Evaluation heuristics for tug fleet optimisation algorithms
A computational simulation study of a receding horizon genetic algorithm

Robin T. Bye and Hans Georg Schaathun
Faculty of Engineering and Natural Sciences, Aalesund University College, Postboks 1517, NO-6025 Ålesund, Norway
{roby,hasc}@hials.no

Keywords: receding horizon control, genetic algorithm, dynamic optimisation, algorithm evaluation, modelling, computational simulation.

Abstract: A fleet of tugs along the northern Norwegian coast must be dynamically positioned to minimise the risk of oil tanker drifting accidents. We have previously presented a receding horizon genetic algorithm (RHGA) for solving this tug fleet optimisation (TFO) problem. Here, we first present an overview of the TFO problem, the basics of the RHGA, and a set of potential cost functions with which the RHGA can be configured. The set of these RHGA configurations are effectively equivalent to a set of different TFO algorithms that each can be used for dynamic tug fleet positioning. In order to compare the merit of TFO algorithms that solve the TFO problem as defined here, we propose two evaluation heuristics and test them by means of a computational simulation study. Finally, we discuss our results and directions forward.

1 INTRODUCTION

Several thousand ships transit along the northern Norwegian coastline every year, with the latest figures of 2013 including 1,584 so-called “risky transports,” of which 298 were ships with oil or other petroleum-related cargo on board (Vardø VTS, 2014b). With the recent increase in traffic through the Northwest Passage and the projected increase in oil exploration in the High North (Havforskningsinstituttet, 2010), the Norwegian coastline is increasingly exposed to the risk of incidents with potentially high impact on the environment. Indeed, in 2013 alone, the Vardø Vessel Traffic Service (VTS) registered 286 operational incidents, including 186 incidents of drifting vessels, 29 of grounding, 36 of pollution, 10 of fire, and 7 shipwrecks (Vardø VTS, 2014a).

The Vardø VTS is located at the northeasternmost point of Norway and is run by the Norwegian Coastal Administration (NCA) (see Figure 1). Among other duties, the VTS constantly monitors ship movements, maintains dialogue with ships, and manages the tug fleet of Norway.

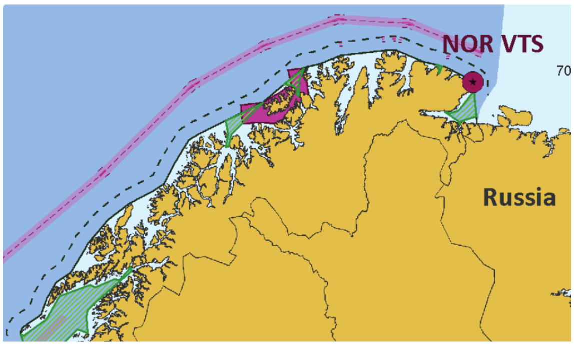

Figure 1: Northern Norwegian coastline and the Vardø VTS (shown with its call signal NOR VTS). Solid line is the geographical baseline; stapled line is the border of the Norwegian Territorial Waters (NTW); thick pink line is the traffic corridor for the Traffic Separation Scheme (TSS). Adapted from Vardø VTS (2011).

As noted above, there is an incident of a drifting vessel occurring about every second day on average. A number of these vessels are high-risk ships such as oil tankers, which if allowed to drift aground can cause serious damage to the environment due to spillage of oil and fuel. In a measure to avoid such incidents, the VTS is constantly instructing its patrolling fleet of tugs to move to new positions in a manner such that if an oil tanker loses manoeuvrability, e.g., because of engine or propulsion problems or steering failure, tugs should be sufficiently close that it can intercept the drifting oil tanker before it runs aground (Eide et al., 2007a).

A set of risk-based decision support tools based on dynamical risk models have been developed previously (Eide et al., 2007a,b). The models incorporate a number of factors such as wind, waves, currents, geography, types of ships in transit, and potential environmental impact should drift grounding occur. Whilst such tools can aid the human operators at the VTS in making informed decisions about tug fleet positioning, they do not tell explicitly where tugs should move; instead, they give the operators a real-time risk map divided into zones of low, medium, and high risk.

The number of oil tanker transits is expected to rise significantly in coming years (Havforskningsinstituttet, 2010), therefore, the problem of commanding tugs to "good" positions may become unmanageable for human operators. Motivated by this challenge, our Dynamic Resource Allocation with Maritime Application (DRAMA) research group at the Aalesund University College (AAUC) has over the last few years developed and refined a receding horizon genetic algorithm (RHGA) (Bye, 2012; Bye and Schaathun, 2014; Bye et al., 2010). The algorithm iteratively plans individual movement trajectories for the fleet of tugs such that the net collective behaviour of the tugs is optimised, that is, it employs a genetic algorithm (GA) in order to minimise cost functions that have been specifically designed to reduce the risk of drift grounding accidents. We have also investigated using mixed integer programming (MIP) for the optimisation component of the algorithm (Assimizele et al., 2013).

In our most recent work (Bye and Schaathun, 2014), we identified a flaw in the cost function we had employed previously, and suggested a number of other cost functions that could be used instead. A challenge, however, is the problem of comparing and evaluating the merit of different cost functions, or in general, of different TFO algorithms. This challenge is the focus of the work we present here.

In the following sections, we proceed by presenting a model of what we have coined as the *tug fleet optimisation* (TFO) problem, before introducing our RHGA and suggesting a set of possible cost functions that can be used in the algorithm. Next, we propose a two new and objective evaluation heuristics designed for making comparisons of TFO algorithms. Finally, we test the method on our RHGA with the set of cost functions in a simulation study and discuss the viability of our approach as well as future work.

2 METHOD

2.1 A Model of the TFO Problem

We employ a 1D model of the TFO problem and adopt most of the principles and assumptions in our earlier work (Assimizele et al., 2013; Bye, 2012; Bye and Schaathun, 2014; Bye et al., 2010).

Oil tankers are required by law to follow a predefined corridor, or lane, parallel to the coastline, depicted as the pink TSS in Figure 1. In topological space, the corridor constitutes a curve, which is locally homeomorphic to a straight line. This means that the curve can be deformed into a straight line by a continuous, invertible mapping, and vice versa. Consequently, for model simplicity, we assume that N_o oil tankers move in one dimension only along a straight line of motion z.

To the inside of the corridor, a fleet of tugs patrol the coastal waters. Ignoring a rugged coastline with islands, peninsulas and shoals, and by the same topological argument above, we may assume that N_p tugs are patrolling along a line of motion y parallel to z, e.g., the geographical baseline depicted in Figure 1. We do appreciate, however, that this approximation is only locally correct, since the curvature of y and z will make the outermost line longer than the innermost line. We also realise that is may be possible that better protection is achieved by allowing the tugs to move freely in a 2D space confined to the area between the coastline and the oil tanker corridor rather than being confined to a 1D line of motion. Indeed, our DRAMA research group is currently investigating using 2D probabilistic models and exact and heuristic optimisation techniques such as MIP, stochastic programming, and tabu search, thus extending our previous work (Assimizele et al., 2013).

We have been informed by the NCA that up until the end of 2013, three tugs have operated in the area depicted in Figure 1, and that the stretch of coastline they protect is about 1,500 km. The number of tugs have since the beginning of 2014 been reduced to two. Consequently, we give the patrol line y a length of 1,500 km, and are mainly interested in fleets of two or three tugs. For simplicity, we model y with "hard" borders to the north and to the south, outside of which the tugs will ignore drifting ships. In reality, ship traffic that may result in drift grounding outside the borders of the patrol line may still be rescued by an NCA tug, especially to the south, which is Norwegian territory.

