To appear in *Engineering Optimization*
Vol. 00, No. 00, Month 20XX, 1–27

Mooring System Design Optimization Using a Surrogate Assisted Multi-Objective Genetic Algorithm

Ajit C. Pillai[a]* Philipp R. Thies[a] and Lars Johanning[a]

[a]*Renewable Energy Group, College of Engineering, Mathematics, and Physical Sciences, University of Exeter, Penryn, United Kingdom*

(*v3.0 draft July 2018*)

This article presents a novel framework for the multi-objective optimization of offshore renewable energy mooring systems using a random forest based surrogate model coupled to a genetic algorithm. This framework is demonstrated for the optimization of the mooring system for a floating offshore wind turbine highlighting how this approach can aid in the strategic design decision making for real-world problems faced by the offshore renewable energy sector. This framework utilizes validated numerical models of the mooring system to train a surrogate model, which leads to a computationally efficient optimization routine, allowing the search space to be more thoroughly searched. Minimizing both the cost and cumulative fatigue damage of the mooring system, this framework presents a range of optimal solutions characterizing how design changes impact the trade-off between these two competing objectives.

Keywords: offshore renewable energy; mooring system design; surrogate modelling; multi-objective optimization; reliability based design optimization

1. Introduction

As the offshore renewable energy sector progresses, it has become increasingly important to ensure that designs simultaneously generate the desired energy, survive in their energetic surroundings for their full lifetime, and remain cost effective. In the quest to satisfy these competing objectives, optimization techniques are now deployed in the design process to identify new design concepts while also aiding the system designer in strategic design decision-making. With progressively more offshore renewable energy devices exploring floating solutions, mooring systems have become one of the key subsystems which impacts both the survivability of the device and its costs (Weller et al. 2015; Thomsen et al. 2018). However, due to the computational time associated with the simulation of mooring systems it is not yet commonplace to deploy optimization algorithms in the design cycle. Without the use of numerical optimization methods, the design of mooring systems is limited to an iterative engineering design approach based on experience and engineering judgement. This often leads to innovative mooring designs not being considered, and the deployment of sub-optimal mooring designs (Johanning, Smith, and Wolfram 2006). In order to implement optimization techniques in complex engineering design problems, surrogate modelling, the use of simpler low fidelity models which approximate the high fidelity results at a lower computation cost, have emerged as an important technique to improve the computational time associated with these

*Corresponding author. Email: a.pillai@exeter.ac.uk

optimization schemes (Won and Ray 2005; Voutchkov and Keane 2006; Jin 2011).

The field of mooring optimization is a relatively nascent field which explores the optimal selection of mooring line materials, lengths, and diameters in order to elicit a desired response or minimize the cost associated with a floating system. As mooring systems represent an important component of offshore renewable energy devices which impact not only the motion dynamics of the device, and therefore how it interacts with the resource from which it is extracting energy, but also affects the cost of the overall system and governs the lifetime of the device (Weller et al. 2015). In the design of mooring systems, it is therefore common to select designs which minimize the cost or excursions subject to constraints on the tension in the lines, and the fatigue in the mooring system. Given this complex set of design considerations, an optimization approach and multi-objective optimization in particular would be appropriate in order to characterize the trade-offs between the competing design objectives and better inform decision making.

Existing work in the optimal design of mooring systems has explored the geometry optimization of the mooring system using a genetic algorithm to minimize the response of the moored vessels and platforms (Carbono, Menezes, and Martha 2005; Shafieefar and Rezvani 2007; Ryu et al. 2007; da Fonesca Monteiro et al. 2016; Ryu et al. 2016). However, as these studies have focused on vessels and platforms, these may not be the most appropriate optimizer objectives for an offshore renewable energy device. The recent work by Thomsen et al. (2018) has specifically explored the optimization of mooring systems for a wave energy converter considering the minimization of cost, however, the use of single objective optimization does not fully capture the complexity of the design problem. Offshore renewable energy devices must be both cost effective and achieve a specific device response in order to effectively harness the energy sources. Work by the authors has, therefore, explored multi-objective optimization of mooring systems for renewable energy platforms in order to highlight potential design trade-offs between the competing objectives that a device designer would face thereby offering information to allow the system designers to make more informed decisions (Pillai, Thies, and Johanning 2017, 2018b).

The assessment of mooring system designs is generally achieved through finite element analysis software operating in either the time domain or frequency domain (Davidson and Ringwood 2017). Time domain finite element models are capable of capturing the dynamic behaviour of the mooring lines and therefore play an important role in the design process. However, in order to effectively assess the response of the mooring behaviour, simulations must be executed for each operating condition and for sufficiently long simulations in order to adequately capture the dynamic behaviour during any operational sea state (Thomsen, Eskilsson, and Ferri 2017). Previous work by the authors has highlighted the importance of utilizing time domain simulations when designing mooring systems for renewable energy devices as these devices are characterized by more dynamic motion than vessels or platforms, therefore, requiring a simulation domain which can capture these dynamic effects and the impact that this has on the fatigue and design life of the mooring system. Mooring system optimization without surrogate models (Carbono, Menezes, and Martha 2005; Shafieefar and Rezvani 2007; Ryu et al. 2007; da Fonesca Monteiro et al. 2016; Ryu et al. 2016) tend to rely on frequency domain simulations which are significantly quicker and less computationally demanding than their time domain counterparts. Frequency domain methods, however, are not as effective in capturing the dynamic motion and loading of mooring systems which may play an important role in selecting appropriate mooring designs for offshore renewable energy applications (Kwan and Bruen 1991; Brown and Mavrakos 1999; Pillai, Thies, and Johanning 2018a).

For many optimization problems, the true objective function(s) are computationally costly. An effective approach to resolve this is to use a simpler objective function, a *surrogate*, which is correlated to the true objective, but computationally less expensive (Forrester, Sóbester, and Keane 2008). Surrogate modelling as a general term includes any model which substitutes for a high fidelity model in order to reduce computational time. These models can therefore attempt to model the underlying science with less detail or can be statistical models built from results using the full model (Forrester, Sóbester, and Keane 2007). Traditional forms of surrogate models include decision trees, support vector machines, radial basis functions, and artificial neural networks, however, there are also now many variations and hybrid approaches (Hastie, Tibshirani, and Friedman 2009; Forrester, Sóbester, and Keane 2008). Recent developments in the field of surrogate modelling in the context of optimization has explored the use of ensembles of surrogates to better define and characterize the search space (Forrester and Keane 2009; Forrester, Sóbester, and Keane 2007; Chugh et al. 2018; Shankar Bhattacharjee, Kumar Singh, and Ray 2016). Previous work in this field has focused on the development of generalized strategies which are relevant to a wide range of engineering problems, while the focus of the present paper is to demonstrate a specific methodology suitable to the mooring system design and optimization problem. The present work, therefore, focuses on the introduction and demonstration of the applicability of a specific methodology for this specific problem.

Surrogate models built for the assessment of the motions of a moored structure and the tensions in the mooring lines has generally made use of artificial neural networks (de Pina et al. 2013, 2016; Sidarta et al. 2017). The use of surrogate models for mooring system assessment, has, however, not been undertaken in the context of optimizing the mooring system.

This paper bridges these two areas of research implementing both a genetic algorithm for the geometry optimization of the mooring system of an offshore renewable energy platform while utilizing a surrogate model built using a machine learning technique in order to reduce the computational complexity of the optimizer evaluation function through a functional approximation architecture. The developed framework represents a pragmatic approach to the design of mooring systems offering a system designer the potential to make more informed decisions regarding the design of the mooring system. Though the optimization and surrogate models deployed are not on their own novel, their integration into a unified framework for the present mooring system design framework represents a novel implementation which is shown to aid the design process and marks an improvement on the present standard approaches.

In the design of mooring systems there are several objectives which are often considered including the cost of the mooring system, the tension in the lines relative to the minimum breaking load (MBL), the excursions of the floating body, or the cumulative fatigue damage. For the presented case study, the optimization routine seeks to minimize the cumulative lifetime fatigue damage in the mooring system and the material cost of the mooring system. These have been selected as they represent two important design criteria for mooring systems and especially for offshore renewable energy developers. Due to increasing challenges in many-objective optimization, the present implementation is as a bi-objective problem, though extensions including further objectives can be explored within the framework in the future in order to simultaneously consider additional objectives during the design process.

2. Mooring System Optimization Problem

The problem addressed in the present article explores the geometry optimization of a mooring system for an offshore renewable energy device. Offshore renewable energy devices extract energy from natural fluxes which cause some device motion relative to this natural flux, be it the blades of a wind turbine relative to the wind, a tidal turbine's rotor relative to the tidal current, or a wave energy device's active surface relative to the sea surface elevation. As a result of this, floating renewable energy devices, must ensure that their mooring systems are designed achieving the desired behaviour while at the same time not adversely impacting the reliability or cost of the overall system. The optimal design of mooring systems must therefore consider the site at which a device is being deployed, the specific device characteristics, the mooring system itself, and the interactions between these elements.

For each of the mooring lines considered in the system, the optimization routine selects the position of the mooring line anchor, the length of the mooring line, the material of each section of the mooring line, and the diameter of each section of the mooring line. These decision variables are given in table 1. The optimization routine does not explicitly select the number of mooring lines, but takes this as an input.

