

---

This is an electronic reprint of the original article.  
This reprint may differ from the original in pagination and typographic detail.

Huotari, Janne; Ritari, Antti; Ojala, Risto; Vepsäläinen, Jari; Tammi, Kari  
**Q-Learning Based Autonomous Control of the Auxiliary Power Network of a Ship**

*Published in:*  
IEEE Access

*DOI:*  
[10.1109/ACCESS.2019.2947686](https://doi.org/10.1109/ACCESS.2019.2947686)

Published: 30/10/2019

*Document Version*  
Publisher's PDF, also known as Version of record

*Published under the following license:*  
CC BY

*Please cite the original version:*  
Huotari, J., Ritari, A., Ojala, R., Vepsäläinen, J., & Tammi, K. (2019). Q-Learning Based Autonomous Control of the Auxiliary Power Network of a Ship. *IEEE Access*, 7, 152879-152890. Article 2169-3536.  
<https://doi.org/10.1109/ACCESS.2019.2947686>

---

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

Received August 28, 2019, accepted October 8, 2019, date of publication October 16, 2019, date of current version October 31, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2947686

# Q-Learning Based Autonomous Control of the Auxiliary Power Network of a Ship

JANNE HUOTARI<sup>1</sup>, ANTTI RITARI, RISTO OJALA<sup>1</sup>, JARI VEPSÄLÄINEN<sup>1</sup>,  
AND KARI TAMMI<sup>1</sup>, (Member, IEEE)

Department of Mechanical Engineering, Aalto University, 02150 Espoo, Finland

Corresponding author: Janne Huotari (janne.huotari@aalto.fi)

This work was supported by the Business Finland-funded collaborative research project: INTENS.

**ABSTRACT** We present a reinforcement learning (RL) model that is based on Q-learning for the autonomous control of ship auxiliary power networks. The development and application of the proposed model is demonstrated using a case-study ship as the platform. The auxiliary power network of the ship is represented as a Markov Decision Process (MDP). Q-learning is then used to teach an agent to operate in this MDP by choosing actions in each operating state which would minimize fuel consumption while also respecting the boundary conditions of the network. The presented work is based on an extensive data set received from one of the cruise-line operators on the Baltic Sea. This data set was preprocessed to extract information for the state representation of the auxiliary network, which was used for training and validating the model. As a result, it is shown that the developed method produces an autonomous control policy for the auxiliary power network that outperforms the current human operated manual control of the case-study ship. An average of 0.9 % fuel oil savings are attained over the analyzed round-trips with control that displayed similar robustness against blackouts as the current operation of the ship. This amounts to 32 tons of fuel oil saved annually. In addition, it is shown that the developed model can be reconfigured for different levels of robustness, depending on the preferred trade-off between maintained reserve power and fuel savings.

**INDEX TERMS** Autonomous shipping, energy consumption reduction, Ferry, machinery, Q-learning, reinforcement learning, ship.

## I. INTRODUCTION

Increasing energy efficiency and autonomous operation are currently two major trends in the shipping industry. In response to accords, such as the Paris Agreement [1], being signed, industrial leaders must develop more energy efficient operation techniques. Although the Paris Agreement does not specifically apply to the shipping sector, the International Maritime Organization has set its own strategy on the best ways to reach similar goals [2]. Reaching these goals depends on the development of various measures that would reduce the amount of air pollutants generated by ships. These measures are analyzed in a comprehensive survey in [3].

Although autonomous shipping has been quite extensively studied, research has often focused on autonomous navigation or maneuvering of the ship as in [4] and [5]. Therefore, the autonomous control of the auxiliary power network of a ship has received far less attention. Autonomous

control of the auxiliary power network presents an opportunity to increase the overall autonomy of a ship. In this article, we present a methodology for controlling the auxiliary power network of a ship with the purpose of achieving autonomous control while reducing fuel consumption and retaining robustness against blackouts. The developed RL model is trained on data gathered from a ship operating on the Baltic Sea and uses Q-learning to create the control logic.

RL has been successfully applied in a multitude of areas where a supervised model would be limited due to a constantly changing environment, or difficulties in defining the model. One of the advantages of RL control algorithms is that they can be constantly adjusted as the algorithm operates, leading to a constantly learning algorithm. The aim is to create a smart and robust system, which is capable of providing the demanded power, as well as predicting future power demand and preliminarily adjusting the system accordingly.

This article is a continuation from the authors' previous work on the analysis of energy storage feasibility in an auxiliary power network of a ship. In that earlier paper, the

The associate editor coordinating the review of this manuscript and approving it for publication was Sudhakar Babu Thanikanti<sup>1</sup>.

optimal control was achieved via off-line optimization methods assuming that the operation and load cycles were known. The study resulted in the best case payback time of a battery and also suggested the battery sizing. [6]

This paper aims to achieve an online control methodology to find efficient usage patterns for on-board power generating equipment. In contrast to the last paper, we do not consider an energy storage to be a part of the auxiliary network. However, the authors plan to extend the control model described here to the control of the auxiliary network with an energy storage.

## II. BACKGROUND

### A. REINFORCEMENT LEARNING

RL [7] is a subgroup of machine learning, in which an agent learns to adopt actions in an environment that maximize a numerical reward signal. Contrary to other machine learning methods, in RL, the agent is not explicitly guided by some pre-trained policy, but must instead learn correct actions by trial and error. Thus, the agent in reinforcement learning is both the formulator of an optimal policy as well as the executor of actions.

The environment is often established as a finite Markov Decision Process (MDP) which is a collection of possible states and actions. The transition between states occurs through actions, one time-step increment at a time. It can be defined as a 4-tuple  $(S, A, P_t, R_t)$  where

- $S$  is a finite set of states  $s$ ,
- $A$  is a finite set of possible actions  $a$ ,
- $P_t = Pr\{s_{t+1} = s' | s_t = s, a_t = a\}$  is a set of probabilities of transitioning from state  $s$  to  $s'$  with action  $a$  at time-step  $t$  and
- $R_t$  is a set of immediate rewards  $r$  obtained by transitioning from state  $s$  to  $s'$  at time-step  $t$ .

The agent tries to learn an optimal policy  $\pi(s)$  that expresses which action in state  $s$  results in maximal reward  $r$  in the long run. The cumulation of rewards by following policy  $\pi(s)$  from state  $s$  is called the value function and can be expressed as

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a) [r + \gamma v_\pi(s')], \quad (1)$$

where  $v_\pi(s)$  is the value of state  $s$  when following a sequence of actions given by policy  $\pi(s)$ .  $\pi(a|s)$  is the probability of choosing action  $a$  in state  $s$  and  $\gamma$  is a discount factor for future rewards. The various methods for iterating an optimal policy for the devised MDP are based on Eq. 1. RL methods have been widely implemented into various control tasks with varying success.

One of the most famous implementations of RL was TD-Gammon; an algorithm developed in [8] to play backgammon that exceeded the level of the world's best human players. Computer games are often the platform on which state-of-the-art RL algorithms are developed, due to the rules in games being absolute, thus rendering them easy to formulate in terms of an MDP. Furthermore, the reward function is often easily determined as it can be tied to a score in the game or a similar construct.

RL has also been successfully utilized in a number of real-world control tasks. For example, in [9], a RL method was demonstrated to control the inverse hovering of a helicopter. The solution was based on first learning the dynamic model of the helicopter by applying supervised learning on flight data, and then searching for the optimal control policy. The authors especially noted the speed at which they were able to develop new kinds of controllers for the helicopter.