Fundamental to our modelling approach is the existence and availability of real-time ship traffic information such as the direction and speed of the oil tankers. This information is readily available by the automatic identification system (AIS) that all ships above 300 gross tonnage are required to use on international voyages due to a regulation by the International Maritime Organization (IMO).

In addition, we require accurate simulation models that can predict the future positions of oil tankers along z and the corresponding potential drift trajectories, given real-time and predicted information about the tanker movements and the environment the tankers are travelling through. Developing such models is outside the scope of the research presented here. Instead, we are concerned with the planning and control of a fleet of tugs given that these models and resulting information is readily available.

We note, however, that due to the relatively slow dynamics of oil tankers, which cannot easily and quickly change speed or direction, obtaining reasonably accurate predicted future positions of the tankers can be done simply by using dead reckoning or linear extrapolation or by more advanced techniques such as a Kalman filter.

Drift trajectory models, on the other hand, represents a more complex problem, and depend on the ever-changing dynamics of the ocean, including currents, waves, and wind, as well the size and shape of the oil tankers. Nevertheless, although we do not integrate any such models here, they do exist and are currently an active focus of research (e.g., see Breivik and Allen (2008); Breivik et al. (2011); Hackett et al. (2006); Sørgård and Vada (1998)).

For any oil tanker moving along the line z, there is a small probability that an incident may occur at the position $z(t)$, resulting in the tanker starting to drift at $t = t_{\mathrm{d}}$. Naturally, most of the time, nothing will happen, and the tanker will continue sailing along z. Employing a discrete-time model with a sampling period of $t_{\mathrm{s}} = 1$ hour, we assume that we can estimate the future tanker positions at discrete points in time, limited to a *prediction horizon* T_{h} hours into the future. For each of the oil tankers, this results in a set of future tanker positions given by $\{\hat{z}(t|t_{\mathrm{d}})\}$ for $t = t_{\mathrm{d}}+1, t_{\mathrm{d}}+2, \ldots, t_{\mathrm{d}}+T_{\mathrm{h}}$.

Furthermore, we assume that we can determine, for example through Monte Carlo simulations, the most likely hypothetical predicted drift trajectories that emanate from each predicted tanker position $\hat{z}(t|t_{\mathrm{d}})$. Such trajectories would depend on a number of actual and forecast conditions in the area, such as ocean currents and wind speed and direction, and may or may not intersect the patrol line y after an *estimated drift duration* $\hat{\Delta}$ into the future.

According to Eide et al. (2007a), situations of "fast drift" can have drift durations as fast as 8–12 hours, whereas more typical drift durations are in the range 16–24 hours. In previous work, in order to be conservative rather than optimistic, we therefore either set the estimated drift duration $\hat{\Delta}$ to be 8 hours for all oil tankers (Assimizele et al., 2013), or to be drawn randomly for each oil tanker such that $\hat{\Delta} \in \{8, 9 \ldots, 12\}$ hours (Bye, 2012; Bye and Schaathun, 2014; Bye et al., 2010).

It should also be kept in mind that there will inevitably be a *detection delay* δ between the time when an oil tanker begins drifting at the *drift time* t_{d}[1] and the time when the VTS centre detects, or is notified of, the incident at time t_{a} some hours later, which we call the *alarm time*. The detection delay is thus given by $\delta = t_{\mathrm{a}} - t_{\mathrm{d}}$.

If we examine all the future predicted positions for all the oil tankers as well as all the corresponding drift trajectories, we obtain a distribution of *cross points* located at points where future drift trajectories will intersect the patrol line y. A cross point of the cth oil tanker's drift trajectory at time t can be defined as the position y_t^c. Assuming a drift duration $\hat{\Delta}$, a drift trajectory starting on $z(t)$ at $t = t_{\mathrm{d}}$ will have a cross point on y at $t = t_{\mathrm{d}} + \hat{\Delta}$. Assuming the same drift duration for all drift trajectories and considering the prediction horizon T_{h}, there is a predicted set of cross points for the cth oil tanker given by

$$\{y_t^c\} = \left\{ y^c_{t_{\mathrm{d}}+\hat{\Delta}}, y^c_{t_{\mathrm{d}}+1+\hat{\Delta}}, \ldots, y^c_{t_{\mathrm{d}}+T_{\mathrm{h}}} \right\}. \quad (1)$$

Moreover, we define a *patrol point* as the pth tug's position on y at time t as y_t^p.

Based on the predicted future distribution of cross points, we define the TFO problem as the problem of calculating patrol trajectories (sequences of patrol points) that start at $t = t_{\mathrm{d}}$ and have some duration T_{h}, along y for each of the patrolling tugs such that the risk of an oil tanker in drift not being reached and prevented from grounding is minimised.

Figure 2 shows a graphical summary of the TFO problem as presented above, exemplified by two patrolling tugs and three oil tankers.

2.2 The RHGA

The TFO algorithm that we study in this paper is the RHGA (Bye, 2012; Bye and Schaathun, 2014; Bye et al., 2010). The algorithm consists of two main components: receding horizon control (RHC) and a genetic algorithm (GA). The GA is a search heuristic for solving search and optimisation problems and is inspired by elements in natural evolution, such as inheritance, mutation, selection, and crossover. It has been attributed to Holland (1975), with subsequent popularisation by Goldberg (1989), and is currently a very popular optimisation tool across many different disciplines, including operations research. The GA we have implemented in our RHGA is based on work by Haupt and Haupt (2004).

The optimisation problem must be defined as a cost function such that, when evaluated for a set of candidate solutions, the GA is able to distinguish

[1]Note that t_{d} also is used as the start time for planning patrol trajectories for the tugs to follow.

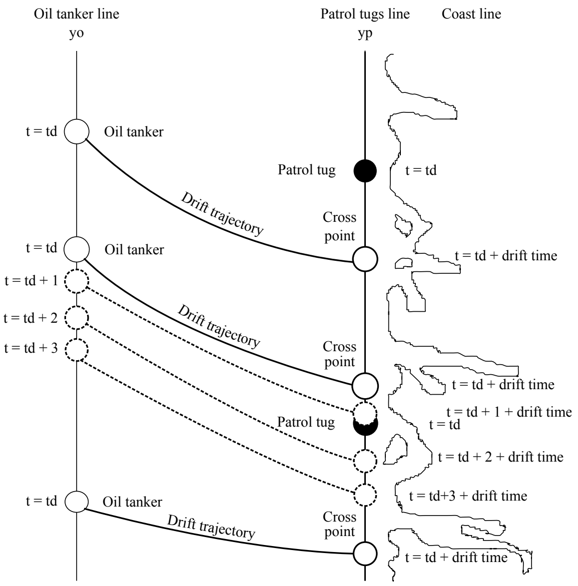

Figure 2: TFO problem: Where should the tugs move?

good solutions from bad ones. Specifically, for the TFO problem, the cost function must be designed such that its solution is a set of future position trajectories, or collective movement plan, for the fleet of tugs that minimises the risk of drift grounding accidents to happen.

At any given point in time, the GA can incorporate real-time information about the current situation, as well as a prediction of the future, to calculate an optimal set of patrolling tug trajectories. However, due to the dynamic nature of the environment and the parameters involved, the solution will quickly become outdated. We therefore require some feedback mechanism in the algorithm that can update the solution with changes in ocean conditions such as wind, current, and waves, as well as speed and direction of oil tankers. The mechanism we adopt is the principle of RHC.