Table 1.: Description of Decision Variables

Variable	**Description**	**Variable Type**
$x_{l,i}$	length of section i of line l	Continuous
$y_{l,i}$	construction of section i of line l	Integer
α_l	anchor horizontal position for line l	Continuous
θ_l	anchor angle for line l	Continuous

Though the mooring system is defined using only a few variables for each line, this formulation is efficient in capturing the elements of interest to a mooring designer and can be used to characterize the mooring system for any floating body. In the present work, each line has been limited to consisting of maximum of three sections which can differ in diameter, material, or both. This limit has been selected in part as this represents the maximum number of sections often utilized for offshore renewable energy devices, and it allows a significant degree of flexibility to the optimization process. Given the flexibility of the framework, should a designer wish to consider a greater degree of flexibility in the designs then additional sections can easily be considered.

While the variables describing the section lengths and anchor position are continuous variables, the line type is a categorical representing which of the predefined line types is to be deployed. A detailed description of the constraints, and restrictions on the decision variables follows in section 2.3.

2.1 *Cumulative Fatigue Damage*

Engineering design must consider different failure modes in order to ensure that the design is fit for purpose. This includes the ultimate limit state (ULS) which considers the maximum extreme loads that the system must withstand, as well as the fatigue limit state (FLS) which considers the possible failure as a result of repeated cyclic loading at levels below the ULS (Schijve 2009). Offshore renewable energy devices seek to be deployed for a period up to 25 years which therefore requires reliable systems which can ensure device survival over this lifetime. The first objective explored in this optimization problem is therefore the fatigue damage in the mooring system. The fatigue damage is assessed using simulated tension time-series for each proposed mooring system for each of the anticipated sea states at the installation site. From this, rainflow counting of the tension cycles is done at each point along the lengths of the mooring lines.

Rainflow counting is a methodology used to evaluate fatigue damage for load cycles of varying amplitude. This method operates by identifying and counting the stress ranges corresponding to individual hysteresis loops. This is then used in combination with S-N or T-N curves which define the number of stress (S-N) or tension (T-N) cycles at a specific amplitude required for the material to reach failure. The Palmgren-Miner rule, shown in eq. (1), allows the individual contribution of each stress cycle to be summed in order to compute the cumulative fatigue damage (Rychlik 1987; Amzallag et al. 1994; Schijve 2009; Thies et al. 2014). The lifetime fatigue damage of the mooring lines is established by carrying out these calculations for each sea state that is expected at the site, and scaling the fatigue contributions based on the relative occurrence of the sea states over the operational lifetime of the device.

$$D(t) = \sum_{t_k<t} \frac{1}{N(S)} = \frac{1}{K} \sum_{t_k<t} (S)^{\beta} \tag{1}$$

where $D(t)$ is the fatigue damage, $N(S)$ is the number of cycles during time t, and S denotes the stress amplitudes established in the rainflow cycle count. The parameters K and β describe the fatigue properties of the material and are given by the S-N and T-N curves.

The cumulative fatigue damage, D_c is then given by:

$$D_c = \sum_{s\in\mathcal{S}} D_s \times \frac{T}{\tau_d} \times P(s) \tag{2}$$

where s represents a sea state from $\mathcal{S}$, the set of sea states which are simulated, T is the operational lifetime of the mooring system, τ_d is the simulation duration, and $P(s)$ is the probability of occurrence associated with sea state s. For each mooring line, the cumulative fatigue is computed at each point along the mooring line in order to consider the possible failure anywhere along the line and not exclusively at the fairleads. Though the highest tensions are experienced at the fairleads, the fatigue damage may be higher elsewhere in the system and it is important to consider the possible failure at any position along the mooring lines.

The objective, the minimization of the cumulative fatigue damage is explicitly given in eq. (4a) in the full problem formulation.

2.2 *Material Cost*

As cost effective solutions are sought, the second objective explored in the mooring design problem is the minimization of the material cost of the mooring lines. This is computed as a sum over the mooring lines by multiplying the unit cost of each line type (combination of material and diameter i.e. MBL) with the length of the line type deployed in the mooring system. In this way, this metric does not include any consideration of the anchors, and in fact the time-domain simulations do not affect this objective. This objective, the material cost of the mooring system, is, however, necessary as it represents a key metric that developers must consider when designing and deploying their mooring systems. The mooring system cost is computed using eq. (3) and the objective is given in eq. (4b) in the problem formulation.

$$C_l = \sum_{l\in\mathcal{L}} \sum_{i=1}^{\varepsilon_m} c(y_{l,i}) \cdot x_{l,i} \tag{3}$$

2.3 *Constraints*

In order to accurately model the design problem it is important to include constraints which limit the search space to feasible solutions and represent the real engineering limitations on the decision variables. Since the decision variables include the line specifications for each line as well as the anchor positions for each line's anchor, the genome is a mixture of various types.

The anchors are defined to be no further than 2500 m away from the floating body, and anchor lines are set to be within 30° of the original orientation defined in the simulation model (eqs. (4c) and (4d)). Specific constraints on the anchor positions will be site and project specific and these values have been selected for the present case study to illustrate the capabilities of the tool. The minimization of the mooring line costs will naturally try to limit the mooring footprint by bringing anchors in closer to the floating body, so this upper limit acts to aid the convergence of the optimizer. It is important to note that the present coupling to OrcaFlex does not simulate or model the anchors or any dynamics at the anchoring point and they are assumed to be a fixed point to the seabed.

Equation (4e) defines the length of mooring line to be the sum of the line segments and constrains this to be greater than zero to ensure that a mooring line is present while eq. (4f) imposes a constraint that the length of a mooring line cannot exceed the sum of the water depth and the horizontal distance to the anchor in order to ensure that the mooring line is not unrealistically long. Equation (4g) limits the tension along the length of the mooring line such that the minimum breaking load (MBL) of the line type at every location of the line is not exceeded. This constraint can optionally include F_s as a safety factor. Equation (4h) ensures that the line type for each line segment of each mooring line is one of those considered in the implementation of the optimization problem. Finally eqs. (4i) and (4j) define a set of points along each mooring line that are in contact with seabed during the dynamic simulation and limits these to chain constructions.

2.4 *Problem Formulation*

Given the decision variables, objectives, and constraints as described above, the full optimization problem can be formulated as follows:

$$\min \quad f_1(x) = max\left(\sum_{s\in\mathcal{S}} \left(D_c\left(x_l, y_l, \alpha_l, \theta_l, s\right) \cdot P(s)\right)\right) \qquad \forall l \in \mathcal{L} \tag{4a}$$

$$\min \quad f_2(x) = \sum_{l\in\mathcal{L}} \sum_{i=1}^{\varepsilon_m} c(y_{l,i}) \cdot x_{l,i} \tag{4b}$$

$$\text{s.t.} \quad \alpha_l \leq 2500 \qquad \forall l \in \mathcal{L} \tag{4c}$$

$$\theta_l \leq \phi_l \pm 30^\circ \qquad \forall l \in \mathcal{L} \tag{4d}$$

$$L_l = \sum_{i=0}^{\varepsilon_m} x_i \geq 0 \qquad \forall l \in \mathcal{L} \tag{4e}$$

$$L_l = \sum_{i=0}^{\varepsilon_m} x_i \leq \alpha_l + h \qquad \forall l \in \mathcal{L} \tag{4f}$$

$$t_{l,a} \leq MBL_{l,a} \times F_s \qquad \forall l \in \mathcal{L}; \quad \forall a \in]0, L_l]; \tag{4g}$$

$$\forall s \in \mathcal{S}$$

$$y_{l,a} \in \mathcal{A} \qquad \forall l \in \mathcal{L}; \quad \forall a \in]0, L_l]; \tag{4h}$$

$$\mathcal{G}_l = \{i | v_{l,a} \leq 0\} \qquad \forall l \in \mathcal{L}; \quad \forall a \in]0, L_l] \tag{4i}$$

$$y_{l,i} \in \mathcal{C} \qquad \forall l \in \mathcal{L}; \quad \forall i \in G_l \tag{4j}$$

where f_1 is the first objective function representing the cumulative fatigue damage, f_2 is the cost objective, x_l is the decision variables for the section lengths of mooring line l, y_l is the decision variables for the section constructions of mooring line l, α_l is the decision variables for the horizontal distance between the platform and mooring line l's anchor, and θ_l is the decision variable for the angle between the platform and mooring line l's anchor. $\mathcal{L}$, $\mathcal{S}$, $\mathcal{A}$ and $\mathcal{C}$ are the sets representing all the mooring lines, the sea states to examine, the available line constructions, and the line constructions which are chain respectively. The remaining variables in the above formulation are: s a sea state from the set of sea states, d the cumulative fatigue damage, $P(s)$ the probability of occurrence for sea state s, $c(y_{l,i})$ the unit cost of a mooring line construction, ϕ_l the initial orientation of mooring line l, $MBL_{l,a}$ the minimum breaking load at position a along line l, F_s the factor of safety on the mooring line tensions, a a position along the line, $\mathcal{G}_l$ the set of nodes along each mooring line which are in contact with the seabed, $v_{l,i}$ the minimum vertical distance between the seabed and node i along mooring line l during the simulation, and h is the water depth.

In this formulation f_1 and f_2 can be evaluated using any relevant model, be it the full dynamic simulations using OrcaFlex or the surrogate model detailed in section 3.2. In this way, either method takes the same input features (i.e. the genome) and provides the estimates of the cumulative fatigue damage and material cost (i.e. the output features).