Additionally, RL methods have been widely implemented in the control of robotics. A comprehensive analysis of these applications is provided in survey [10]. The survey also stands as an exhaustive analysis of modern RL, emphasizing problems and aspects of RL that occur when applied to real-world problems. These include topics such as the curse of dimensionality, problems associated with continuous state-spaces and the difficulty of assigning suitable rewards for states.

### B. Q-LEARNING

Q-learning [11] is an off-policy, model-free reinforcement learning algorithm. The name, Q-learning, derives from the formulation of Q-functions:  $Q(s, a)$ , which are iteratively updated.  $Q(s, a)$  equals the value of taking action  $a$  in state  $s$ . In pseudocode, the complete Q-learning algorithm is as follows:

$$Q(s, a) = 0 \text{ for all } s \in S \text{ and } a \in A$$

$s = \text{starting state}$

**While** *current iteration* < *total iterations* **Do**

Choose action  $a$  according to the  $\epsilon - greedy$  policy

Execute action  $a$  to get new state  $s'$  and reward  $r$

$$Q(s, a) = (1 - \lambda)Q(s, a) + \lambda(r + \gamma * \max_{a' \in A} Q(s', a'))$$

$s = s'$

*current iteration* = *current iteration* + 1

$\lambda$  is the learning rate of the agent, which specifies how much the value of  $Q(s, a)$  is altered from its previous evaluation by a new discovery. The  $\epsilon - greedy$  policy determines that the agent should mostly choose random actions initially, while shifting towards choosing the actions which maximize  $Q(s, a)$  as more iterations are completed. The idiom that Q-learning is an off-policy algorithm is due to the agent not necessarily executing actions according to the best policy it has found, given by  $\max_{a \in A} Q(s, a)$  in a certain state  $s$ .

The formula of Q-learning proposed by Watkins has afterwards been altered so that it can be used in larger state-spaces as well as converge faster. In [12], the Q-learning algorithm was developed further by introducing a neural network to assess the values of state-action pairs. This allowed Q-learning to be employed in vast and continuous state-spaces, as previously the algorithm was limited by computer memory due to the large number of stored  $Q(s, a)$  values. Furthermore, this allowed the agent to propagate the information of previous actions to actions in similar states, which reduced the time to convergence.

In [13], it was shown that the convergence time of the algorithm could be decreased by carrying a probability distribution along the expected Q-values. The probability

distribution was constructed as an estimate of the likelihood that the analyzed  $Q(s, a)$  value was correct. In this way, the agent could factor in its uncertainty about the correctness of an action, even though the face-value of that action was the highest.

### C. STATE-OF-THE-ART

As explained at the end of the last section, the original version of Q-learning was not well suited for large state-spaces. This problem has been addressed in for example [14], where the convergence of a Q-learning algorithm was evaluated with various linear function approximators.

More recently, it was demonstrated that deep neural networks can be effectively utilized as function approximators by employing memory replay functionality in the Q-learning algorithm [15]. The authors coined the algorithm “Deep Q Network” or DQN. The Deterministic Policy Gradient (DPG) algorithm was developed in [16] on the basis of previous stochastic policy gradient algorithms [17], to render the estimation of action values more efficient in continuous action spaces.

In [18], the authors combined DQN and DPG to further increase the performance deep RL. The resulting algorithm was applicable to continuous state and action spaces with stable value function approximation. The authors used various physics tasks to benchmark their algorithm, and compared it to different planning controllers with full access to the physical model of the task. The results were surprisingly good with the RL algorithm outperforming the model-based controllers on various tasks only using pixel images of the physics tasks as state representation.

Perhaps the most well-known modern implementation of RL is the artificial intelligence that mastered the Chinese game of Go [19]. This implementation used deep neural networks to evaluate state-action values and a pipe-lined learning phase in which the neural networks were first trained by supervised learning, and then reinforcement learning. The supervised learning portion used data gathered from moves that Go experts had made, and the reinforcement learning portion learned on its own by playing against itself. In addition to this, a novel Monte Carlo tree search algorithm was implemented to more accurately evaluate the values of state-action pairs.

In [20], a DQN was implemented to control an energy storage in a micro-grid with photovoltaic panels. The authors claimed that the control algorithm was able to capture the stochastic nature of power consumption and solar irradiance, and they were able to increase the operational revenue of the system on a test-bed.

The application of Q-learning for the control of residential energy management systems was proposed in [21]. The authors proposed a system which would distribute power to consumers, such as dishwashers and ovens, at specific time-slots with the intention of minimizing total energy cost and power peaks. The control logic behind this system was based on Q-learning, which captured the stochasticity of energy

pricing and user behaviour. They concluded that the system was able to reduce the average cost of energy for the user, as well as smoothen the overall energy usage.

In [22], the application of RL was studied for the power management of an HDD storage device and a WLAN card in a computer. The application was based on TD( $\lambda$ ) learning [23], which is a method applicable to continuous state-spaces while also sharing some similarities with Q-learning. The authors demonstrated that their power management system reduced energy consumption by 16.7 % compared to the previous rule-based control. A similar study on the application of RL for the power management of computer systems was conducted in [24].

An optimization based approach was proposed in [25] for the optimal management of power distribution in an all-electric ship power topology with an energy storage. The target of the optimization was to minimize fuel consumption while respecting the technical boundaries of the ship and limiting the total greenhouse gas emissions. The control logic was established in a cascading three-phase dynamic programming model. First, the usage of the energy storage was calculated, then, the optimal dispatch of the generators, and finally, a control logic was chosen that fulfilled a constraint of minimum distance to travel.

While the proposed method was elaborately created, it still relied on knowing the power demand profile beforehand. In [26], the same author improved the devised methodology further by including the control of propulsion power in the model as a decision variable. The main benefit of such an approach was that an energy storage was no longer needed for optimizing the dispatch of the energy system. Similar optimization models have been devised in [27] and [28].

In [29], an optimization approach was employed to control the energy system of an all-electric tugboat with an energy storage. The authors recognized the need for predicting future power demand thus proposing a novel load prediction algorithm to work in conjunction with the optimization approach. The combined usage of these two methodologies worked to some extent, and the authors were able to achieve roughly 9% fuel savings in simulated runs, compared to a rule-based controller. Nonetheless, the authors acknowledged the challenges associated with the approach due to the stochasticity of the power demand. Load prediction algorithms have also been proposed in [30] for ships and for land-based electricity grids in [31].

The problem formulation in this work is closest to the work carried out in [32]. In the doctoral dissertation, the unit commitment problem of switching generators on and off aboard a vessel is studied, and a methodology is proposed based on the optimization of load dependent start tables that signify when the process of starting and stopping generating sets should occur. A similarity can be identified between the load tables and state-action pairs in Q-learning.

To the best of the authors’ knowledge, no literature exists that applies RL methods to the control of the auxiliary power network of a ship. The literature examined in the

TABLE 1. Ship main parameters.

Parameter	Value
Length	203 m
Breadth	31.93 m
Year built	1990
Deadweight	3779 t
Gross tonnage	58376
Ice class	1A Super
Main engines	4 x 8145 kW
Auxiliary engines	2 x 2400 kW, 2 x 3200 kW
Top speed	23 knots

previous paragraphs is related to electric power management in general. The passenger ferry auxiliary power network has certain unique properties, such as considerably high power peaks and the need to retain a variable power reserve. These features distinguish the presented ferry case from the existing literature on RL applications.

Furthermore, the current literature on on-board power management is focused on optimization approaches in which the power demand profile is either assumed to be known beforehand, or then a separate load prediction algorithm is employed alongside the optimization. Additionally, these approaches are often exceedingly computationally heavy, depending on the complexity of the optimization model.