From control theory, it is known that RHC, which is also called model predictive control (MPC), is one of very few control methods able to handle constraints in the design phase of a controller and not via post hoc modifications (e.g., see Goodwin et al. (2001); Maciejowski (2002); Rossiter (2004)). For the TFO problem, one such constraint is the maximum speed of tugs, which is constrained by factors such as ship design and weather conditions. This maximum speed will necessarily limit the number of *reachable cross points*. Using RHC it is possible to constantly incorporate such constraints in the planning of tug patrol trajectories, even as conditions change.

In our RHGA, the GA component plans a set of tug trajectories starting at t_d and with a prespecified duration, namely the prediction horizon T_h introduced previously. However, the tugs only execute the very first time step of their trajectories. In the mean time, with a start time of $t_\mathrm{d}+1$, another set of of tug trajectories is planned, based on new and predicted information available. This new solution replaces the old one but again only the first portion is implemented. This process repeats as a sequence of planning steps, thus creating a feedback loop where updated information is fed back to the GA. Effectively, the prediction horizon keeps being shifted into the future, and this has led to the term receding horizon control.

A thorough presentation of our simulator framework and algorithm implementation is not possible within the the scope limitations of this paper. Whilst the RHGA was implemented in Matlab in earlier versions (Bye, 2012; Bye et al., 2010), we recently rewrote the entire code base in the advanced, purely-functional programming language Haskell (Bye and Schaathun, 2014), and have used this framework for the work presented here. For further details about various aspects of our implementation, we refer to Bye (2012); Bye and Schaathun (2014); Bye et al. (2010).

2.3 Cost Functions

Determining suitable cost functions for a TFO algorithm is a key design challenge for the algorithm to be successful. Below, we will present three possible cost functions, each of which can be configured to yield different properties by means of parameterisation.

2.3.1 Cost Function f_1

For the cost function we used in our earliest work (Assimizele et al., 2013; Bye, 2012; Bye et al., 2010), we employed a metric defined as the sum of the distances between all cross points and the *nearest* patrol points, based on the argument that if an oil tanker in drift can be saved by the nearest tug, then it is not relevant if the other tugs are also able to save the tanker, and if an oil tanker in drift cannot be saved by the nearest tug, then it cannot be saved by the other tugs either. This argument assumes that the tugs all have the same maximum speed. In this case, this metric is equivalent to minimum rescue time, since distances will be directly proportional to rescue times. If instead the tugs do not have identical maximum speeds, one can easily define rescue time as distance divided by maximum tug speed and add up the minimum rescue times for each cross point.

Recently, we decided to examine some other metrics, namely squaring the distances and also incorporating a safe zone (Bye and Schaathun, 2014). The effect of squaring the distance from a cross point to a patrol point is that cross points further away will be penalised more in the cost function. It makes intuitive sense that higher costs should be awarded to the cross points of tankers that are less likely to be saved. Similarly, tankers with cross points that are very close to the position of one or several tugs are very likely to be saved and can thus be ignored in the cost function. We incorporate these situations by the inclusion of a "power" parameter e and raise the distance to the power $e = 2$ for squaring, whereas we use a safe region parameter r to impose no penalty in the cost function for close cross points.

Combining our original cost function with the recent modifications, we obtain the cost function f_1 given by

$$f_1(t) = \sum_{t=t_\mathrm{d}}^{t_\mathrm{d}+T_\mathrm{h}} \sum_{o\in O} \max\left\{0, \min_{p\in P} \left|y_t^c - y_t^p\right|^e - r\right\} \quad (2)$$

for N_o oil tankers $o \in O = \{o_1, \ldots, o_{N_\mathrm{o}}\}$, N_p patrol tugs $p \in P = \{p_1, \ldots, p_{N_\mathrm{p}}\}$, $e \in \{1, 2\}$, and r chosen as some distance that can confidently be reached by a tug to enable drift interception and hookup to the ship. A reasonable and conservative choice for r could for instance be half the expected distance a tug can travel from an alarm is received until the first hypothetical cross points occur.

In terms of minimising this cost function for non-zero r, a challenge will be that of flat cost surface regions for cross points within the safe range, which makes it more difficult for the GA to find an optimal solution.

Note that letting $e = 1$ and $r = 0$ yields the cost function used in our earlier work (Assimizele et al., 2013; Bye, 2012; Bye et al., 2010).

2.3.2 Cost Function f_2

In Bye and Schaathun (2014), we identified a problem in cost function f_1, namely the lack of taking the detection delay $\delta = t_\mathrm{a} - t_\mathrm{d}$ into account. That is, there will be a delay from the time t_d when an oil tanker starts drifting until the VTS and its tugs are being alarmed at the time t_a some hours later. Cost function f_1 actually implicitly assumes that the VTS will be notified immediately when a ships starts drifting; an assumption that is clearly far too optimistic. Instead, we should assume that oil tankers typically have drifted for some time before the tugs are being alarmed, and consequently, we must define a new, and shorter, drift-from-alarm (DFA) time $\hat{\Delta}_\mathrm{a} = \hat{\Delta} - \delta$, which is the drift time from the tugs receive an alarm at t_a until the drifting tanker crosses the patrol line at a cross point. We will keep our somewhat arbitrary, but realistic, choice of $\delta = 3$ hours presented previously (Bye and Schaathun, 2014) in this paper.

In Bye and Schaathun (2014), we also discovered a serious flaw with cost function f_1 in that it implicitly assumes that tugs will continue to execute their original plans even after receiving an alarm, because it compares cross points and patrol points at the same times into the future. Instead, the tugs should of course abandon their original plans immediately upon an alarm about a drifting tanker, and make every effort to intercept it before it runs aground. Therefore, the cost function should compare the positions of the tugs (patrol points) *when* they receive the alarm at time t_a, and the hypothetical future positions where drifting tankers will cross the patrol line some $\hat{\Delta}_\mathrm{a}$ hours later, where $\hat{\Delta}_\mathrm{a}$ is the total drift time $\hat{\Delta}$ (8–12 hours) less the detection delay δ (3 hours), leaving $\hat{\Delta}_\mathrm{a}$ in the range 5–9 hours.

To address the issues raised above regarding the original cost function f_1, we propose a modified cost function f_2 given by

$$f_2(t) = \sum_{t=t_\mathrm{a}}^{t_\mathrm{a}+T_\mathrm{h}} \sum_{o\in O} \max\left\{0, \min_{p\in P} \left|y_{t+\hat{\Delta}_\mathrm{a}}^c - y_t^p\right|^e - r\right\}. \quad (3)$$

Compared with f_1, we observe that the cost evaluation starts at the notification time of alarm t_a, and not at the time of start of drift t_d, and that in the distance term, we measure the distance between each cross point at some future cross time $t + \hat{\Delta}_\mathrm{a}$ and the position of the nearest tug at the alarm time t, and do this for the current alarm time $t = t_\mathrm{a}$ and future potential alarm times $t = t_\mathrm{a} + 1, t_\mathrm{a} + 2, \ldots, +t_\mathrm{a} + T_\mathrm{h}$.

2.3.3 Cost Function f_3

In contrast with f_1 and f_2 above, another and probably more realistic cost funtion f_3 is simply the number of unsalvageable tankers. That is, from a pragmatic point of view, we merely want to consider whether a tug can reach a drifting tanker in time to prevent it from grounding, and the distance is otherwise immaterial.