3. Solution Approach

3.1 *Process Overview*

Optimization algorithms are methods which seek to identify the best possible solution from those available. To do this, they make use of a search algorithm to explore the possible decision variable values with respect to some objective functions (Burke and Kendall 2013). For real-world problems, it is often challenging to accurately formulate these evaluation functions such that the intra-relationships between the decision variables are captured in a time-efficient manner (Jin 2005, 2011). To overcome this, optimization of real-world problems can opt to replace the complex evaluation function with a simpler, less expensive approximate model: a surrogate model. For these surrogate models to be of use, they need to be able to capture the trends of the full evaluation function, so that on a relative basis, the results of the surrogate optimization problem can inform the original problem.

For the mooring optimization problem, the full time-domain simulations are run using OrcaFlex, an industry standard software package for the time domain analysis of offshore structures. This software package is capable of modelling the tension in mooring lines involving multiple members and materials, as well as the excursions of the moored body (Thomsen, Eskilsson, and Ferri 2017). Using these full time domain simulations, the surrogate model is built and trained, allowing proposed mooring systems during the optimization process to be assessed without the use of the full time-domain simulations.

The overall methodology is pictured in fig. 1 and makes use of both a multi-objective genetic algorithm, as well as the machine learning based surrogate model.

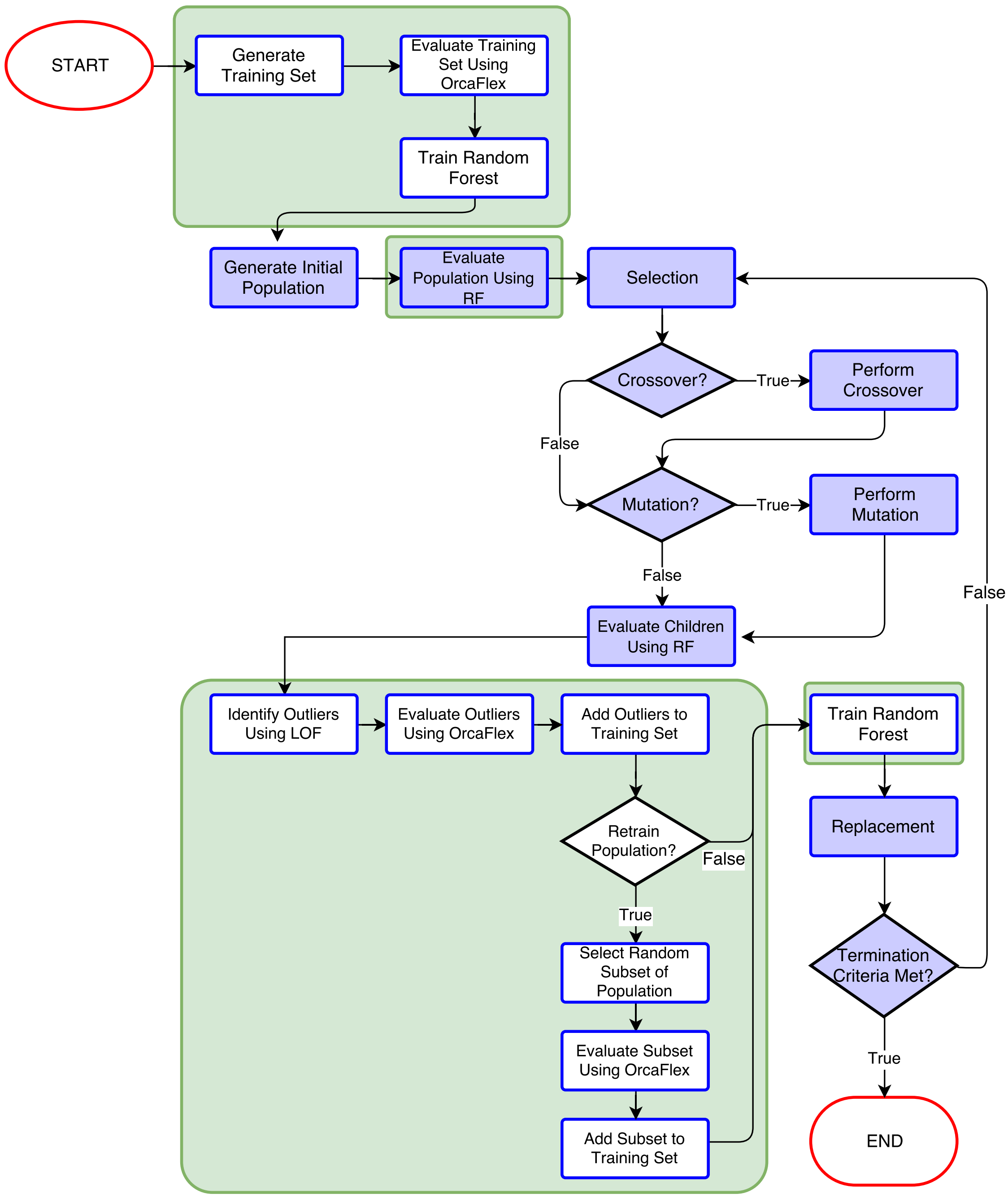
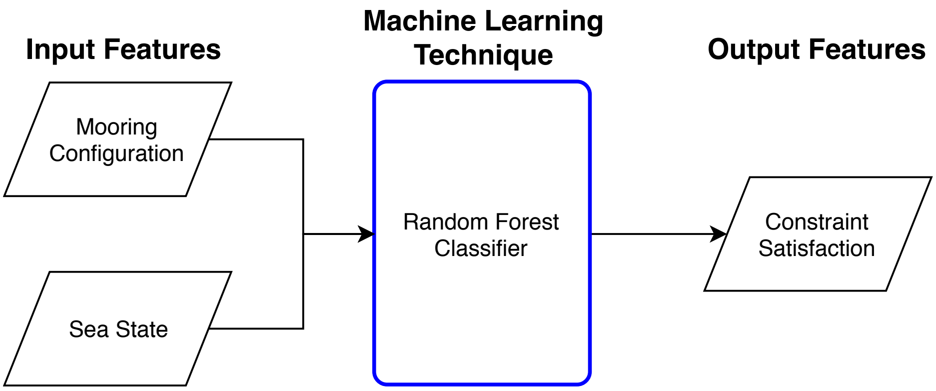
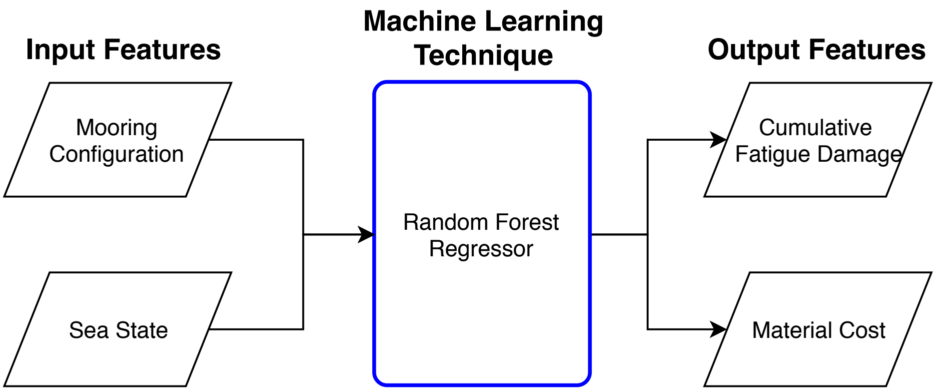


Figure 1.: Optimization process using a random forest surrogate model. The steps related to the surrogate model are highlighted in light green boxes, while the core steps of the genetic algorithm are shown in blue.

Machine learning techniques operate according to the principles illustrated in fig. 2 and are generally divided into *classification* and *regression* problems. In the case of a classification problem, the output feature represents the classes that the input elements are grouped into, while for a regression problem the output features represent the quantities



(a) Classifier



(b) Regressor

Figure 2.: Overview of machine learning estimators. Note that the number of input and output features are not necessarily related, though generally, there are fewer output features than there are input features. For the case of a classifier, the output features represent the classes to which each individual belongs while in the case of regression, the output features represent the values of interest.

of interest. Machine learning algorithms are often thought of as black boxes which seek to correlate the outputs features to the inputs features without simulating or modelling the underlying physics or engineering principles, but are purely statistical models. For any machine learning strategy, a training set, a set of inputs and outputs, is used to calibrate the black box model in order to build these statistical relationships. Machine learning techniques in general, therefore, work best with large training datasets from which the statistical correlations can be built. Furthermore, machine learning algorithms such as a neural network or random forest work best when they are *interpolating* between values on the training set rather than *extrapolating*. These algorithms therefore require that the training set cover the extent of the search space thereby allowing interpolation. Some machine learning algorithms such as random forests are capable of extrapolating output features, however, at a cost in accuracy.

In the present implementation, the input features to the machine learning technique are the decision variables of the optimization problem and the output features are the evaluated objective functions and the mooring system's satisfaction of the constraints. In this scheme, the surrogate model first estimates if the proposed solution will satisfy or violate the constraints, in the event that the model predicts that the constraints will be satisfied, the second phase of the surrogate estimates the objective function values. In effect, this surrogate model therefore, uses a classifier to determine the constraint satisfaction component of the problem and then a regression method to determine the objective function values. OrcaFlex is therefore only used when training and retraining the learning algorithm and is no longer directly tied to the evaluation functions for the optimization. The full deployed procedure is shown in fig. 1 with the creation of the surrogate model highlighted in green. This new methodology follows five basic steps:

(1) Build a training set of possible mooring systems;
(2) Evaluate the training set using the original full time domain simulation-based evaluation function;
(3) Use result from OrcaFlex model to train the surrogate model;
(4) Use surrogate model to perform optimization using NSGA-II;
(5) Retrain the surrogate as required.