RL methods fit the problem of power management very well, as both load prediction and system control are inherently included in the learned policy. Furthermore, the actuation of the control logic is drastically computationally lighter compared to optimization methods. This is because the resource intensive training phase is completed beforehand.

Due to these reasons, it is worthwhile exploring the possibility of applying RL methods for the control of the auxiliary power network of a ship. This paper aims to take the first step towards this goal by introducing the way to take into account concepts which are unique to the problem in a RL framework. This work makes the following original contributions:

- the formulation of a ship's auxiliary network as a MDP,
- the application of Q-learning for near-optimal, autonomous control of the power network,
- the demonstration of how improvements in control logic affect fuel efficiency and,
- open access to the used Python scripts.

### III. METHODS

#### A. CASE-STUDY SHIP

The case-study ship is a conventional, direct-driven passenger ferry. It operates on the Baltic Sea between Helsinki and Stockholm, briefly stopping at Mariehamn along the way. The main parameters of the ship are shown in Table 1. This study is based on a comprehensive data set extracted from the automation system of the ferry. The data set contains 9 round-trip measurements of the ferry's auxiliary power network operation with a measurement interval of five minutes.

The auxiliary power is produced by four generating sets, of which two have a power rating of 2400 kW and the other two a power rating of 3200 kW. The auxiliary power demand

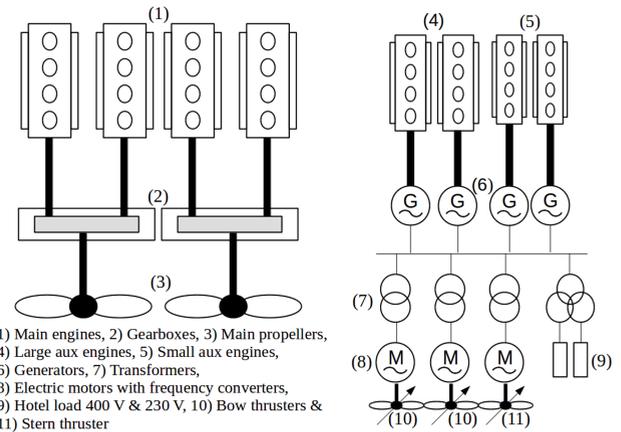


FIGURE 1. The ferry energy system topologies: main propulsion (left) and auxiliary power network (right).

can be roughly divided into power consumed by the bow and stern thrusters, and ferry hotel load. The hotel load is composed of consumers such as the HVAC system, lighting and other appliances.

Power for the main propulsion is delivered by four diesel engines with a power rating of 8145 kW. These engines drive the ships two main propellers, and they are completely separated from the auxiliary network. As such, the main propulsion side of the energy system will not be considered as a part of this study. A detailed description of the whole ship's energy system topology can be seen in Fig. 1. Note that the actual cylinder count of the engines does not correspond with the cylinders shown in the picture.

Fig. 2 shows a typical auxiliary power demand profile for the ferry during one 48-hour round-trip. The auxiliary power demand is characterized by a fluctuating base power demand around 2000 kW corresponding to the hotel load, and by infrequent peaks of power demand. The six sharp peaks in the power demand profile occur when the ship is maneuvering with thrusters while entering or leaving the harbor.

The shown auxiliary power profile corresponds to one round-trip cruise from Helsinki to Stockholm, and then back to Helsinki. The six peaks signify the following maneuvering events of the ferry in the order in which they occur: 1) leaving Helsinki harbor, 2) entering and leaving the harbor of Mariehamn, 3) entering the harbor of Stockholm, 4) leaving the harbor of Stockholm, 5) entering and leaving the harbor of Mariehamn and finally, 6) entering the harbor of Helsinki. The maneuvering events in Mariehamn appear as a single peak in the data, because the mooring time is only 10 minutes.

Figure 2 also shows the way the generating sets of the ship are operated. Generating sets 1 and 2 are the smaller, 2400 kW, variants and 3 and 4 the larger, 3200 kW engines. A typical trend in the operation of the generating sets is that one generating set is used to produce power when staying in port, and at least two are used when operating in the Swedish archipelago between the port of Mariehamn and Stockholm. During open sea operation, the choice between

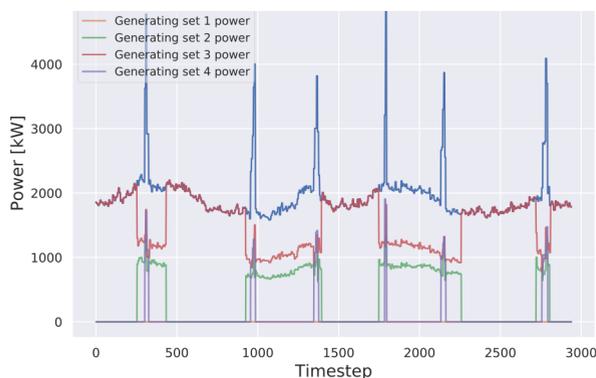


FIGURE 2. The ferry auxiliary power demand on a round-trip cruise.

using one or two generating sets seems to depend on the actual power demand. Typically when demand exceeds 2000 kW on the open sea, another generating set is started. This behavior is also consistent with the other round-trip data sets.

The reason for using more than one generating set to produce power when one set would suffice is to prevent blackouts by increasing the power redundancy. It is evident that a blackout occurs if the only online generating set suddenly fails; therefore, as a safety measure, at least two sets are operational when redundancy is needed, for example, for maneuvering. Ship classifications include passages that determine auxiliary thrusters as Type 1 Redundancy, which means that the allowed time-lag for re-establishing functionality cannot exceed 30 seconds in case the case of a single system failing [33]. This imposes a redundancy requirement on the generating sets as well, because auxiliary thruster power demand forms a significant proportion of the overall demand when they are used. This is also reflected by the fact that generating set manufacturers provide recommendations for minimum amounts of power retainment when using multiple generating sets [34].

However, running several generating sets to increase redundancy leads to a lower fuel efficiency. Maritime regulations state situations for which redundant power generation is required and ship automation systems include hard-coded rules to switch on and off generating sets, but on-board crew can also manage the number of active generating sets. According to the presented data, the rule of operation is not as simple as just having at least two generating sets running constantly.

In open sea operation, the operation policy seems to balance between robustness against blackouts and saving fuel by using just one generating set. There is a minimum amount of reserve power that the auxiliary network tries to maintain by switching on generating sets in case the amount of current reserve power falls below a given threshold value. The threshold varies depending on the position on the route.

The control logic of the auxiliary power network is quite complex due to these characteristics. To the best of the authors' knowledge, this control is typically established with a rule based logic in modern ships, which can become

decidedly complicated as the logic attempts to consider all relevant factors. With RL, the power system can be described using a MDP. The complex control logic of the power system is contained in the learned policy.

## B. DATA PROCESSING

A total of nine 48 hour round-trips in the data set were utilized in this study. These data contained the following values indexed by timestamps:

- auxiliary power demand,
- actual operation of generating sets,
- fuel consumption of generating sets,
- ship position as coordinates.

The complete data set was first divided into 9 round-trips, each starting and ending at the port of Helsinki. To establish a comprehensive state representation of the auxiliary power network of the ship, it was determined that the operational mode and destination of the ship must be included in the model as state variables.

Four distinct operational modes were distinguished in this case which were described with an integer number:

- 1 for maneuvering with auxiliary thrusters,
- 2 for open sea operation
- 3 for operation in the Swedish archipelago and
- 4 for staying in port.