We may use the safe range r for counting the number of unreachable cross points and let this number constitute a measure for unsalvageable tankers. If cross points are outside the safe range, we add 1 to the accumulated cost, otherwise we add 0. The cost function f_3 can then be described by

$$f_3(t) = \sum_{t=t_\mathrm{a}}^{t_\mathrm{a}+T_\mathrm{h}} \sum_{o\in O} g\left(\min_{p\in P} \left|y_{t+\hat{\Delta}_\mathrm{a}}^c - y_t^p\right| - r\right), \quad (4)$$

$$g(x) = \begin{cases} 1, & x > 0 \quad (\text{outside } r), \\ 0, & x \leq 0 \quad (\text{inside } r), \end{cases} \quad (5)$$

where $g(x)$ is the Heaviside unit step function.

Note that cost function f_3 accumulates a binary penalty (1 or 0) and relies on a safe region r only, which makes it much more difficult to optimise for the GA than f_1 and f_2 due to numerous plateaus of flat cost surface regions in the cost landscape that the GA searches through.

2.4 Algorithm Evaluation

How can we compare the performance of different TFO algorithms, or in our case, the performance of the RHGA configured with various different cost functions? By definition, the metrics of different cost functions are not generally directly comparable, and it is not always possible to use a cost function which directly reflects the real cost of the solution. In the TFO problem, there are many random elements without well-understood probability models. Incorporating these elements in the cost function would make it too complex to be practical.

2.4.1 Simulation Framework

The solution is a Monte Carlo simulation as shown in Figure 3, where the complete optimisation algorithm with cost function can be tested against a large number of (pseudo) random scenarios.

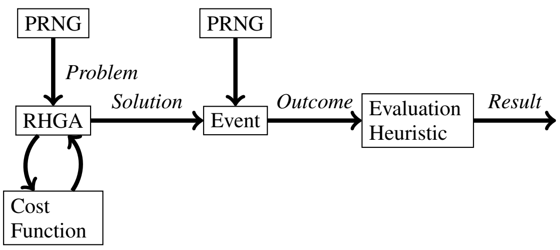

Figure 3: Simulation model. A pseudo-random number generator (PRNG) generates simulation scenarios in which the RHGA uses some cost function Cost 1 to determine a solution (where tugs should move). The PRNG then generates an event of drift, which depending on the current position of the tugs may be critical or not. The outcome (saving or not saving the drifting ship) is then quantified as Cost 2 by an evaluation heuristic.

We have two pseudo-random algorithms (PRNG); one to generate the problem as observed by the optimisation algorithm (RHGA), and one to generate the situation, or event, where the solution is to be executed. For example, the positions of the tankers is known *a priori*, and is part of the problem. A tanker starting to drift is an *event* which is only known after the RHGA has provided the solution. Given the solution and the event, we can evaluate the cost of the result, e.g., of a grounding accident or a successful rescue.

The evaluation heuristic may look similar to the cost function, but there is a critical difference. The evaluation heuristic evaluates the cost of a particular event. The cost function has to evaluate a solution without the knowledge of which event will occur.

There are several stochastic processes governing the outcome of an event in the model. These processes can be internalised either in the Monte Carlo simulation or integrated analytically in the evaluation heuristic. We can illustrate this with an example. The event in the simulation model can be subdivided into two stages:

1. Oil tanker o starts drifting.
2. Oil tanker o grounds.

If the first event occurs, one ore more patrol tugs will attempt to rescue the drifting tanker. This rescue operation may or may not succeed, depending largely on the maximum tug speed as determined by weather conditions. If it does not, the second event occurs.

One possibility is to stop the Monte Carlo estimation after the first event, and let the cost be equal to the conditional probability of the second event, that is, the probability of a failed rescue operation. The second possibility is to simulate the entire rescue operation. If it succeeds the cost is zero, otherwise it is the cost of the grounding accident, which may depend on the type of cargo, geographical location, weather conditions, and so on.

For the purpose of this work, we evaluate the result when tanker o starts drifting. We do not simulate the rescue operation. The steps of the evaluation method can be summarised as follows:

1. randomly generate a deterministic and reproducible simulation scenario;
2. run the RHGA (or another TFO algorithm) for a given number of planning steps;
3. considering each oil tanker separately, assume each tanker begins drifting and count the number of salvageable tankers;
4. for the same simulation scenario, repeat (2) and (3) with a different cost function configuration in the RHGA (or a different TFO algorithm); and
5. repeat steps (1)–(4) for a number of different simulation scenarios and find the accumulated evaluation cost for each RHGA configuration (or TFO algorithm).

Note that instead of evaluating one random event, we evaluate one event for each tanker o, where o starts to drift. This is possible because the number of tankers is small, and it lets us evaluate a larger number of scenarios with little extra time. We propose two candidate evaluation heuristics h_1 and h_2.

2.4.2 Evaluation Heuristic h_1

The first heuristic is similar to cost function f_3 counting the number of salvagable tankers at some alarm time t_a. We will simply assume that each patrol tug p can save any ship with cross points inside the safe region $r = v^p_{\max}\hat{\Delta}_\mathrm{a}$ away, where $\hat{\Delta}_\mathrm{a}$ is the DFA time and $v^p_{\max}$ is the pth tug's maximum speed, that is, within the maximal reach of a tug upon a drift alarm. In a more realistic model, the heuristic should probably be weather dependent and direction dependent, e.g., going against the wind is slower than going with it. It should also include hookup times.

A simulation scenario in this case is simply a set of pre-determined oil tanker movements and the resulting hypothetical drift trajectories and cross points for a pre-specified duration. For testing purposes, we can generate a number of such scenarios offline and use them as input data for testing TFO algorithms. In a real-world application, the actual scenario is what that is happening right now, and future oil tanker positions, drift trajectories, and cross points would have to be predicted in real-time.

To sum up, we define the evaluation heuristic h_1 as

$$h_1(t_\mathrm{a}) = \sum_{o\in O} g\left(\min_{p\in P}\left|y^c_{t_\mathrm{a}+\hat{\Delta}_\mathrm{a}} - y^p_{t_\mathrm{a}}\right| - r\right), \tag{6}$$

$$r = v^p_{\max}\hat{\Delta}_\mathrm{a} \tag{7}$$

$$g(x) = \begin{cases} 1, & x > 0 \quad \text{(outside } r\text{)}, \\ 0, & x \le 0 \quad \text{(inside } r\text{)}. \end{cases} \tag{8}$$

Other possible objective measures exist, e.g., we could sum up the total fuel consumption and use it as a component of an overall objective measure if that is of interest. Furthermore, the cost of measuring the number of salvagable tankers does not need to be discrete (yes/no) but could instead have a continuous probability distribution attached to it. We could then sum these probabilities to find an evaluation cost for the TFO algorithm.

2.4.3 Evaluation Heuristic h_2

The evaluation heuristic h_1 does not discriminate between cross points far away from the nearest tug and cross points that are much closer, as long as they are all inside the maximal reach from any tug as given by $r = v^p_{\max}\hat{\Delta}_\mathrm{a}$. However, it is clear that due to varying and non-optimal weather conditions, the maximum speed of each tug may be much lower than during ordinary operation. Moreover, h_1 does not take into account that there will be a hookup time when the tug attaches itself to the drifting ship. In an attempt to address these issues, we suggest the following evaluation heuristic h_2 given by

$$h_2(t_\mathrm{a}) = \sum_{o\in O}\left(\max\left\{0, \min_{p\in P}\left|y^c_{t_\mathrm{a}+\hat{\Delta}_\mathrm{a}} - y^p_{t_\mathrm{a}}\right| - r\right\}\right)^2, \tag{9}$$

$$r = v^p_{\min}\hat{\Delta}_\mathrm{a}, \tag{10}$$

where the safe region has been reduced to the area reachable for any tug with some minimum speed $v^p_{\min}$, which we assume the tug will always be able to maintain. Inside the safe region, there is zero cost for cross points of salvageable tankers, whereas outside, the cost increases with the square of the distance to cross points of unsalvageable tankers. Squaring ensures that we punish larger distances more.