A non-dominated sorting genetic algorithm II (NSGA-II) is used to optimize over multiple objective functions. This method and the full methodology deployed in this study are described in greater detail in section 3.3. Particular care has been taken to avoid premature convergence issues by accurately and consistently implementing both the crossover and mutation operators.

3.2 *Random Forest*

Random forests represent an ensemble learning method that can be used for either classification or regression. In either application, random forests work by constructing several decision trees each from a subset of the training set and its features (Breiman 2001).

A decision tree, is a basic machine learning technique in which inputs are entered and as the decision tree is traversed, the features are binned into smaller and smaller sets allowing an output to be determined based on the given input features. From a computational perspective, decision trees are generally implemented as binary trees. Where a single tree may have difficulty to accurately classify or predict an output for a complex set of inputs, the use of many trees (i.e. a forest rather than a single tree) can overcome this. The trees in a random forest each use a subset of the input features and the training set thereby reducing the biases that may result from using a single tree (James et al. 2013; Hastie, Tibshirani, and Friedman 2009). The procedure of a random forest is given in algorithm 1.

The decision variables of the present problem include a categorical variable representing the line type of the mooring line sections and continuous variables for the lengths of the mooring lines and the anchor position. The categorical variable ($y_{l,i}$) is handled in the surrogate model using one-hot encoding wherein the categorical variable is converted to a binary string in which only one bit can be a 1. Using this encoding, there is no assumption of natural ordering of the categories which improves performance.

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} f_b(x') \tag{5}$$

Once the forest is constructed, subsequent input data can be run through each of the decision trees. The outputs of all the trees are then averaged in order to determine the output of the forest (eq. (5)). In machine learning, an *ensemble* method is any method that uses multiple simpler machine learning techniques in its implementation. In this case, the random forest uses a series of decision trees thereby operating as an ensemble method (Olaya-Marín, Martínez-Capel, and Vezza 2013; Ahmad, Mourshed, and Yacine 2017; Bagnall et al. 2016). The initial mooring designs used to train the random forest

Algorithm 1 Random Forest

Require: a training set consisting of input features (x) and output features (z), $\mathcal{S} := (x_1, z_1), ..., (x_n, z_n)$, features $\mathcal{F}$, and number of trees in forest, B
  for $i = 1$ to B **do**
    draw a random sample $\mathcal{S}^*$ of size n with replacement from $\mathcal{S}$
    while minimum node size not reached **do**
      randomly select f features from $\mathcal{F}$
      select a split point among the f features
      split the node into two daughter nodes
    end while
    add constructed tree, T_i to forest, $\mathcal{A}$
  end for
  return $\mathcal{A}$

are generated using a Monte Carlo based sampling approach. In order to increase the accuracy of the surrogate in particular to the regions being explored during the optimization process further mooring designs are added to the training set and the surrogate is retrained periodically in what is known as the *growing set approach* (Kourakos and Mantoglou 2009).

Though artificial neural networks (ANNs) currently receive much attention in the research literature, there are many problem types where a random forest (RF) is better suited. Prior to building a model, however, it is often difficult to identify which machine learning approach is best suited to a problem a priori (Olaya-Marín, Martínez-Capel, and Vezza 2013). Extending the *'no free lunch theorem'* implies that though ANNs are effective for solving a particular problem does not demonstrate that they will efficiently solve all problems (Wolpert and Macready 1997; Wolpert 1995; Murphy 2012). For the present work, an RF has been deployed, as it is an effective technique for a wide range of problem types with relatively few tunable hyper parameters. This means, that from an implementation perspective, the RF is one of the easiest to set-up and get useful results from (Statnikov, Wang, and Aliferis 2008; Ahmad, Mourshed, and Yacine 2017). Though the RF has been deployed here, the modular nature of the method allows an alternate machine learning method to be implemented with minimal changes to the structure of the tool.

3.3 *Genetic Algorithm*

Genetic algorithms represent a family of biologically inspired population based metaheuristic optimization algorithms that borrow ideas from natural evolution as observed in biological systems (Holland 1992). Both genetic algorithms and evolutionary algorithms in general operate on biological analogies based on evolution. As these types of algorithms consider a set of potential solutions each iteration rather than a single solution, they are further classed as population-based. Evolutionary algorithms are commonly applied to a wide array of engineering optimization problems due to its generalized form which allows the same strategy to be applicable to a wide range of different problems. These algorithms are unable to guarantee that the true global optima is found, however, they generally converge to a high quality solution in an acceptable runtime (Burke and Kendall 2013; Rao 2009; Mitchell 1998). These algorithms are therefore only implemented when the size of the search space or the complexity in the objective space make it infeasible to deploy traditional optimization algorithms.

Classical optimization strategies are generally limited to continuous, differentiable objective functions. Due to their complexity, simulation based objective functions such as those relating to real-world engineering optimization problems, e.g. the mooring system optimization problem, are therefore better solved by heuristics and metaheuristic algorithms such as the genetic algorithm (Rao 2009). Figure 3 illustrates the relationship between the time complexity of an optimization problem and the selection of the correct solution approach. As indicated in this figure, as the complexity increases, heuristics and metaheuristics become the algorithms of choice as these allow solutions to be found within acceptable timescales without requiring full enumeration.

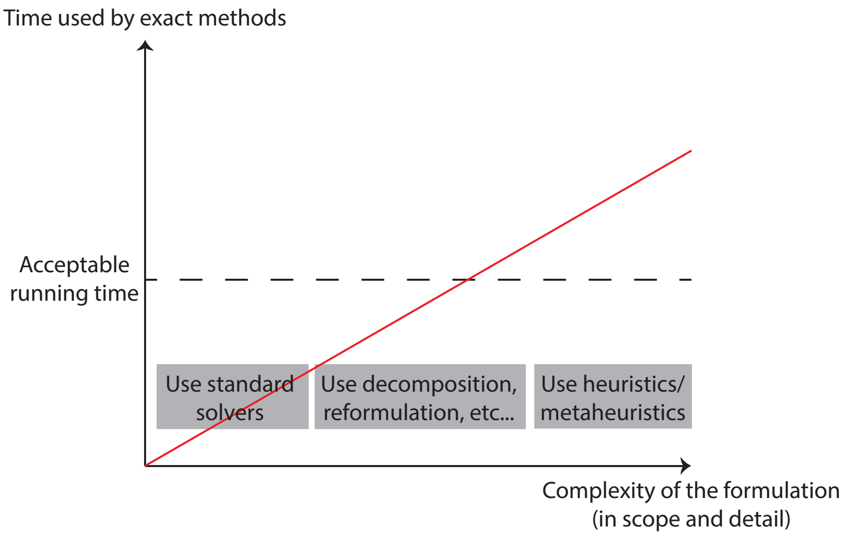


Figure 3.: Depending on the complexity of the model at hand and the time required to execute the optimization method, different algorithm types can be more appropriate to the problem.

In a GA, the candidate solutions within the population are formulated such that the combination of the decision variables are considered a *genome* which defines the individual solutions. In keeping with the evolutionary analogy, each solution is assigned a *fitness* by the evaluation functions with higher fitness values resulting in a higher probability of contributing genetic material towards new candidate solutions. Poor solutions, as judged by the evaluation functions, are therefore assigned lower fitness scores and therefore are less likely to have traits which are passed on to the next generation. The flowchart in fig. 1 shows the steps of a GA in blue. After selecting pairs of individuals among the population to reproduce (i.e. to generate new candidate solutions), the pair undergoes what is referred to as *crossover*. During crossover, the two *parent* solutions are combined in such a way that two new solutions are generated, each with approximately 50% of their genome being defined by each parent. In order to ensure that the GA does not prematurely converge to a local solution, a *mutation* operator is used to randomly alter the child solutions. This process is repeated until the solutions converge, or there is insufficient diversity within the remaining population for the process to continue effectively.

In the present implementation, a uniform crossover operator is deployed with a Gaussian mutation operator. Uniform crossover uses a fixed probability (50% in the present work) to determine which of the parents contributes a given gene to the child solutions. The Gaussian mutation operator uses a Gaussian distribution to alter a given gene *if* that gene is undergoing mutation (Beyer et al. 2002). Uniform crossover is selected as it ensures that the crossover process does not suffer from positional bias (Spears and Jong 1995). The Gaussian mutation operator is one of the simplest to implement, and is generally seen as a quick and effective means of applying mutation (Cazacu 2017). This combination of operators which are commonly deployed in tandem, work as an effective means of ensuring that all possible solutions within the solution space are obtainable during the optimization process regardless of the initialization or the convergence of the algorithm. This helps stave off premature convergence and aids in preserving diversity within the population.