And similarly for the current destination of the ship:

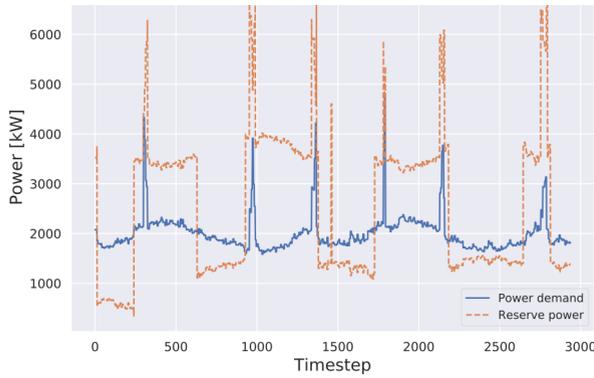
- 0 for staying in port,
- 1 for Mariehamn,
- 2 for Stockholm and
- 3 for Helsinki.

These values were added to the data sets by first analyzing the derivative of a moving mean over the data sets to identify the power peaks. With the peaks identified, it was possible to assign each timestep new data-values corresponding to the operational mode and destination of the ship.

The auxiliary power network should maintain a suitable amount of reserve power in order to maintain the ability to turn on power consumers at short notice, which may be necessary in scenarios that require actions, such as sudden maneuvering. The reserve power needs to be quantified as a numerical value in the data in order for the Q-learning agent to adopt this robust operation as well. The reserve power was calculated by subtracting the power demand from the maximum possible power output of the online generating sets. A numerical reserve power threshold for open sea operation can be deciphered from Fig. 3.

Looking at the operation around the 700th timestep, we can see that a generating set is switched off when the retained reserve power with one generating set reaches 1200 kW. This threshold is used as the base-line value of reserve power in the proposed methodology in the next sections.

This data set forms a good platform on which the feasibility of the proposed method can be verified. Note that actual implementations should use a more comprehensive data set, preferably with data of the vessels year-round operation.



**FIGURE 3.** Ship auxiliary power demand and reserve power retention on a round-trip cruise.

This is to ensure the entire possible range of operation is captured in the data, and learned by the Q-learning agent.

### C. AUXILIARY NETWORK AS A MDP

The performance of Q-learning depends significantly on the way the auxiliary power network is established as a MDP. This section follows the same nomenclature as in II-A. The possible states of the system are described as vectors of the form

$S = (G_1, G_2, G_3, G_4, M_o, D_g, G_s)$ , where

- $G_1 = \{0, 1\}$ ,
- $G_2 = \{0, 1\}$ ,
- $G_3 = \{0, 1\}$ ,
- $G_4 = \{0, 1\}$ ,
- $M_o = \{1, 2, 3, 4\}$ ,
- $D_g = \{0, 5, 10, \dots, 400\}$ ,
- $D = \{0, 1, 2, 3\}$ .

$G_{1..4}$  is the state of a single generating set, where 0 means that the generating set is off-line and 1 that it is online.  $M_o$  is the operational mode of the ship,  $D_g$  the distance to the current destination in kilometers and  $D$  the current destination. The possible actions  $A$  represent commands to switch the corresponding generating sets on or off with the addition of a possible action to do nothing.

When an action is taken a timestep variable is incremented by one. This timestep variable is used to update state-variables  $M_o, D_g$  and  $D$ , as well as to calculate the immediate reward received from the current state.

Some partial observability manifests itself through the stochasticity of the immediate reward. Nevertheless, the prevalence of this stochasticity does not necessitate the consideration of belief states when assessing the value of state-action pairs; thus, the MDP is treated as if it was fully observable. It could be argued that an underlying MDP exists for this problem formulation perfectly capturing the upcoming power demand changes in its state-variables. Such state-representation would need to include an immense amount of variables, rendering it unfeasible for real-world applications. In this work, the variables were chosen to ensure that the state-representation is sufficiently extensive, while still choosing

values that are easily available for the actual ferry. It is the task of the Q-learning agent to formulate a policy which takes into account the stochasticity of the power demand.

The immediate reward  $R_t$  is calculated from the specific state that the agent inhabits according to:

$$r = c_{max} - c_{current} + p_{on} + p_{off} + p_{reserve} + p_{demand}, \quad (2)$$

where  $c_{max}$  is the maximum possible consumption in the current state and  $c_{current}$  is the actual consumption.  $p_{on}$  and  $p_{off}$  are penalties associated with starting and shutting down a generator, respectively, and  $p_{reserve}$  and  $p_{demand}$  are the penalties for not fulfilling the reserve power and power demands.

$c_{max}$  in Eq. 2 is defined as the consumption in case the current power demand was fulfilled by having all of the four power generators online. The consumption of a single generating set is calculated in  $g/s$  from the SFOC-curve of the diesel engine of the set, and defined as:

$$c(L) = SFOC(L) * P_{g\_max} * L * 3600s, \quad (3)$$

where  $c$  is the consumption of the engine in  $g/s$  as a function of the engines load  $L$ .  $SFOC$  is the SFOC-curve of the engine, which is the consumption ( $g/kWh$ ) as a function of the engine load and  $P_{g\_max}$  is the power rating of the engine.

The SFOC values were attained from the engine manufacturers product guide [35]. The product guide offered values for operating points at 50%, 75%, 85% and 100% engine load. These points were used to interpolate the complete SFOC curve with quadratic polynomial interpolation.

$c_{current}$  in Eq. 2 is the combined consumption of the generators currently online. The engine controllers of the auxiliary power network employ load balancing methods to ensure that the generating sets produce electricity at a stable frequency in parallel operation. This means that all online generating sets are maintained at the same load percentage to ensure that power transients equally affect their speed. Thus, the load percentage can be calculated in each state from the current power demand according to:

$$L = P_d / P_{max}, \quad (4)$$

where  $P_d$  is the current power demand and  $P_{max}$  is the sum of the online generating set power ratings.

In Eq.2,  $p_{on}$  is set as the fuel oil consumption of an idle engine for 180 seconds, which corresponds to the generating set ramp-up time before it can be connected to the auxiliary power network [35]. Similarly,  $p_{off}$  is equal to the idle consumption for 300 seconds. The reasoning behind this value is that according to the examined data sets, the engines are run on idle for five minutes before they are completely shut down.

$p_{reserve}$  in Eq. 2 was calculated as  $R_a - R_d$  multiplied by a reserve violation penalty weight, where  $R_a = P_{max} - P_d$  and  $R_d$  is the reserve demand. While the other penalty and reward terms are determined by the fuel oil consumption of the auxiliary power network,  $p_{reserve}$  is more of an abstract

penalty associated with the robustness of the operation. This is the reason for a weight term being associated with the penalty, which could be adjusted like a model hyperparameter. Finally,  $p_{demand}$  was declared to be an arbitrarily large quantity, in this case, 90 million, to clearly signal the Q-learning agent that power demand violation is not tolerated under any circumstances.

#### D. Q-LEARNING

As described in section III-B, data from 9 trips were used to train and test the Q-learning model. The 9 trips were divided into 7 training sets and 2 testing sets. These trips included a single trip in which the ship did not stop at Mariehamn when traveling from Stockholm due to rough weather. The data set of this trip was restricted to appear only in the 7 training sets, as the algorithm would most likely not be able to operate competently on a testing set that does not resemble any of the sets used for training. The 7 training sets were concatenated together to form a cohesive set of data, spanning over a total time-frame of 336 hours.

The Q-learning algorithm was executed with a future reward discount factor of 0.7, and a learning rate of 0.055. The weight factor on violating the reserve power requirement was set at 75. The selection process of these hyperparameter values will be explored in the next section.