3 SIMULATION STUDY

3.1 Basic Parameters

Until the end of 2013, the NCA have been using $N_\mathrm{p} = 3$ tugs for patrolling the northern Norwegian coast. However, since the beginning of 2014, the number of tugs have been reduced to $N_\mathrm{p} = 2$. In previous papers, we have consistently assumed three tugs in the fleet. Using our evaluation heuristics, we are able to examine by means of simulations whether there are any differences in coastal protection (as we have defined it by our evaluation heuristics) from this reduction in fleet size. For completeness, and in order to test our propositioned evaluation heuristics, we will also examine tug fleets of four, five, or six tugs, as well as the case of a single patrol tug, hence $N_\mathrm{p} \in \{1,\ldots,6\}$.

The stretch of coastline patrolled by the tug vessels that we have termed the patrol line y is about $1,500$ km long. Hence, we define our patrol zone Y as unidimensional along y in the continuous interval $Y = [-750, 750]$ km, and constrain cross points y^c and patrol points y^p to lie in Y, or $y^c, y^p \in Y$. For implementation purposes, we also define a tanker zone Z in the same interval much further away from land but underline that the (simulated) VTS will still observe ship traffic outside of this zone. The reason is of course that a tanker outside the tanker zone Z may still drift and ground inside the patrol zone Y.

For simplicity, we will assume a patrol zone with "hard" borders, where cross points outside the patrol zone are ignored, but as argued in Section 2.1, we do realise that in reality, the VTS will also consider potential cross points outside of such borders.

Based on historical traffic data, we have previously assumed that the typical average number of tankers sailing along z and being watched by the Vardø VTS is $N_o = 6$. From communication with the NCA, the actual number can be less on certain days but also higher, especially if we add high risk ships other than oil tankers that may also be watched carefully. Indeed, as mentioned in the introduction, there were close to 1600 risky transports in the area in 2013 alone, of which about 300 were carrying petroleum-related cargo (Vardø VTS, 2014b). Additionally, it is desirable to obtain results comparable with our previous work. Thus we keep this number of oil tankers unchanged in this study.

Under normal conditions, we assume that the patrol tugs are limited to a maximum speed of $v^p_{\max} =$ 20 km/h, whereas the speed of each oil tanker v^o is randomly drawn from a uniform distribution such that $v^o \in [20, 30]$ km/h. Note that compared with previous work, we have reduced the maximum speed of the tugs from 30 km/h to 20 km/h, thus making it more difficult for tugs to cover potential cross points. These speeds are in line with the literature (e.g., Det Norske Veritas (2009), Eide et al. (2007a)). We also assume that even in very bad weather conditions, the tugs are able to maintain at least a minimum speed of $v^p_{\min} = 5$ km/h.

Drift trajectories are set to be perpendicular (eastbound) onto the south-north patrol line y, with associated drift times drawn randomly from the interval $\{8, 9, \ldots, 12\}$ hours and cross points generated from extrapolating the predicted future positions of oil tankers and their resulting drift trajectories.

The parameter settings are summarised in Table 1.

3.2 Simulation Scenarios

A simulation scenario consists of simulation-generated tanker movements along z as well as hypothetical drift trajectories with corresponding cross points on y. The scenario acts as an input to a TFO algorithm such as the RHGA and is completely independent of what the RHGA calculates and how the tugs move.

We initialise a scenario by placing N_o oil tankers at random positions and with random speeds along z, headed in either the southbound or the northbound direction. Next, we sample each of the tankers' positions, speeds, and directions at every simulation step $t_s = 1$ h from the start of the simulation at $t_i = 0$ h to the final simulation time at $t_f = 24$ h. For any simulation time t_d in $\{t_i, t_i + t_s, \ldots, t_f\}$, we suppose that we have precise real-time information about the speed and direction of each oil tanker, as provided by AIS. We also assume that we have an accurate model that, given this real-time actual information, is able to predict future positions and speeds of the tankers at future times $t_d + t_s, t_d + 2t_s, \ldots, t_d + T_h$, where T_h is the prediction horizon. Finally, we assume that we have another accurate model that is able to predict hypothetical drift trajectories and cross points for each tanker if it starts drifting at time t_d and also at the future times just listed.

Note that since the scenario consisting of oil tanker movements, drift trajectories, and cross points is independent of how the fleet of tugs move, we can replay the same scenario as an input to other TFO algorithms (or variations of the RHGA) in order to evaluate and compare the algorithms.

3.3 TFO by the RHGA

For all scenarios, the N_p tugs are initialised at simulation time $t_i = 0$ by being uniformly positioned along the coast at stationary base stations in a manner such that they can cover as much of the patrol line y as possible. For example, since we have defined y as a line constrained to $[-750, 750]$ km, a single tug will be placed at $y^p_{t_i} = 0$, a fleet of two tugs will be placed at $y^p_{t_i} = \{-375, 375\}$, a fleet of three tugs will be placed at $y^p_{t_i} = \{-500, 0, 500\}$, and so on. From these initial positions, tugs will begin to actively pursue good positions for reducing the risk of drift grounding accidents depending on how the scenario plays out and how the TFO algorithm will control them.

At any simulation time t_d, the GA component in the RHGA uses the predicted distribution of potential cross points to calculate a plan, which consists of a position trajectory for each of the tugs in the fleet. The plan consists of future desired positions for each tug at times $\{t_d + 1, t_d + 2, \ldots, t_d + T_h\}$. The plan is optimal (or close to optimal) in the sense that it minimises (or tries to minimise) a cost function.

Using RHC, we let the tugs execute only the first step of this plan and move the tugs from their positions at $t = t_d$ to future positions at $t = t_d + 1$. At $t = t_d + 1$, the GA plans a new set of desired trajectories from $t = t_d + 2$ to $t = t_d + T_h + 1$, but again, we let the tugs execute only the first step from $t = t_d + 1$ to $t = t_d + 2$. This process repeats until the final simulation time $t_f = 24$, at which we plan for $t = t_f, t_f + 1, \ldots, t_f + T_h$, and again, and finally, let the tugs execute only the first step from $t = t_f$ to $t = t_f + 1$.

We have then completed one simulation of this particular scenario using one particular TFO algorithm, in our case, the RHGA employing a particular configuration of one of the cost functions f_1–f_3. The end-of-simulation positions of tugs and tankers and their cross points are then be used as input to the evaluation heuristics.

Table 1: Simulation parameters, settings, and units.