In multi-objective optimization, the optimizer seeks to identify a set of solutions which highlight the trade-off between the competing objectives (Deb 2001). Most multi-objective optimization approaches combine the competing objectives in such a way that the problem can be treated as a single objective problem using traditional approaches, however, in doing so much of the problem complexity and nuance is often lost. True multi-objective optimization is not simply an extension of single-objective optimization, but requires additional considerations in order to simultaneously address the various competing objectives. In a true non-trivial multi-objective optimization problem with conflicting objectives, there is not a single solution which simultaneously optimizes all of the objectives, but a Pareto front which represents the trade-off between the competing objectives (see fig. 4). While an optimization algorithm applied to a single-objective optimization problem seeks to identify a single solution representing the global optima, a multi-objective optimization algorithm seeks instead to identify this Pareto front of potentially an infinite number of solutions. In the event that the objectives do not compete, but are rather complimentary, then a Pareto front will not be realized, as from the optimizer perspective, the problem reduces to a single objective problem.

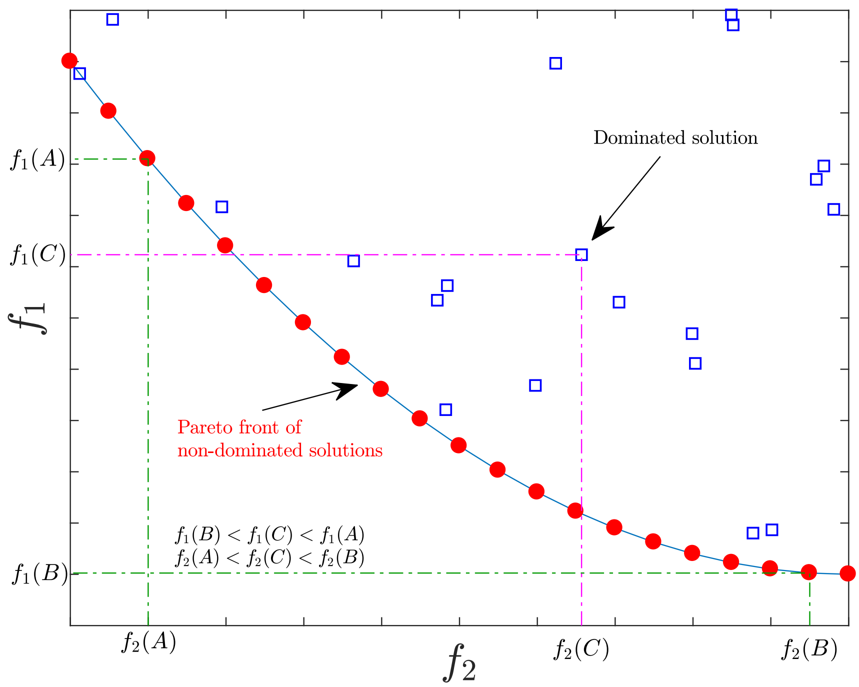


Figure 4.: Illustration of a Pareto front with dominated and non-dominated solutions for a case of two objectives both of which are to be minimized. The non-dominated solutions (red circles) are explicitly better in at least one objective and no worse in the others. For example in this figure solutions A and B lie on the Pareto front, while solution C is dominated by other solutions on the Pareto front and therefore not a member of the non-dominated set.

NSGA-II developed by Deb (2001); Deb and Pratap (2002) is a multi-objective genetic algorithm (MOGA) which uses a sorting algorithm to identify fronts of non-dominated solutions. NSGA-II is similar to the canonical GA, but differs by using a sorting algorithm to identify fronts of non-dominated solutions which is combined with a diversity preservation measure referred to as the *crowding distance*. The non-dominated fronts are ranked for use in a tournament selection in which the crowding distance is used as a tie breaker in the event that the two individuals in the tournament have the same non-dominated front (Deb 2001; Deb and Pratap 2002; Burke and Kendall 2013; Brownlee 2011). From here, standard crossover and mutation operations are used. The full NSGA-II methodology is well described in Deb and Pratap (2002) and Deb (2001). In the present implementation of NSGA-II, the parameters given in table 2 are used. In this implementation there are two crossover and mutation rates applied. The first set, those for the entire genome reflect the probability that the individual is subjected to crossover or mutation respectively while the second set, those for an individual gene (i.e. decision variable), reflect the probability, given that crossover or mutation occurs, that an individual decision variable is crossed-over or mutated.

Table 2.: Genetic Algorithm Parameters

Parameter	**Value**
Population Size	200
Number of Generations	50
Crossover Operator	Uniform
Mutation Operator	Gaussian
Probability of Crossover (Genome)	0.9
Probability of Crossover (Gene)	0.5
Probability of Mutation (Genome)	0.1
Probability of Mutation (Gene)	0.05
Elitism	Implicit to NSGA-II

The parameters used in the present implementation which are given in table 2 have been selected using a combination of recommendations from Grefenstette (1986); Deb and Pratap (2002) and from preliminary tuning of the algorithm. The current parameters are found to work well for the present problem, and as they are in line with general rules of thumb for GA parameters will likely be suitable for a wide range of problems, however, the parameters will be impacted by the specific problem at hand and should be tuned for the specific implementation of and problem instance.

3.4 *Anomaly Detection and Retraining the Surrogate Model*

In order to ensure that the surrogate model remains relevant to the region of the search space being explored by the optimizer, additional solutions are added to the training set (growing set approach) and the model is periodically retrained (Kourakos and Mantoglou 2009; Ong, Nair, and Keane 2003). Often, retraining of surrogates is done to augment the training set with solutions in the area of interest (i.e. near the Pareto front) in order to improve the quality of solutions in this region of the search space. Alternatively, however, retraining can be done to improve the surrogate's performance more evenly across the entire search space by using samples across the space when growing the training set. In the present work, increasing the size of the training set was done with two goals in mind: 1) increasing the surrogate's accuracy across the entire search space and 2) increasing the applicability of the surrogate by adding designs to the mooring system to ensure that the surrogate is always interpolating and not extrapolating.

Following each generation of the GA, the solutions estimated by the surrogate model are analysed using a local outlier factor (LOF) method which identifies potential outliers in a dataset based on a local density measure (Breunig et al. 2000; Chandola, Banerjee, and Kumar 2009). LOF is a proximity-based anomaly detection algorithm which operates by comparing the local deviation of a sample with respect to its neighbours (Breunig et al. 2000). LOF operates by comparing the distance between a sample and its nearest neighbours in order to establish a density, samples which have substantially lower densities than their neighbours are classed as outliers. In this case, the density is defined by a *local reachability density* (lrd) of a point. The reachability distance (d_r) and the lrd are given by eqs. (6) and (7) respectively.

$$d_r(p,o) = max[d_k(o), d(p,o)] \tag{6}$$

$$lrd(p) = \frac{\sum\limits_{o \in \mathcal{N}(p)} d_r(p,o)}{|\mathcal{N}(p)|} \tag{7}$$

These metrics are then combined to compute the LOF of a sample:

$$LOF(p) = \frac{\sum\limits_{o \in \mathcal{N}(p)} \dfrac{lrd(o)}{lrd(p)}}{|\mathcal{N}(p)|} \tag{8}$$

where $d_k(o)$ is the distance from o to its k-th nearest neighbour, $d(p,o)$ is the true distance between p and o, $\mathcal{N}(p)$ is the set of nearest neighbours to p, d_r represents the reachability distance. LOF values of approximately 1 indicate that a sample is comparable to its neighbours while values below 1 represent inliers, and values above 1 represent the outliers.

Individuals which are classed as potential outliers are added to the training set and the surrogate model is retrained. In this way, as the GA proceeds, the training set from which the surrogate model is built continues to grow and covers an increasing portion of the search space. This ensures that the surrogate model is interpolating rather than extrapolating thereby reducing potential errors. Though the surrogate will still struggle with outliers, and solutions surrounding the limits of the surrogate, the use of retraining should keep these

Furthermore, every five generations 10% of the population is selected at random for inclusion in the training set, ensuring that not only are the extent of the model improving through the inclusion of outliers, but the surrogate also improves across the entire search space. A random subset of the population rather than those closest to the Pareto front are selected as this ensures that the surrogate has an equal probability of improving throughout the search space rather than intensifying the search only in one particular region of the space potentially leading to premature convergence to a local solution.

Retraining the model in this way comes at increased computational expense as additional solutions must be assessed using OrcaFlex and the training itself must also be completed at regular intervals. A preliminary sensitivity study in the development stages of this methodology found that without the retraining, the final solutions were inferior unless a much larger initial training set was used. The net computational cost to achieve solutions of similar quality was therefore similar, however, using the retraining allowed the algorithm to adaptively select solutions to include in the training set thereby providing the maximum gain.

4. Case Study

4.1 *Case Description*

Continuing with the case study used for Pillai, Thies, and Johanning (2017, 2018b), the Offshore Code Comparison and Collaboration Continuation (OC4) semi-submersible designed for offshore wind turbines is modelled for deployment at Wave Hub. The OC4 semi-submersible is defined in Robertson, Jonkman, and Masciola (2014) and the hydrodynamic data is distributed as part of NREL's FAST software package. A schematic of the OC4 semi-submersible is shown in fig. 5. The conditions at Wave Hub are defined by long term measurements in Pitt, Saulter, and Smith (2006) and shown in table 4. Using extracts from the DTOcean Database, a range of chains and polyester ropes between 24 mm to 200 mm were provided to the OrcaFlex model and the optimizer (see table 3). These represent the materials and sizes likely to be deployed for offshore renewable energy applications (JRC Ocean 2016; Weller et al. 2014).