The agent begins exploring the state-space by taking random actions with a probability of  $\epsilon$ , and actions that maximize  $Q(s, a)$  otherwise.  $\epsilon$  is calculated with a decaying  $\epsilon$ -greedy policy:

$$\epsilon = e^{(i/I)}, \quad (5)$$

where  $i$  is the current iteration and  $I$  the total amount of iterations to perform. Equation 5 causes exploration of the state-space to be initially preferred heavily over exploitation of previous knowledge. The probability of random actions diminishes over iterations, which allows the agent to eventually start leveraging its previous knowledge, and focus on exploring the sequence of actions it considers optimal. The auxiliary power network state-space is traversed so that the action given by the agent is executed, which naturally changes the  $G_{1...4}$  terms of the current state. A timestep variable is also incremented by one, and variables  $M_o$ ,  $D_g$  and  $G_S$  of the state are updated according to the new timestep. This is continued until the last timestep in the data is encountered, in which case, the MDP is reset to the initial state and the time-step is set to 1 again.

The state-space was formatted to include 20480 possible states. The possible actions in states in which a single generating set was online were limited in such a way that the agent could not shut down the last operating generating set. Subsequently, 97280 possible state-action pairs remained in  $Q$ . This number of the state-action pairs is quite manageable for modern computers, which is the reason why no function or neural network approximator was required for the evaluation of state-action values in this case. An illustration

TABLE 2. Hyperparameter test ranges and increments.

Hyperparameter	Range	Increment
Discount factor ( $\gamma$ )	0.6 ... 0.9	0.1
Learning rate ( $\lambda$ )	0.005 ... 0.1	0.005
Reserve requirement violation weight	50 ... 150	25

of how the Q-learning algorithm advances and how the MDP is utilized can be seen in Fig. 4.

#### E. HYPERPARAMETER TUNING

Ideally, a developed RL model should be robust against small changes in its hyperparameters to avoid unnecessarily meticulous hyperparameter tuning. The changeable hyperparameters in the Q-learning model were the learning rate  $\lambda$  and the discount factor  $\gamma$  as per usual. In addition to these, the weight factor on reserve power demand violation penalty was also considered a tuning parameter. This is not a hyperparameter of the model in itself, but it has significant impact on the value estimation of states.

To conduct a comprehensive analysis, suitable discrete ranges were selected for all of the parameters mentioned above, and then all possible combinations of those parameter selections were created. The changed parameters, their ranges and increments are shown in Table 2.

The model was then executed for 10 million iterations for each combination of the hyperparameters. Afterwards, hyperparameter combinations were discarded, which resulted in a policy which violated the power demand requirement after 5 million iterations. Finally, the results of the policies with the remaining hyperparameter combinations were manually evaluated based on their convergence time and fuel savings. Based on the analysis, the best hyperparameters in terms of convergence time and stability were  $\lambda = 0.055$ ,  $\gamma = 0.7$  and a reserve power violation weight of 75.

## IV. RESULTS

Fig. 5 illustrates the learned control logic for trip number 8, which was part of the testing set. The operation of the generating sets resembles that of the measured real-world operation closely, in which the auxiliary power generating sets were controlled manually. This is due to the boundary condition of reserve power being derived from the real-world operation data. One of the major differences between the manual control and the one established in this work, is that the learning agent prefers to start just one additional generating set for maneuvering events whereas three generating sets are used in the manual control of the ship.

Notably, the agent has learned to operate in the harbors using only one generating set and two generating sets in the critical area of the Swedish archipelago. The agent also prefers to use one of the smaller generating sets while in harbor, contrary to most of the harbor operations in the measurements. In addition, the agent has also learned that hotel load tends to increase right after leaving the port of

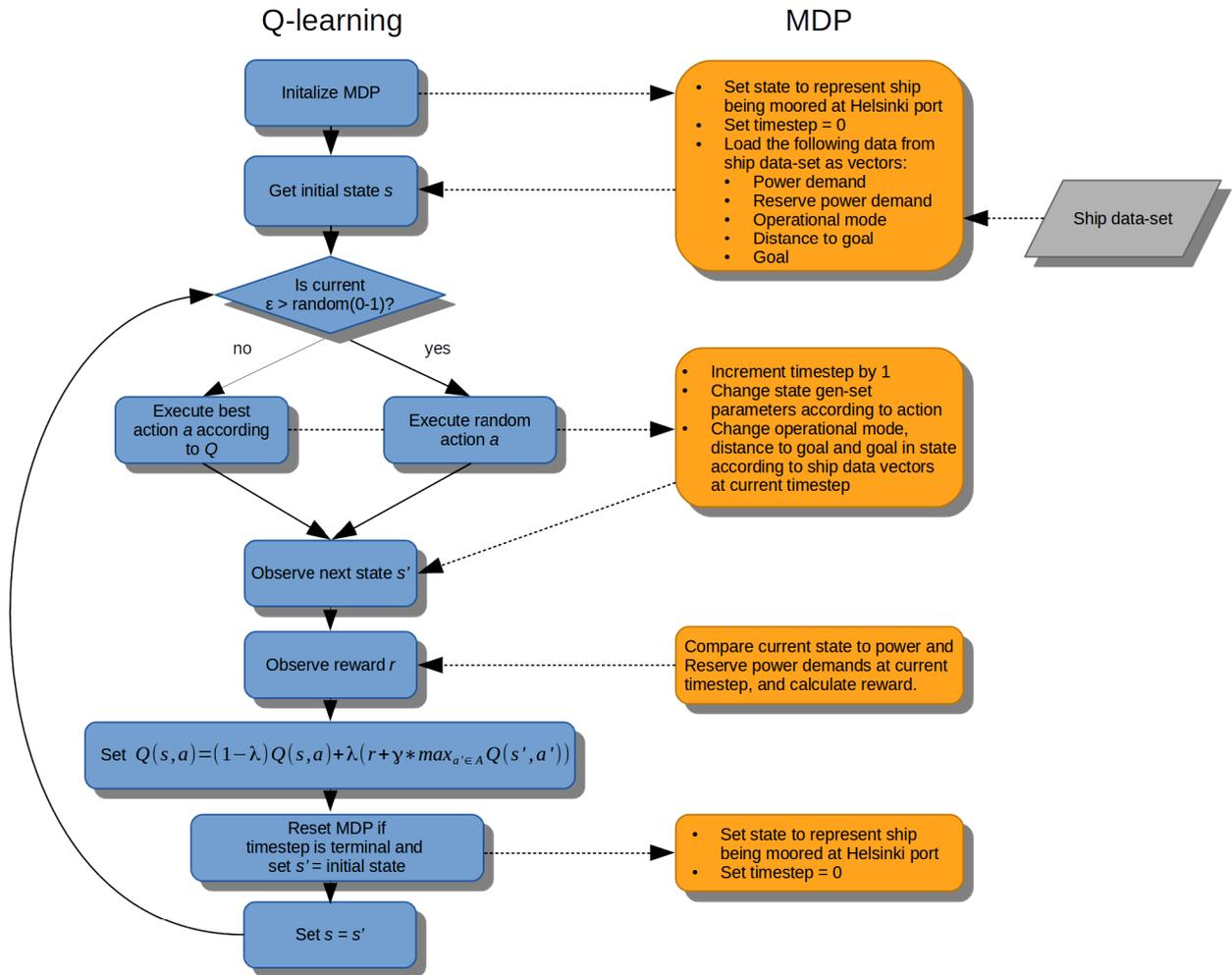


FIGURE 4. Flowchart describing how the Q-learning algorithm advances, and how information is exchanged between it and the MDP.

Helsinki, compared to the value of hotel load when staying in the port of Helsinki. Therefore, an additional generating set is employed despite the operational mode being open sea operation; an operational mode in which one generating set is usually preferred. Conversely, when arriving in Helsinki, the agent prefers to use only one generating set.