Parameters	Settings	Units
Patrol zone (south-north line)	$Y = [-750, 750]$	km
Tanker zone (south-north line)	$Z = [-750, 750]$	km
Number of oil tankers	$N_\mathrm{o} = 6$	-
Set of oil tankers	$O = \{1, 2 \ldots, N_\mathrm{o}\}$	-
Number of tugs	$N_\mathrm{p} = \{1, \ldots, 6\}$	-
Set of tugs	$P = \{1, 2 \ldots, N_\mathrm{p}\}$	-
Initial tug positions (base stations)	Uniformly distributed	km
Random initial tanker positions	$y^\mathrm{o} \in Z, \forall o \in O$	km
Maximum speed of tugs	$v^p_\mathrm{max} = 20,\ \forall p \in P$	km/h
Minimum speed of tugs	$v^p_\mathrm{min} = 5,\ \forall p \in P$	km/h
Random speed of oil tankers	$v^o \in [20, 30],\ \forall o \in O$	km/h
Initial simulation time	$t_\mathrm{i} = 0$	h
Simulation step	$t_\mathrm{s} = 1$	h
Final simulation time	$t_\mathrm{f} = 24$	h
Prediction horizon	$T_\mathrm{h} = 24$	h
Time of start of drift	$t_\mathrm{d} \in \{t_\mathrm{i}, t_\mathrm{i}+1, \ldots, t_\mathrm{f}\}$	h
Detection delay	$\delta = 3$	h
Alarm time	$t_\mathrm{a} = t_\mathrm{d} + \delta \in \{t_\mathrm{i}+\delta, t_\mathrm{i}+\delta+1, \ldots, t_\mathrm{f}+\delta\}$	h
Drift direction	Eastbound	-
Estimated drift times	$\hat{\Delta} \in \{8, 9, \ldots, 12\}$	h
Drift-from-alarm (DFA) times	$\hat{\Delta}_\mathrm{a} = \hat{\Delta} - \delta \in \{5, 6, \ldots, 9\}$	h
Static strategy	$y^p_t = y^p_{t_i},\ \forall t$	km
Cost functions	$F = \{f_1, f_2, f_3\}$	-
Distance power	$e = \{1, 2\}, \quad \text{in} \quad f_1, f_2$	-
Safe region	$r = \begin{cases} \{0, 50, 100\}, & \text{in} \quad f_1, f_2 \\ \{50, 100\}, & \text{in} \quad f_3 \\ v^p_\mathrm{max}\hat{\Delta}_\mathrm{a} = [100, 180], & \text{in} \quad h_1 \\ v^p_\mathrm{min}\hat{\Delta}_\mathrm{a} = [25, 45], & \text{in} \quad h_2 \end{cases}$	km
TFO algorithms	Configurations of RHGA$(f_i, e, r, N_\mathrm{p})$	-
Number of RHGA$(f_i, e, r, N_\mathrm{p})$ configurations	$N_\mathrm{conf} = 15$	-
Number of scenarios	$N_\mathrm{sc} = 1600$	-
Total number of simulations	$N_\mathrm{sim} = N_\mathrm{conf} \times N_\mathrm{sc} \times \dim N_\mathrm{p} = 144{,}000$	-

3.4 GA Description and Settings

The GA we employ in this study is based on the continuous GA presented in Haupt and Haupt (2004) and has been presented in detail in our previous work (Bye, 2012; Bye et al., 2010). We initialise the GA with a population size of chromosomes that are randomly generated. At every iteration of the GA, $N_\mathrm{keep} = 10$ chromosomes are selected from the population by roulette wheel selection, with low cost chromosomes having a greater chance of being picked. These chromosomes survive from one generation to the next and are also used for mating to generate new offspring that replace the chromosomes that were not picked. Mating is performed by a combination of an extrapolation method and a single crossover point to obtain new offspring variable values bracketed by the parents' variable values (see Haupt and Haupt, 2004, for details). After mating, the new population of chromosomes is ranked and the $N_\mathrm{elite} = 10$ best chromosomes are categorised at elite chromosomes and are not allowed to mutate. Of the remaining non-elite chromosomes, each has a mutation rate $\mu = 0.1$ probability of being mutated. After mutation has taken place, the GA repeats the process for a total of $N_\mathrm{iter} = 200$ iterations, after which the best solution obtained is used for moving the tugs one step ahead as per the RHC strategy presented above.

We chose these GA parameter settings by manually evaluating a number of test runs, where we were able to find suitable settings that ensured satisfactory run times for all RHGA configurations while at the same time obtaining satisfactory minimisation of cost functions. That is, only negligible improvements were attainable from tuning the GA to other, often more time-consuming settings, e.g., increasing the population size or number of iterations.

Our choice of GA settings are summarised in Table 2.

3.5 RHGA Configurations

The three cost functions f_1, f_2, and f_3 are parameterised by a distance power e and a safe region r,

Table 2: GA settings.

Parameters	Settings
Population size	$N_{\mathrm{pop}} = 50$
Chromosomes kept for mating	$N_{\mathrm{keep}} = 10$
Elite chromosomes	$N_{\mathrm{elite}} = 10$
Mutation rate	$\mu = 0.1$
Number of iterations	$N_{\mathrm{iter}} = 200$

whereas the evaluation heuristics h_1 and h_2 are parameterised by a safe region r, which in turn is a function of $v^p_{\max}$ or $v^p_{\min}$ for h_1 or h_2, respectively.

We decided to implement and evaluate 14 different configurations of the cost functions for the RHGA and evaluate each configuration using both the evaluation heuristics. In addition, we wanted to evaluate a static strategy, in which tugs are stationary at base stations and do not move until notified about a drifting ship. For convenience in our data processing, we have labelled the static strategy as configuration #0 and defined its configuration as f_0, with e and r both set to zero. For the same reason, we have set $e = 0$ for f_3, where e is not applicable. Each configuration can be thought of as a unique TFO algorithm. Indeed, our approach generalises to the evaluation of any TFO algorithm able to calculate tug fleet control decisions based on the parameters, settings, and input scenarios that we have described above.

The 15 configurations are summarised in Table 3.

Table 3: RHGA configurations.

Cost function f_i	Power e	Safe region r	#
0	0	0	1
1	1	0	2
		50	3
		100	4
	2	0	5
		50	6
		100	7
2	1	0	8
		50	9
		100	10
	2	0	11
		50	12
		100	13
3	0	50	14
		100	15

4 RESULTS

We randomly generated $N_{\mathrm{sim}} = 1600$ unique simulation scenarios and tested the performance of tug fleet optimisation for each of the 15 RHGA configurations given in Table 3 when faced with $N_{\mathrm{p}} = \{1, \ldots, 6\}$ tugs to control, yielding a grand total of 144,000 simulations. For each configuration, that is, each combination of cost function f_i, distance power e, and safe region r, denoted as $f_i(e, r)$, the sample mean, standard deviation, coefficient of variance (relative standard deviation), standard error (standard deviation of the sample mean), and relative standard error for the evaluation heuristics h_1 and h_2 were calculated for $N_{\mathrm{p}} = \{1, \ldots, 6\}$ tugs, respectively. Both heuristics were evaluated at the end of each simulated scenario at $t = t_{\mathrm{f}}$. In the sections below, the results of the sample means as well as comparisons of the active control configurations of the RHGA versus the static strategy are our main concern and will be presented graphically, whereas the other statistics will be presented briefly in text.

4.1 Evaluation Heuristic h_1

The evaluation heuristic h_1 is a measure of the number of unsalvageable tankers. Figure 4 shows the maximum (worst performance) and minimum (best performance) sample mean $\bar{h}_1$ of cost functions f_1–f_3 over all configurations (combinations of power e and safe region r), as well as the static strategy, evaluated for 1–6 tugs. Unsurprisingly, the size of the tug fleet

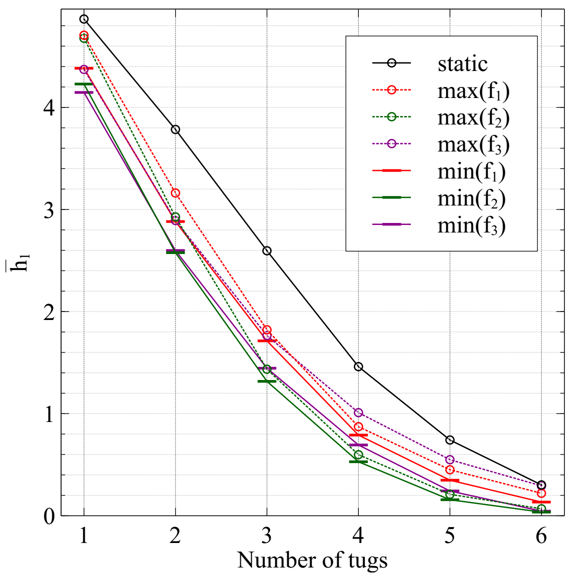

Figure 4: Maximum and minimum mean h_1 evaluated for 1–6 tugs and cost functions f_1–f_3 and the static strategy.

strongly affects h_1. With a single tug, $\bar{h}_1$ was in the range [4.15–4.87], and then decreased with the number of tugs to the range [0.034–0.30] for six tugs.