Table 3.: Available Line Types – Data from JRC Ocean (2016)

Material	**Diameter** [mm]	**MBL** [MN]	**Mass** [$\mathrm{kg\,m^{-1}}$]	**Axial Stiffness** [MN]	**Cost** [$£\mathrm{m^{-1}}$]
Chain	24	0.48	12.36	58.18	23.80
Chain	32	0.83	22.18	103.42	42.70
Chain	84	5.16	154.55	712.66	201.48
Chain	105	7.70	240.00	1113.53	312.89
Chain	152	14.43	480.00	2333.50	625.78
Chain	200	24.98	876.00	4040.00	920.11
Polyester	52	0.83	2.06	Variable	15.24
Polyester	104	3.07	7.30	Variable	54.02
Polyester	152	6.36	15.20	Variable	103.36
Polyester	192	10.10	24.08	Variable	156.52

Table 4.: Wave scatter table for Wave Hub site (Pitt, Saulter, and Smith 2006)

Sig. wave height, H_s [m]	**Wave Period,** T_z [s] 4	6	8	10	12
6.5	-	-	9	-	-
5.5	-	-	62	27	-
4.5	-	9	114	27	-
3.5	-	280	298	27	-
2.5	96	1253	298	27	-
1.5	1813	1945	298	35	9
0.5	1436	693	18	-	-

To demonstrate the capabilities of this optimization framework, relatively small training sets of 500 feasible mooring designs and approximately 2000 infeasible mooring de-

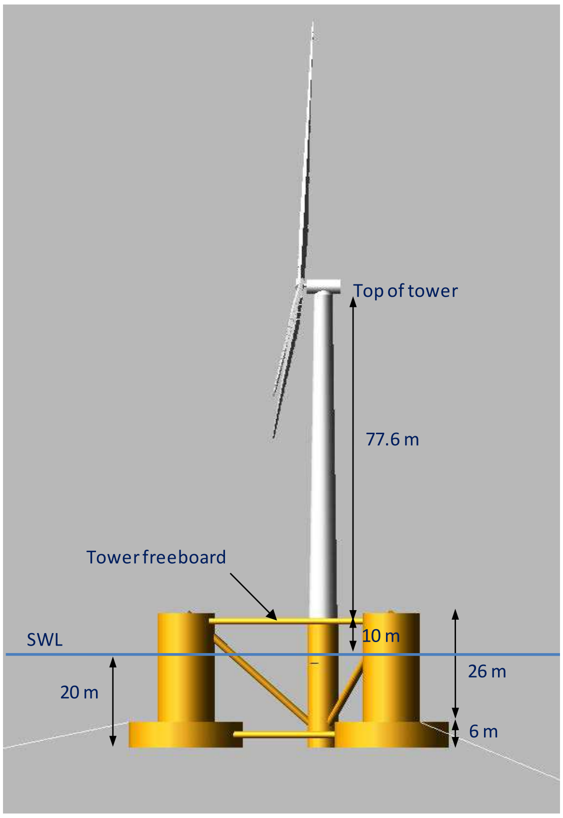


Figure 5.: DeepCwind floating wind system used as part of the Offshore Code Comparison Collaboration Continuation (OC4) project (Robertson, Jonkman, and Masciola 2014).

signs were used to train the classification and regression forests. Based on Oshiro, Perez, and Baranauskas (2012) the forests were designed to contain between 64 and 128 trees. A standard cross-validated grid search was deployed to determine the optimal number of trees in the forest on each occasion that the random forest was trained (Rao 2009; Müller and Guido 2016). In general, the greater the number of trees in the forest, the better the quality of the fit, however, this comes at an increase in the processing time required to construct the random forest estimator and to use the forest to estimate. Sensitivity studies into the number of trees in a random forest have found that for a range of problems, implementing beyond 128 trees offers diminishing returns Oshiro, Perez, and Baranauskas (2012).

4.2 *Results*

The final generation of feasible solutions from execution of the surrogate-model based multi-objective genetic algorithm are shown in fig. 6 with solutions of interest highlighted. These solutions of interest, the minimum cumulative fatigue damage, minimum cost, and a compromise solution are described in tables 5 to 7 respectively. Figure 7 explores the knee of this curve showing the solutions which simultaneously best minimize both solutions representing an equal priority between the two objectives.

Following 50 generations of the optimization, the surrogate models had classification ROC AUC of 0.862 and an outright accuracy of 0.998. The regression model had an R^2 of 0.915. These results indicate that through the use of this hybrid surrogate model for constraint satisfaction and for output feature values achieves high accuracy.

Though metrics such as the mean averaged error (MAE) and root mean squared error

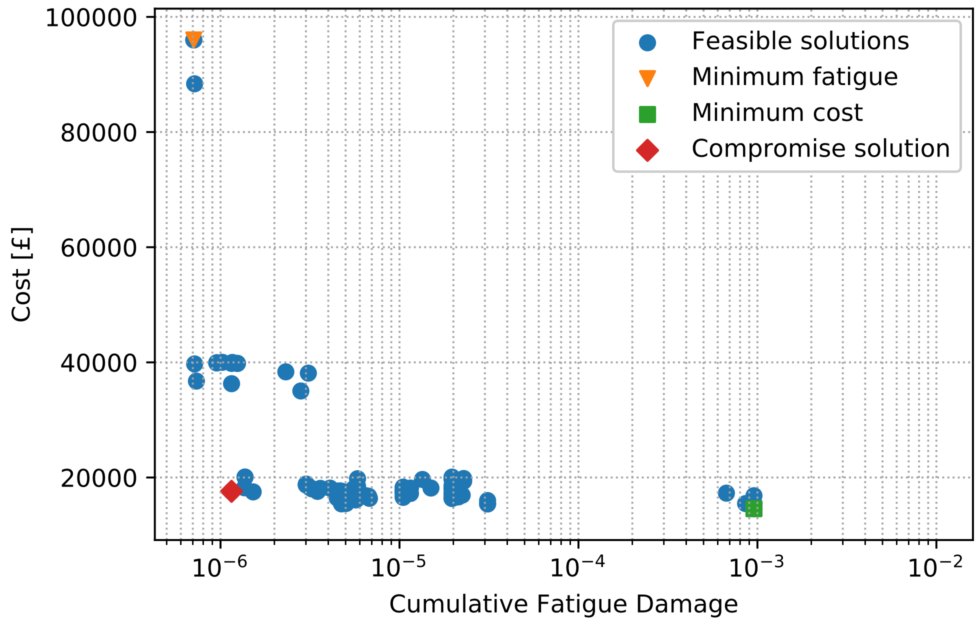


Figure 6.: Feasible solutions following final generation of optimization showing the trade-off between the mooring system cost and the cumulative fatigue damage; minimum cost and minimum fatigue solutions highlighted.

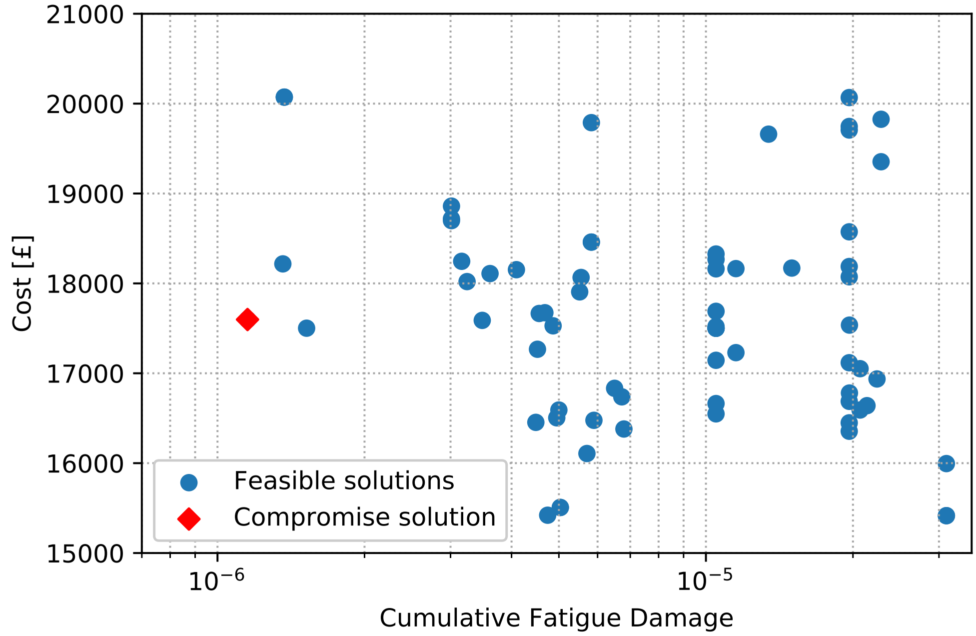


Figure 7.: Focus on solutions at the knee of the trade-off curve after the final generation of the optimization, highlighting the wide range of cost levels for any given fatigue.

528 (RMSE) are commonly used, we use the root mean square logarithmic error (RMSLE)
529 here. The RMSLE is given by eq. (9).