The saved fuel compared to the actual operation was 211 kg, corresponding to a consumption decrease of 1.09%. The control logic shown in Fig. 5 was learned by the model after 80 million iterations. The learning period took about 6 hours on a standard desktop computer equipped with an Intel Core i7-9700k processor running at 3.7 GHz and with 16 Gb of RAM.

The fuel oil savings differ by a small amount depending on which trip is selected for analysis. The savings for each trip are shown in Table 3.

The variance in fuel oil savings can be explained by the human factor effect in actual operations. Some of the actual operational profiles, namely 2 and 3, exhibited fuel oil consuming control sequences, such as switching from generating set 3 to 4 during open sea operation, and starting additional

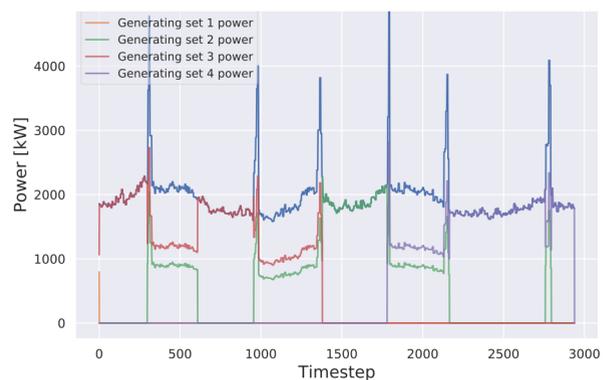


FIGURE 5. Auxiliary network control with Q-learning.

generating sets well in advance for maneuvering events. These trips were also the ones in which the Q-learning agent managed to save most fuel compared to the actual operation. Conversely, the operational profile for trip number 8, which is depicted in Fig. 2 and Fig. 5, contains no such sequences in the actual operational profile. Subsequently, the fuel oil

TABLE 3. Fuel oil savings in each of the analyzed trips.

Trip number	Fuel saved [tons]	Fuel saved [%]
1	214	1.1
2	266	1.4
3	229	1.2
4	151	0.8
5	89	0.5
6	194	1.0
7	160	0.8
8	158	0.8
9	120	0.6
Average	176	0.9

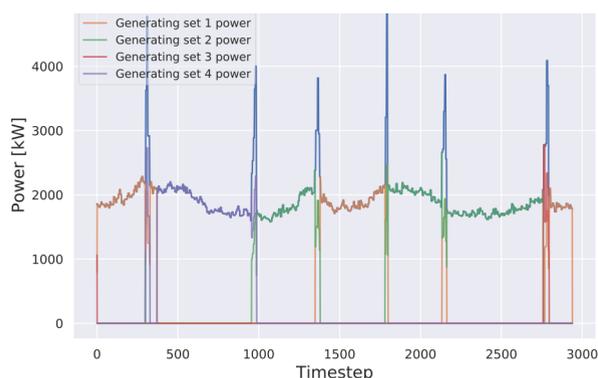


FIGURE 6. Auxiliary network control with Q-learning ignoring reserve power requirement.

savings are less, being formed primarily due to using two generating sets for maneuvering operations and one of the smaller generating sets for some of the port operations.

An interesting case emerges when we set the weight factor of reserve power violations to 0. In this case, the Q-learning agent does not receive a negative reward signal even if it violates the established reserve power requirement. Such a modification causes the agent to seek an optimal policy to minimize the fuel oil consumption with no consideration of reserve power.

Figure 6 shows the way such a control logic operates the generating sets. As expected, the agent prefers to use only one of the generating sets for most of the operations, excluding maneuvering events in which an additional generating set is started. The fuel saved in such a scenario was 861 kg compared to the manually controlled real-world operation, a consumption decrease of 4.45%. This greedy control strategy provides insight into the relationship between the optimal energy saving operation and flexible operation which retains reserve power for unexpected events.

Fig. 7 provides an insight into the actual learning process. The cumulative reward signal was recorded every 25000th iteration, by extracting the learned Q-values at that iteration, and then using those as a policy for operating through the testing data sets. The reward signals of each time step were then summed together to form the cumulative reward signal. The cumulative reward signal is characterized by a seemingly fast

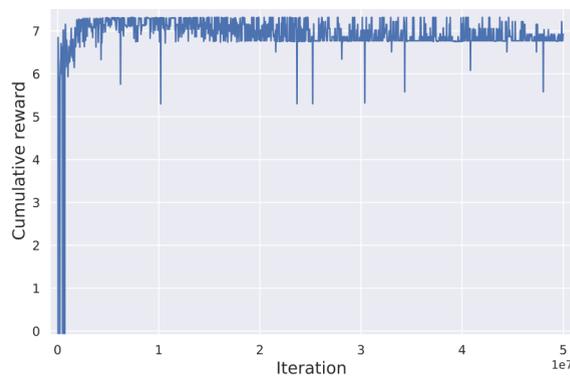


FIGURE 7. Cumulative reward signal development while learning.

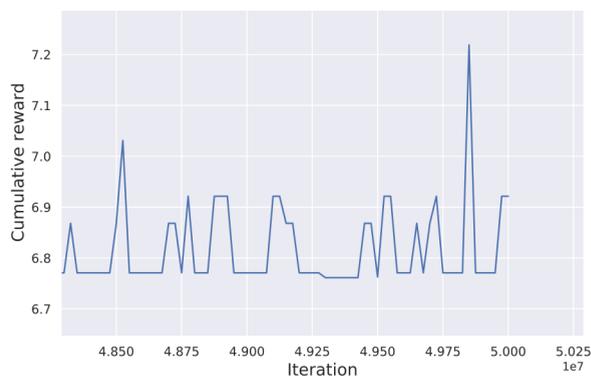


FIGURE 8. Close-up of discrete fluctuations in the cumulative reward.

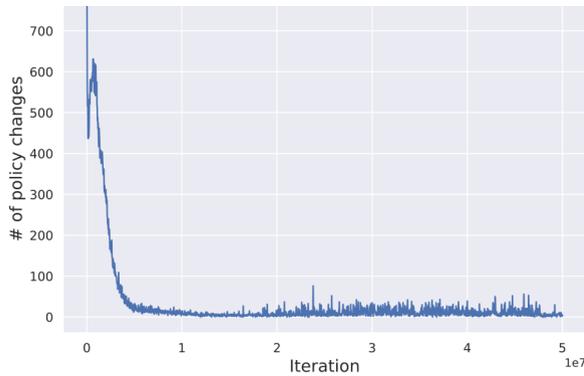
climb to around 70 million. Afterwards, the reward fluctuates in a discrete manner around the value of 70 million.

A close-up of these fluctuations is shown in Fig. 8. The close-up reveals that the fluctuations appear to cycle through a few distinct cumulative reward values. A closer analysis revealed that these fluctuations correspond with the agent deciding whether to use a large or a small generating set for port operations. In this case, trip number 9, followed by 8 were used as the testing data sets.

Fig. 9 depicts the amount of policy changes made during the measurement period of 25000 iterations. The amount of policy changes saturates in approximately 15 million iterations to just a few changes in the measurement period. Afterwards, the amount of policy changes starts growing larger. However, the majority of these latter policy changes have no effect on the value of the policy, because they are composed of changes between identical actions, such as choosing between starting two identical generating sets.