For all configurations with 1–3 tugs, the standard deviation showed no trend and was in the range [0.99–1.25], whereas it was decreasing with the number of tugs for configurations with 4–6 tugs and ranged from 0.19 to 0.91. The standard error of the mean was in the range [0.005–0.032] for all configurations, with typically smaller values for smaller means. The relative standard error (found by dividing by the mean) was small for all configurations, increased with number of tugs (and thus smaller means), and was in the range [0.0051–0.13].

As expected, all RHGA configurations of f_1–f_3 outperform the static strategy for the same number of tugs. In addition, when the RHGA is configured with a tug fleet with one tug less than the static strategy, the following observations are made: With a single tug, no RHGA configuration is able to outperform the static strategy with two tugs; with two tugs, the best configurations of f_2 and f_3 outperform the static strategy with three tugs; and with 3–5 tugs, all configurations of f_2 and the best configuration of f_3 outperform the static strategy with 4–6 tugs, respectively.

Comparing the RHGA configurations with respect to the number of tugs, we observe the following: For a single tug, f_3 has a better respective minimum and maximum performance than the other cost functions; for two tugs, f_2 and f_3 have approximately equal minimum and maximum performance; for 3–6 tugs, f_2 has a better minimum and maximum performance than the other cost functions. Also, for any number of tugs, the best configuration of f_3 is with safe region $r = 100$ and the worst is with $r = 50$. For any number of tugs, the best configuration of f_3 is not much worse than that of f_2 while the worst configuration of f_3 is clearly worse than that of both f_1 and f_2 for 4–6 tugs.

Finally, we note that when compared with f_2, f_1 has a similar trend and relationship between maximum and minimum $\bar{h}_1$ with increasing number of tugs but consistently with worse performance.

4.1.1 Comparison with Static Strategy

Figure 5 shows the normalised mean of h_1 for all configurations of cost functions f_1 (left), f_2 (middle), and f_3 (right) evaluated for 1–6 tugs and normalised by dividing the results with those of the static strategy.

For both cost functions f_1 and f_2 and 1–3 tugs, a power setting of $e = 1$ has better performance than $e = 2$, whereas there is a slight overall performance improvement for $e = 2$ for 4–6 tugs. The overall best safe region setting is $r = 50$.

For f_3, a safe region of $r = 100$ performs well,

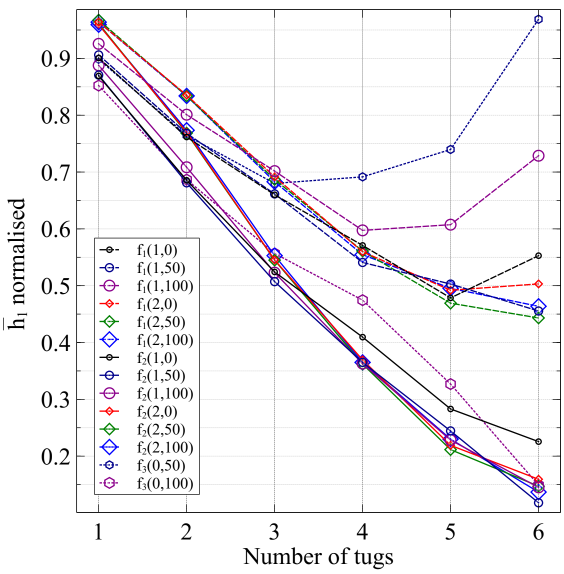

Figure 5: Mean h_1 for cost functions f_1 (left), f_2 (middle), and f_3 (right), normalised by the static strategy.

whilst $r = 50$ performs badly, especially with increasing number of tugs.

Finally, only f_2 (all configurations) and f_3 ($r = 100$) is able to steadily improve its normalised performance with increasing number of tugs, reaching an improvement of 73–88% compared to the static strategy for any configuration with a tug fleet of six tugs.

4.2 Evaluation Heuristic h_2

Figure 6 shows the same results as that of Figure 4 but for the evaluation heuristic h_2, which is a measure of the sum of squared distances to cross points of unsalvageable tankers.

As for h_1, the size of the tug fleet strongly affects this evaluation heuristic. With a single tug, $\bar{h}_2$ was in the range $[8.5\text{–}9.7]\cdot 10^5$, and then decreased with the number of tugs to the range $[3.7\text{–}9.5]\cdot 10^3$ for six tugs.

Generally, for any configuration, the standard deviation decreased with the number of tugs, ranging from $3.3 \cdot 10^3$ to $5.1 \cdot 10^5$. The standard error of the mean was in the range from 82 to $9.3 \cdot 10^3$ for all configurations, with typically smaller values for smaller means. The relative standard error was small for all configurations and in the range [0.0091–0.026].

Due to the large difference in magnitude of $\bar{h}_2$ (due to the square term in the heuristic) depending on the number of tugs, we have plotted $\bar{h}_2$ on a logarithmic scale to enhance readability. The results are similar to those for evaluation heuristic h_1, with the same relationships between the various cost functions and configurations. The exception is f_3 when employed with

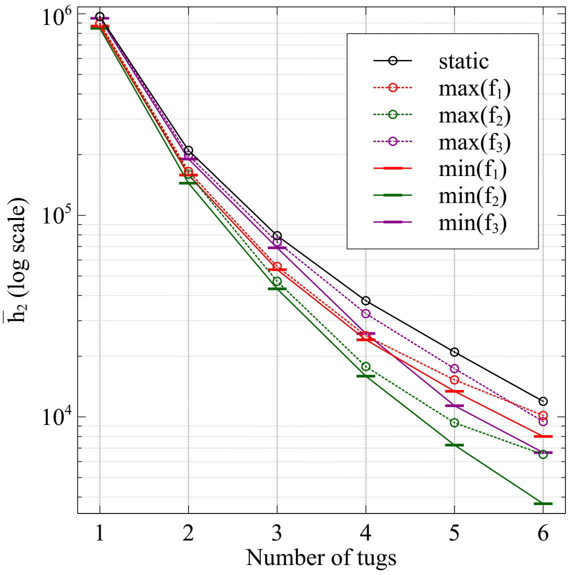

Figure 6: Maximum and minimum mean h_2 evaluated for 1–6 tugs and cost functions f_1–f_3 and the static strategy.

1–4 tugs, for which its relative performance compared to the other cost functions is worse than when evaluated with h_1.

4.2.1 Comparison with Static Strategy

Figure 7 shows the same results as that of Figure 5 but for the evaluation heuristic h_2.