Table 5.: Numerical result - minimum fatigue damage

Line	**Anchor distance** [m]	**Anchor direction** [°]	**Line length** [m]	**Line type**
1	122	242	119	192 mm polyester
1			32	32 mm chain
2	379	10	340	32 mm chain
3	358	121	338	192 mm polyester
3			17	200 mm chain

Table 6.: Numerical result - minimum cost

Line	**Anchor distance** [m]	**Anchor direction** [°]	**Line length** [m]	**Line type**
1	120	239	13	152 mm polyester
1			159	24 mm chain
2	172	353	208	24 mm chain
2			25	32 mm chain
3	200	119	254	24 mm chain

Table 7.: Numerical result - knee

Line	**Anchor distance** [m]	**Anchor direction** [°]	**Line length** [m]	**Line type**
1	183	239	18	152 mm polyester
1			212	24 mm chain
2	172	358	236	24 mm chain
3	200	135	252	24 mm chain

$$RMSLE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[ln(h_i+1) - ln(\hat{h_i}+1)\right]^2} \tag{9}$$

where there n samples, h_i is the true value of sample i and $\hat{h_i}$ is the predicted value of sample i using the surrogate model. The RMSLE differs from the RMSE in that the RMSLE applies the natural logarithm to both the predicted and true values prior to computing the root mean square error. This is done to balance the impact of both big and small predictive errors. Especially given the different scales on which the output features operate, it was felt that using the MAE or RMSE would cause any errors in the cost estimate to dominate the error function and therefore give a biased measure of the error. The RMSLE avoids this and allows the error to convey greater meaning on the performance of the surrogate. Even in the event that all the output features are normalized to similar scales, the RMSLE still has the advantage over the MAE and RMSE in that it is not biased by the sizes of the error.

Table 8.: Surrogate Model RMSLE

Output Feature	**RMSLE**
Line 1 Cumulative Fatigue	0.60
Line 2 Cumulative Fatigue	2.91
Line 3 Cumulatiev Fatigue	1.26
Cost	1.88
Overall	1.87

4.3 *Comparison to Direct Optimization*

The surrogate assisted optimization methodology developed in this paper seeks to offer an improved means of optimizing the mooring designs of offshore renewable energy devices. In order to demonstrate the value of this approach, a comparison against direct optimization using NSGA-II has been completed.

The final Pareto front from executing the surrogate assisted optimization routine as described above is shown again against the results following 9 generations of direct optimization. Unfortunately, due to the increased computational complexity incurred when executing the direct optimization, it was not possible to execute the optimization for the same number of generations in a sensible time scale. From these results, it can be seen that in a fraction of the time (see table 9); the surrogate model can evaluate significantly more mooring systems, identifying a superior Pareto front. Furthermore, the best solutions with respect to the fatigue damage are an order of magnitude lower when using the surrogate assisted model as a result of the more complete optimization that can be achieved for a given computational effort. As the surrogate assisted solutions dominate the direct optimization results, with respect to aiding decision making, the surrogate assisted results will be of greater value.

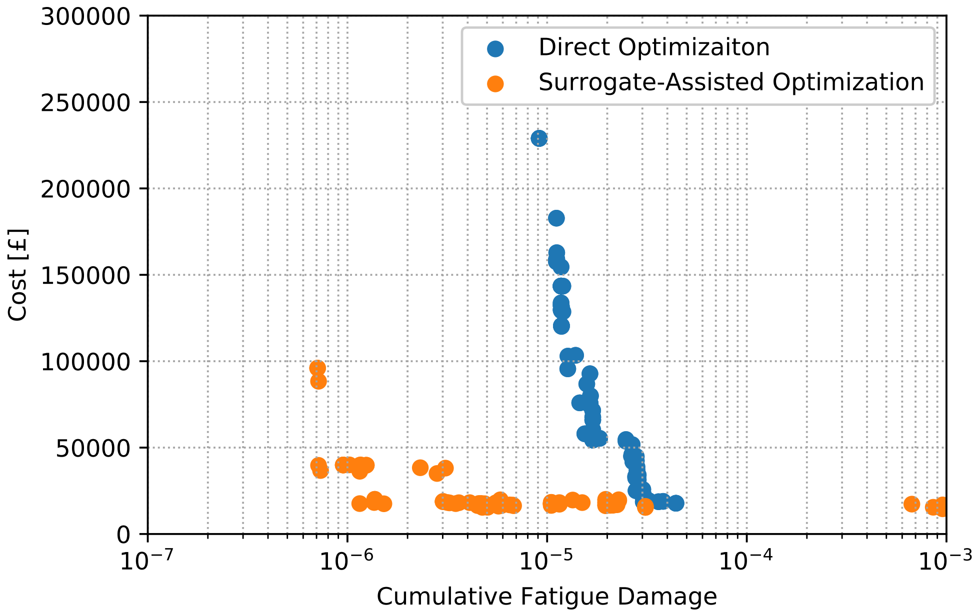


Figure 8.: Comparison of feasible solutions identified by direct optimization and surrogate assisted optimization routines.

Table 9.: Time Complexity of Surrogate Model

	Relative Time
Surrogate (No-Retraining)	1
Surrogate (With Retraining)	17
Direct Optimization	1938

5. Discussion

The presented work has detailed a new time efficient approach for the multi-objective optimization of mooring systems for renewable energy systems. This implementation of a trained random forest to replace the time-intensive time-domain simulations generally used in the design process reduces the average time required to evaluate a single mooring design (including time spent retraining the surrogate) from 692.2 s to 6.1 s running on an Intel Xeon E5440 rated at 2.83 GHz with 16 GB RAM representing a time reduction on the order of 114. This is a marked improvement over the traditional design approaches especially considering the high level of accuracy in both the classifier's ability to identify if solutions are compliant with respect to the constraints, and the regressor's ability to determine the cost and cumulative fatigue damage. In fact, without implementation of the surrogate assisted framework, a direct NSGA-II based optimization routine exceeds 30 h in evaluating and evolving each generation of solutions while the surrogate assisted framework requires on average approximately 15 min.

In fig. 6, the minimum cost solution and minimum fatigue solution are both highlighted. These solutions represent the extents of the Pareto front and can be thought of as the solutions to the single objective optimization problems along either of these objectives. From the shape of the curve it is apparent that the two objectives are indeed competing, however, there are a high density of solutions near the knee of the curve that may potentially represent a good compromise solution between the two extremes. In fact, though the minimum cost solution coincides with the maximum fatigue damage solution, there are many solutions of similar cost values at significantly lower fatigue levels.

Figure 7 highlights the solutions of the final population located at the knee of the Pareto front. This figure shows more solutions than just the Pareto front highlighting that there is a wide range of cost levels for a given fatigue level. This is important information for a decision maker as it indicates that the overall cost of the mooring system can be changed, however, if the high fatigue lines or components are not altered, it may not impact the overall cumulative fatigue damage.

The result described in table 5 minimizes the fatigue loading by increasing the length of the heavily loaded line, line 2, utilizing a long catenary chain thereby reducing the fatigue damage by reducing the tension experienced relative to the MBL. Furthermore, compared to the lower cost solutions, greater lengths of polyester are used throughout the mooring system and a much larger mooring footprint is required as a result of the longer catenary moorings.

Exploring the other extreme, the minimization of the system's material cost as shown in table 6 reduces the use of polyester lines in favour of chain constructions. Furthermore, the mooring lines are shorter, and anchors moved closer to the platform for a smaller footprint. Though this significantly reduces the cost, the fatigue levels are also significantly increased.

The 'compromise' solution detailed in table 7 represents an attempt at trying to balance the two objectives. In this case, the knee of the curve is targeted trying to find a solution which most equally balances the two objectives. This solution similar to the low cost solution, however, makes use mooring lines in order to reduce the fatigue with limited impact in cost. If the relevant mooring system designer had a different prioritization of the objectives, then an alternate design from the non-dominated front would prove to be more important, however, this is specific to the relative importance of the objectives to the mooring system designer.

Though the RF has been deployed to develop the present surrogate, the present framework can be used in future work to benchmark different machine learning algorithms for this specific application allowing the most suitable surrogate to be deployed.

6. Conclusion

The results presented indicate that for the present case study, the surrogate assisted optimization methodology is an effective means of mapping the design space and subsequently of optimizing the mooring system with a reduction of the time required on the order of 114 times. The surrogate model can in this case accurately estimate the features of interest to sufficient accuracy to provide useful information to the optimization process. The use of two separate models, one for the classification of solutions as feasible or infeasible had an outright accuracy of 0.998 indicating high reliability of the classifier. The use of both a classifier and a regression model ensures that the regression is only done for valid solutions, and the deployment of an anomaly detection algorithm helps in the identification of outliers which should be added to the training set to improve the performance of the surrogate. This works to orient the surrogate so that it has a relevant scope for interpolation and is not forced to extrapolate predictions which has helped the regression model achieve an RMSLE across all output features of 1.87.

The multi-objective approach implemented here does not identify a single optima for the given problem, but aids in decision making by presenting the trade-off between competing objectives. The results from using this methodology must then be assessed by a decision maker in order to determine where along the proposed Pareto front they wish to operate. The case study presented therefore only presents a series of solutions which from an optimization perspective are of equal value.

Though a large training set is used and significant time is required to generate this training set, once this information is compiled for a given device and site, the optimization process simply augments to this. As a result, though there could be further improvements with regards to the time efficiency of the overall procedure, the present methodology does demonstrate how a random forest based surrogate model could be integrated with a genetic algorithm in order to aid in the design and optimization of mooring systems for floating offshore renewable energy devices.