## V. DISCUSSION

Based on the simulation results on real measured input data, moderate fuel savings can be attained by adjusting the control methodology of the auxiliary power network, in the case of the restricting boundary conditions, such as reserve power retainment, being kept the same as in manual operation. The average 0.9% average savings in fuel oil result in approximately 17000 \$ savings annually, according to the



**FIGURE 9.** Amount of policy changes made during the learning process.

3-year average price of LSMGO in the Rotterdam harbor [36]. Further increases in energy efficiency depend on either relaxing the reserve power requirement, or introducing new features into the power system such as an energy storage.

A comparison can be made between the proposed methodology and the methodology presented in [32], in which the on/off switching of generating sets was governed by optimized load dependent start tables. The load dependent start tables were optimized according to the probability distribution of possible operational modes of the ship. This leads to the load dependent start tables usually being correct when choosing to switch a generating set on or off, but situations may occur when the operational mode changes quickly leading to generating sets being unnecessarily switched on or off. The proposed methodology in the present work does not suffer from the same problem, because the agent learns the complete operating cycle rather than the probability distribution of possible operating modes. The cost-functions of the load dependent start tables were also non-convex in some situations, which leads to uncertainty in the optimality of the solution. The author also hints in [37] that the load dependent start tables could also be formulated for power networks that contain an energy storage. However, this claim is left unverified, as the author does not discuss the means by which such an implementation could be achieved.

A proportion of the declared fuel oil savings are a result of the Q-learning agent using only two generating sets for maneuvering events, whereas three were used in the measured manually controlled operation. There might be an underlying reason as to the reason that three might be preferred, but based on the analyzed data, two sets satisfy the power demand for all the presented maneuvering events. In the case of three sets being needed for safety reasons, the fuel oil consumption of the auxiliary power network with control by Q-learning would more closely resemble that of the actual control.

Furthermore, this analysis was based on data collected during the winter. It is expected that the hotel load depends

on the time of year, because the power demand of the HVAC system depends on the interior temperature of the ship. In the case of the changes in hotel load magnitude becoming considerable, the state-representation should include the time of year as a state-variable. This is a topic for further research.

In theory, the model is capable of formulating an optimal control policy for the auxiliary network without the explicitly declared reserve power amounts. Learning such a policy would depend on the presence of a data set that perfectly captures the possible state-space of the ship, including blackouts which are rarely experienced on board, as well as a perfectly evaluated penalty for undergoing a blackout. In such a case, the Q-learning agent would eventually experience an event that extremely rarely leads to the blackout, and adjust its value estimations as if a reserve power demand was explicitly declared, as it was in the present study. This idea was explored in the results which analyzed the control policy in which the reserve power violations were ignored. Such an analysis leads to a model which captures the stochasticity present in the data set that it was trained on, which can be suitable in the case of the training data set being sufficiently extensive. Thus, the authors suspect that with more real-world operation data, the hyperparameter tuning is less sensitive and the number of rules and limitations of the control can be reduced.

In this study, the parameters of the engines in the auxiliary network were derived from the documents of the manufacturer. This naturally leads to two equally sized engines being identical from the perspective of the agent forming the control policy. If the presented methodology was applied on a real ship, the automation system of the ship could be furnished with a method for evaluating the individual specific fuel consumption curves of engines, which can change based on the condition of the engine. As a consequence, the policy learning agent would learn to prefer engines with smaller fuel consumption, leading to smaller overall consumption. The policy would be fairly different from the current practice in manual operation, in which engines are run in turns in order to accumulate operation hours evenly.

The benefits of intelligent autonomous control of ship auxiliary power networks become more evident when an energy storage is included in the network topology. In such systems, the power production of the generating sets can be decoupled from the power demand in time with the energy storage. This renders the control of these hybrid networks inherently more complex. On the other hand, the energy storage operates as a passive source for reserve power, thus increasing the overall fuel efficiency of the ship.

The inclusion of an energy storage into the MDP representation of the network could be achieved by discretizing the input and output power of the energy storage, and including the power levels in the MDP as possible actions. If this results in an exceedingly large state-action space, the Q-Learning algorithm can be modified to employ either a function approximator or a neural network for the Q-values. In this way, the input and output powers of the energy storage can be treated as continuous variables.

## VI. CONCLUSION

This paper presents a RL method for automating the operation of the auxiliary power network of a ship. Real operational data was gathered from the automation system of a passenger vessel and used to train the RL model which was based on Q-Learning. The focus of the work was on modelling the auxiliary power network of the vessel as a MDP, respecting the real-life restrictions of the network, such as the need to retain a certain amount of reserve power. The MDP was also formulated to ensure that the original version of Q-Learning could be utilized to achieve sufficient, near-optimal control of the generating sets in the auxiliary power network.

Results showed that the devised RL method was suitable for achieving autonomous control of the auxiliary power network, based on the used data. Compared to the actual operation of the network, an average of 0.9% fuel savings were attained. Consequently, the conclusion was drawn that further increases in fuel efficiency depend either on relaxing the requirement of retaining reserve power, or then introducing novel technologies into the auxiliary power network, such as an energy storage.

This study forms a basis for future work by establishing the RL model of the auxiliary power network, which can be extrapolated to more demanding control tasks, such as controlling the auxiliary power network with an energy storage. The overall goal is to advance the research of energy efficient ships towards the goals set by IMO for sustainable, autonomous, and safe maritime operation.

## ACKNOWLEDGMENT

Code used in this study is available at: <https://github.com/jannejhuotari/shipqllearning.git>.

