and Schapire, R. E. 2006. Maximum entropy modeling of species geographic distributions. *Ecological Modelling* 190(3):231 – 259.

Robbel, P.; Oliehoek, F. A.; and Kochenderfer, M. J. 2016. Exploiting anonymity in approximate linear programming: Scaling to large multiagent MDPs. In *AAAI Conference on Artificial Intelligence*, 2537–2543.

Segar, M. 2015. Challenges of Navigating In Congested and Restricted Water. http://www.mpa.gov.sg/web/wcm/connect/www/968fafb8-7582-4091-9bcd-ec0a332f73a6/segar_challenges_of_navigating.pdf.

Sonu, E.; Chen, Y.; and Doshi, P. 2015. Individual planning in agent populations: Exploiting anonymity and frame-action hypergraphs. In *International Conference on Automated Planning and Scheduling*, 202–210.

Spaan, M. T. J., and Melo, F. S. 2008. Interaction-driven Markov games for decentralized multiagent planning under uncertainty. In *International COnference on Autonomous Agents and Multi Agent Systems*, 525–532.

Tan, A. 2017. Big cleanup of Singapore's north-eastern coast after oil spill. http://www.straitstimes.com/singapore/environment/big-cleanup-of-n-e-coast-after-oil-spill.

Varakantham, P.; Adulyasak, Y.; and Jaillet, P. 2014. Decentralized stochastic planning with anonymity in interactions. In *AAAI Conference on Artificial Intelligence*, 2505–2512.

Wiering, M. 2000. Multi-Agent reinforcement learning for traffic light control. In *International Conference on Machine Learning*, 1151–1158.

Williams, R. J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning* 8(3):229–256.

Witwicki, S. J., and Durfee, E. H. 2010. Influence-based policy abstraction for weakly-coupled Dec-POMDPs. In *International Conference on Automated Planning and Scheduling*, 185–192.