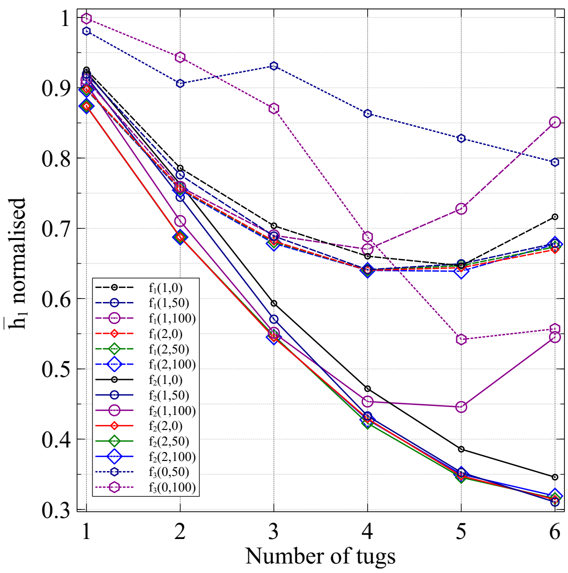

Figure 7: Mean h_2 for cost functions f_1 (left), f_2 (middle), and f_3 (right), normalised by the static strategy.

For both cost functions f_1 and f_2 and any number of tugs, with the exception of a safe region of $r = 50$, a squared power setting of $e = 2$ has better performance than $e = 1$, especially for 4–6 tugs.

For f_3, a safe region of $r = 50$ performs better for 1–2 tugs, whilst letting $r = 100$ is better for 3–6 tugs.

Finally, only f_2 (all configurations except $f_2(1, 100)$) is able to steadily improve its normalised performance with an increasing number of tugs, reaching an improvement of 65–69% compared to the static strategy for any configuration with a tug fleet of six tugs.

5 DISCUSSION

Both evaluation heuristics are able to quantify the performance of TFO algorithms designed to solve the TFO problem as defined in this paper. The small standard error for both heuristics is obtained by employing a large number of simulation scenarios (total 1,600) and means that the uncertainty in the means of both h_1 and h_2 is small. This is important in order to reliably measure the performance of TFO algorithms.

The general effect of increasing the number of oil tankers is that a static strategy will become increasingly suitable, whereas a dynamic scheme such as the RHGA configurations tested here will become less important. Thus, it is very impressive that cost function f_2 is able to increase its performance relative to the static strategy as measured by both evaluation heuristics, even with five or six tugs, and for all its configurations. The evaluation heuristics show that the RHGA configured with cost function f_2 has the best overall performance, with most configurations outperforming the other cost functions. The best choice of safe region for f_2 was $r = 50$, whereas the best power setting was $e = 1$ for h_1 and $e = 2$ for h_2, for for 1–3 or 1–4 tugs, respectively. With these configurations, f_2 was able to outperform the static strategy even with one less tug.

Cost function f_3 also performs well if $r = 100$, and is comparable with f_2 when evaluated by h_1 and to a lesser extent when evaluated by h_2. However, if $r = 50$, f_3 is the worst of all the RHGA configurations. The similarities between cost function f_3 and h_1 in measuring the number of tankers are salvageable or not is probably what makes f_3 perform better for h_1 than for h_2.

Cost function f_1 is similar to but consistently worse than f_2 and and the best configuration of f_3 and should be rejected. We propose that this is a direct result of its flaw that we have documented previously (Bye and Schaathun, 2014).

5.1 Future Work

The main hurdle before our RHGA can be used in real-world systems is to test and verify it under realistic conditions. This includes considering historical data of oil tanker traffic, realistic estimates of the variable maximum tug speeds attainable under various conditions, realistic drift trajectories and cross point distributions, downtime of tugs due to secondary missions or change of crew, and so on. It may also be necessary to extend the algorithm to 2D, in particular high risk scenarios where oil tankers enter or leave port and therefore are much closer to land than when sailing along the TSS corridor. Although challenging, we do welcome the prospect of TFO algorithms being adopted as decision-support tools for VTS centres around the world.

ACKNOWLEDGEMENTS

The DRAMA research group is grateful for the support provided by Regionalt Forskningsfond Midt-Norge and the Research Council of Norway through the project *Dynamic Resource Allocation with Maritime Application* (DRAMA), grant no. ES504913.

References

Assimizele, B., Oppen, J., and Bye, R. T. (2013). A sustainable model for optimal dynamic allocation of patrol tugs to oil tankers. In *Proceedings of the 27th European Conference on Modelling and Simulation*, pages 801–807.

Breivik, Ø. and Allen, A. (2008). An operational search and rescue model for the Norwegian Sea and the North Sea. *Journal of Marine Systems*, 69(1–2):99–113.

Breivik, Ø., Allen, A., Maisondieu, C., and Roth, J. (2011). Wind-induced drift of objects at sea: The leeway field method. *Applied Ocean Research*, 33(2):100–109.

Bye, R. T. (2012). A receding horizon genetic algorithm for dynamic resource allocation: A case study on optimal positioning of tugs. *Series: Studies in Computational Intelligence*, 399:131–147. Springer-Verlag: Berlin Heidelberg.

Bye, R. T. and Schaathun, H. G. (2014). An improved receding horizon genetic algorithm for the tug fleet optimisation problem. In *Proceedings of the 28th European Conference on Modelling and Simulation*, pages 682–690.

Bye, R. T., van Albada, S. B., and Yndestad, H. (2010). A receding horizon genetic algorithm for dynamic multi-target assignment and tracking: A case study on the optimal positioning of tug vessels along the northern Norwegian coast. In *Proceedings of the International Conference on Evolutionary Computation (ICEC 2010) — part of the International Joint Conference on Computational Intelligence (IJCCI 2010)*, pages 114–125.

Det Norske Veritas (2009). Rapport Nr. 2009-1016. Revisjon Nr. 01. Tiltaksanalyse — Fartsgrenser for skip som opererer i norske farvann. Technical report, Sjøfartsdirektoratet.

Eide, M. S., Endresen, Ø., Breivik, Ø., Brude, O. W., Ellingsen, I. H., Røang, K., Hauge, J., and Brett, P. O. (2007a). Prevention of oil spill from shipping by modelling of dynamic risk. *Marine Pollution Bulletin*, 54:1619–1633.

Eide, M. S., Endresen, Ø., Brett, P. O., Ervik, J. L., and Røang, K. (2007b). Intelligent ship traffic monitoring for oil spill prevention: Risk based decision support building on AIS. *Marine Pollution Bulletin*, 54:145–148.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional.

Goodwin, G. C., Graebe, S. F., and Salgado, M. E. (2001). *Control System Design*. Prentice Hall, New Jersey.

Hackett, B., Breivik, Ø., and Wettre, C. (2006). Forecasting the Drift of Objects and Substances in the Ocean. In *Ocean Weather Forecasting: An Integrated View of Oceanography*, pages 507–523. Springer.

Haupt, R. L. and Haupt, S. E. (2004). *Practical Genetic Algorithms*. Wiley, 2nd edition.

Havforskningsinstituttet (2010). Fisken og havet, særnummer 1a-2010: Det faglige grunnlaget for oppdateringen av forvaltningsplanen for Barentshavet og havområdene utenfor Lofoten. Technical report, Institute of Marine Research (Havforskningsinstituttet).

Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press, Oxford, England.

Maciejowski, J. M. (2002). *Predictive Control with Constraints*. Prentice Hall, first edition.

Rossiter, J. A. (2004). *Model-based Predictive Control*. CRC Press.

Sørgård, E. and Vada, T. (1998). Observations and modelling of drifting ships. Technical report, Det Norske Veritas Research, Høvik Norway.

Vardø VTS (2011). The Vardø Vessel Traffic Service – For increased safety at sea. Information pamphlet accessed at `http://kystverket.no/Om-Kystverket/Brosjyrer-skjema-og-andre-publikasjonar/Brosjyrer2/Brosjyre-om-Vardo-VTS/`.

Vardø VTS (2014a). Annual report on petroleum incidents. Technical report, Norwegian Coastal Administration.

Vardø VTS (2014b). Annual report on petroleum transport statistics. Technical report, Norwegian Coastal Administration.