## REFERENCES

- [1] United Nations. (2015). *Paris Agreement*. Accessed: Jun. 18, 2019. [Online]. Available: <https://unfccc.int/process-and-meetings/the-paris-agreement/the-paris-agreement>
- [2] *Adoption of the Initial IMO Strategy on Reduction of GHG Emissions from Ships and Existing IMO Activity Related to Reducing GHG Emissions in the Shipping Sector*, 2018.
- [3] E. A. Bouman, E. Lindstad, A. I. Riialand, and A. H. Strømman, "State-of-the-art technologies, measures, and potential for reducing GHG emissions from shipping—A review," *Transp. Res. D, Transp. Environ.*, vol. 52, pp. 408–421, May 2017.
- [4] Y. Liu and R. Bucknall, "Path planning algorithm for unmanned surface vehicle formations in a practical maritime environment," *Ocean Eng.*, vol. 97, pp. 126–144, Mar. 2015.
- [5] V. Ferrari, L. Perera, F. Santos, M. Hinojroza, S. Sutulo, and C. G. Soares, *Initial Experimental Tests of a Research-Oriented Self-Running Ship Model*. London, U.K.: Taylor & Francis, 2015, pp. 913–918.
- [6] A. Ritari, J. Huotari, J. Halme, and K. Tammi, "Feasibility analysis of a battery system for a passenger ferry," in *Proc. 2nd Int. Conf. Modeling Optim. Ship Energy Syst. (MOSES)*, 2019, pp. 1–9.
- [7] R. S. Sutton and A. G. Barto, *Reinforcement Learning—An Introduction*, vol. 135. Cambridge, MA, USA: MIT Press, 1998.
- [8] G. Tesauro, "Temporal difference learning and TD-Gammon," *Commun. ACM*, vol. 38, no. 3, pp. 58–68, Mar. 1995.
- [9] A. Y. Ng, A. Coates, M. Diel, V. Ganapathi, J. Schulte, B. Tse, E. Berger, and E. Liang, "Autonomous inverted helicopter flight via reinforcement learning," in *Experimental Robotics IX*. Cham, Switzerland: Springer, 2006, pp. 363–372.
- [10] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1238–1274, 2013.
- [11] C. J. C. H. Watkins, *Learning From Delayed Rewards*. 1989.
- [12] G. A. Rummery and M. Niranjan, *On-Line Q-Learning Using Connectionist Systems*, vol. 37. Cambridge, U.K.: Univ. Cambridge, Department of Engineering Cambridge, 1994.
- [13] R. Dearden, N. Friedman, and S. Russell, "Bayesian Q-learning," in *Proc. Aaai/aaai*, Jul. 1998, pp. 761–768.
- [14] F. S. Melo, S. P. Meyn, and M. I. Ribeiro, "An analysis of reinforcement learning with function approximation," in *Proc. 25th Int. Conf. Mach. Learn.*, Jul. 2008, pp. 664–671.
- [15] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with deep reinforcement learning," 2013, *arXiv:1312.5602*. [Online]. Available: <https://arxiv.org/abs/1312.5602>
- [16] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. ICML*, Jun. 2014, pp. I-387–I-395.
- [17] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2000, pp. 1057–1063.
- [18] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*. [Online]. Available: <https://arxiv.org/abs/1509.02971>
- [19] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, J. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016.
- [20] V. François-Lavet, D. Taralla, D. Ernst, and R. Fonteneau, "Deep reinforcement learning solutions for energy microgrids management," in *Proc. Eur. Workshop Reinforcement Learn. (EWRL)*, 2016, pp. 1–7.
- [21] D. O'Neill, M. Levorato, A. Goldsmith, and U. Mitra, "Residential demand response using reinforcement learning," in *Proc. 1st IEEE Int. Conf. Smart Grid Commun.*, Oct. 2010, pp. 409–414.
- [22] Y. Wang, Q. Xie, A. Ammari, and M. Pedram, "Deriving a near-optimal power management policy using model-free reinforcement learning and Bayesian classification," in *Proc. 48th ACM/EDAC/IEEE Design Automat. Conf. (DAC)*, Jun. 2011, pp. 41–46.
- [23] S. J. Bradtko and M. O. Duff, "Reinforcement learning methods for continuous-time Markov decision problems," in *Proc. 7th Int. Conf. Neural Inf. Process. Syst.*, 1995, pp. 393–400.
- [24] Y. Tan, W. Liu, and Q. Qiu, "Adaptive power management using reinforcement learning," in *IEEE/ACM Int. Conf. Comput.-Aided Design-Dig. Tech. Papers*, Nov. 2009, pp. 461–467.
- [25] F. D. Kanellos, "Optimal power management with GHG emissions limitation in all-electric ship power systems comprising energy storage systems," *IEEE Trans. Power Syst.*, vol. 29, no. 1, pp. 330–339, Jan. 2014.
- [26] F. D. Kanellos, G. J. Tsekouras, and N. D. Hatzigargyriou, "Optimal demand-side management and power generation scheduling in an all-electric ship," *IEEE Trans. Sustain. Energy*, vol. 5, no. 4, pp. 1166–1175, Oct. 2014.
- [27] C. Shang, D. Srinivasan, and T. Reindl, "Economic and environmental generation and voyage scheduling of all-electric ships," *IEEE Trans. Power Syst.*, vol. 31, no. 5, pp. 4087–4096, Sep. 2016.
- [28] F. D. Kanellos, A. Anvari-Moghaddam, and J. M. Guerrero, "A cost-effective and emission-aware power management system for ships with integrated full electric propulsion," *Electr. Power Syst. Res.*, vol. 150, pp. 63–75, Sep. 2017.
- [29] T. L. Vu, A. A. Ayu, J. S. Dhupia, L. Kennedy, and A. K. Adnanes, "Power management for electric tugboats through operating load estimation," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 6, pp. 2375–2382, Nov. 2015.
- [30] B. P. Pedersen and J. Larsen, "Prediction of full-scale propulsion power using artificial neural networks," in *Proc. 8th Int. Conf. Comput. Appl. Maritime Ind. (COMPIT)*, Budapest, Hungary, May 2009, pp. 10–12.
- [31] S. Fan and R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model," *IEEE Trans. Power Syst.*, vol. 27, no. 1, pp. 134–141, Feb. 2012.
- [32] D. Radan, "Integrated control of marine electrical power systems," Tech. Rep., 2008.
- [33] DNV-GL, "Rules for classification, ships, part 4 systems and components, chapter 1 machinery systems, general," DNV-GL, Høvik, Norway., Tech. Rep., 2018.

- [34] *MAN 51/60DF, Project Guide—Marine*, 2015.
- [35] *Wärtsilä 32 Product Guide*, Wärtsilä, Helsinki, Finland, 2018.
- [36] Ship & Bunker. (2019). *LSMGO Fuel Price*. Accessed: Jun. 30, 2019. [Online]. Available: <https://shipandbunker.com/prices/emea/nwe/nl-rtm-rotterdam#LSMGO>
- [37] D. Radan, A. J. Sorensen, T. A. Johansen, and A. K. Adnanes, "Probability based generator commitment optimization in ship power system design," *WSEAS Trans. Syst.*, vol. 5, no. 8, pp. 1901–1906, Jan. 2006.



**JANNE HUOTARI** received the B.Sc. and M.Sc. degrees from Aalto University, Espoo, Finland, in 2015 and 2018, respectively, where he is currently pursuing the D.Sc. degree, all in mechanical engineering.

In 2016, he was a Course Assistant with the Department of Electrical Engineering, Aalto University and a Mechanical Design Trainee with Eaton Power Quality Oy., Espoo, in 2017. During his doctoral studies, he has been a course assistant on courses related to mechatronics. He received the METEX-Award for his master's thesis on the positioning tolerances of sensor elements in the CMS particle detector at CERN.



**ANTTI RITARI** received the B.Sc. and M.Sc. degrees in mechanical engineering from Aalto University, Espoo, Finland, in 2015 and 2017, respectively. He has been a Researcher with the Mechatronic Machine Systems Group, Aalto University, since 2017.

His current research interests include the development of mathematical programming models for energy efficiency improvement of electric and hybrid vehicles and ships. His recent application areas include development of mixed integer linear programming models for optimal load sharing between generating sets and an energy storage systems in hybrid RoPax ships.



**RISTO OJALA** received the B.Sc. degree from Aalto University, in 2019, where he is currently pursuing the M.Sc. degree in mechanical engineering.

He is currently a Research Assistant with the Smart Mobility Group alongside his studies. He has also participated in the teaching of CAD as a Course Assistant. He has had remarkable success in his studies, being included in the Dean's list for the academic year 2017–2018.



**JARI VEPSÄLÄINEN** received the B.Sc. and M.Sc. degrees from Aalto University, in 2014 and 2016, respectively, where he is currently pursuing the Ph.D. degree. During his studies, he was also a Research Assistant with the Fluid Power Laboratory.

During his doctoral studies, he has also been involved in course planning and assisting in the teaching of mechatronics.



**KARI TAMMI** received the M.Sc., Lic.Sc., and D.Sc. degrees from the Helsinki University of Technology, in 1999, 2003, and 2007, respectively. He received a Teacher's pedagogical qualification from the Häme University of Applied Sciences, in 2017.

He was a Postdoctoral Researcher with North Carolina State University, USA, from 2007 to 2008, and a Researcher with CERN, the European Organization for Nuclear Research, from 1997 to 2000. He was a Research Professor, a Research Manager, and a Team Leader and in other positions with the VTT Technical Research Centre of Finland, from 2000 to 2015. He has been an Associate Professor with Aalto University, since 2015. He also serves with the Finnish Administrative Supreme Court as a Chief Engineer Counselor. He has authored more than 60 peer-reviewed publications cited in more than 1500 other publications. Dr. Tammi is a member of the Finnish Academy of Technology. He currently serves as the Deputy Chair of IFTOMM Finland.

...