Future work using this framework can directly aid in the design of mooring systems for prototype devices considering deployment at test facilities such as FaBTest, WaveHub, or EMEC. Furthermore it can be used to explore the impact of novel mooring line materials which have been designed for offshore renewable energy applications. It should also be noted, that the results presented here represent the outputs from a single run in order to establish the capabilities and applicability of the developed methodology. Given the reduction in computational time through the deployment of this methodology it is reasonable to expect that when utilizing this methodology for real design problems multiple runs or a larger population size are used in order to avoid any seeding bias of both the GA and the surrogate's training set.

Nomenclature

D_c Cumulative fatigue damage

D_c	Cumulative fatigue damage
D_t	Fatigue damage during time t
F_s	Factor of safety
K	Material fatigue parameter derived from S-N or T-N curve
LOF	Local outlier factor
$MBL_{l,a}$	Minimum breaking load at position a along mooring line l
$N(S)$	Number of stress cycles
$P(s)$	Probability of occurrence of sea state s
S	Stress amplitudes established in the rainflow cycle count
T	Expected operational lifetime of the mooring system
α_l	The decision variables for the horizontal distance between the platform and the anchor attached to mooring line l
β	Material fatigue parameter derived from S-N or T-N curve
$\hat{h}_i$	Estimated value of sample i
$\mathcal{A}$	The set of available line constructions
$\mathcal{C}$	The set of available chains (a subset of $\mathcal{A}$)
$\mathcal{G}_l$	The set of nodes along mooring line l that are in contact with the seabed during the dynamic simulation
$\mathcal{L}$	The set of mooring lines
$\mathcal{N}(p)$	Set of nearest neighbours to p
$\mathcal{S}$	The set of sea states
ϕ_l	Initial heading of mooring line l
τ_d	Simulation duration
θ_l	The decision variables representing the angle between the platform and the anchor attached to mooring line l
$c(y_{l,i})$	Unit cost of a mooring line construction
d	True distance between two points
d_k	Distance to k-th nearest neighbour
d_r	Reachability distance
f_1	Cumulative fatigue damage objective function
f_2	Material cost objective function
h_i	True value of sample i
lrd	Local reachability distance
n	Number of samples
s	A specific sea state in set $\mathcal{S}$
$v_{l,a}$	The minimum vertical distance between position a along mooring line l and the seabed
x_l	The decision variables relating to the section lengths in mooring line l
y_l	The decision variables relating to the material of each section in mooring line l
z	Target output features

Acknowledgements

This work is funded by the EPSRC (UK) grant for the SuperGen Marine United Kingdom Centre for Marine Energy Research (UKCMER) [grant number: EP/P008682/1]. The authors would also like to thank Jason Jonkman at NREL who provided the hydrodynamic data for the OC4 semi-submersible and Orcina Ltd. for providing OrcaFlex.

References

Ahmad, Muhammad Waseem, Monjur Mourshed, and Rezgui Yacine. 2017. "Trees vs Neurons: Comparison between Random Forest and ANN for high-resolution prediction of building energy consumption Trees vs Neurons: Comparison between Random Forest and ANN for high-resolution prediction of building energy consumption." *Energy & Buildings* 147: 77–89. http://dx.doi.org/doi:10.1016/j.enbuild.2017.04.038{\%}0Ahttp://dx.doi.org/10.1016/j.enbuild.2017.04.038.

Amzallag, C., J. P. Gerey, J. L. Robert, and J. Bahuaud. 1994. "Standardization of the rainflow counting method for fatigue analysis." *International Journal of Fatigue* 16 (4): 287–293. arXiv:1011.1669v3.

Bagnall, Anthony, Jason Lines, Aaron Bostrom, and James Large. 2016. "The Great Time Series Classification Bake Off: An Experimental Evaluation of Recently Proposed Algorithms. Extended Version." *eprint arXiv:1602.01711* 19. 1602.0171. http://arxiv.org/abs/1602.0171.

Beyer, Hans-Georg, Hans-Georg Beyer, Hans-Paul Schwefel, and Hans-Paul Schwefel. 2002. "Evolution strategies A comprehensive introduction." *Natural Computing* 1 (1): 3 – 52. http://www.springerlink.com/content/2311qapbrwgrcyey{\%}5Cnhttp://link.springer.com/10.1023/A:1015059928466.

Breiman, Leo. 2001. "Random Forests." *Machine Learning* 45 (1): 5–32. arXiv:1011.1669v3.

Breunig, Markus M., Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. 2000. "LOF: Identifying Density-Based Local Outliers." *Proceedings of the 2000 Acm Sigmod International Conference on Management of Data* 1–12. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.35.8948.

Brown, D. T., and S. Mavrakos. 1999. "Comparative study on mooring line dynamic loading." *Marine Structures* 12 (3): 131–151.

Brownlee, Jason. 2011. *Clever Algorithms - Nature-Inspired Programming Recipes*. http://www.cleveralgorithms.com.

Burke, Edmund K, and Graham Kendall. 2013. *Search Methodologies*. 2nd ed. Boston, MA: Springer US.

Carbono, Alonso J Juvinao, Ivan F M Menezes, and Luiz Fernando Martha. 2005. "Mooring Pattern Optimization using Genetic Algorithms." *6th World Congress of Structural and Multidisciplinary Optimization, Rio de Janeiro, Brazil* (June): 1–9.

Cazacu, Razvan. 2017. "Comparative Study between the Improved Implementation of 3 Classic Mutation Operators for Genetic Algorithms." *Procedia Engineering* 181: 634–640.

Chandola, Varun, Arindam Banerjee, and Vipin Kumar. 2009. "Anomaly detection." *ACM Computing Surveys* 41 (3): 1–58. arXiv:1011.1669v3. http://portal.acm.org/citation.cfm?doid=1541880.1541882.

Chugh, Tinkle, Yaochu Jin, Kaisa Miettinen, Jussi Hakanen, and Karthik Sindhya. 2018. "A Surrogate-Assisted Reference Vector Guided Evolutionary Algorithm for Computationally Expensive Many-Objective Optimization." *IEEE Transactions on Evolutionary Computation* 22 (1): 129–142.

da Fonesca Monteiro, Bruno, Aline Aparecida de Pina, Juliana Souza Baioco, Carl Horst Albrecht, Beatriz Souza Leite Pires de Lima, and Breno Pinheiro Jacob. 2016. "Towards a methodology for the optimal design of mooring systems for floating offshore platforms using evolutionary algorithms." *Marine Systems & Ocean Technology* 11 (156): 55–67.

Davidson, Josh, and John V Ringwood. 2017. "Mathematical Modelling of Mooring Systems for Wave Energy ConvertersA Review." *Energies* 10 (5): 666. http://www.mdpi.com/1996-1073/10/5/666.

de Pina, Aline Aparecida, Bruno da Fonseca Monteiro, Carl Horst Albrecht, Beatriz Souza Leite Pires de Lima, and Breno Pinheiro Jacob. 2016. "Artificial Neural Networks for the analysis of spread-mooring configurations for floating production systems." *Applied Ocean Research* 59: 254–264. http://dx.doi.org/10.1016/j.apor.2016.06.010.

de Pina, Aloísio Carlos, Aline Aparecida de Pina, Carl Horst Albrecht, Beatriz Souza Leite Pires de Lima, and Breno Pinheiro Jacob. 2013. "ANN-based surrogate models for the analysis of mooring lines and risers." *Applied Ocean Research* 41: 76–86.

Deb, Kalyanmoy. 2001. *Multi-Objective Optimization using Evolutionary Algorithms.* Chichester: Wiley & Sons.

Deb, Kalyanmoy, and Amrit Pratap. 2002. “A fast and elitist multiobjective genetic algorithm: NSGA-II.” *IEEE Transactions on Evolutionary Computation* 6 (2): 182–197. http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=996017.

Forrester, Alexander, András Sóbester, and Andy Keane. 2008. *Engineering Design via Surrogate Modelling.* 1st ed. Chichester: John Wiley & Sons, Ltd.

Forrester, Alexander I.J., and Andy J. Keane. 2009. “Recent advances in surrogate-based optimization.” *Progress in Aerospace Sciences* 45 (1-3): 50–79. 1106.2697.

Forrester, Alexander I J, András Sóbester, and Andy J Keane. 2007. “Multi-Fidelity Optimization via Surrogate Modelling.” *Proceedings: Mathematical, Physical and Engineering Sciences* 463 (2088): pp. 3251–3269. http://www.jstor.org/stable/20209374.

Grefenstette, John J. 1986. “Optimization of Control Parameters for Genetic Algorithms.” *IEEE Transactions on Systems, Man, and Cybernetics* SMC-16 (February): 122–128. http://scholar.google.com/scholar?hl=en{\&}btnG=Search{\&}q=intitle:Genetic+Algorithms{\#}2.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning.* 2nd ed. New York, NY: Spring Science+Business Media.

Holland, John H. 1992. “Genetic Algorithms.” *Scientific American* July: 66–72.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An Introduction to Statistical Learning.* Vol. 103. arXiv:1011.1669v3. http://link.springer.com/10.1007/978-1-4614-7138-7http://linkinghub.elsevier.com/retrieve/pii/S0166531607000570{\%}0Ahttp://link.springer.com/10.1007/978-1-4614-7138-7.

Jin, Y. 2005. “A comprehensive survey of fitness approximation in evolutionary computation.” *Soft Computing* 9 (1): 3–12. http://link.springer.com/10.1007/s00500-003-0328-5.

Jin, Yaochu. 2011. “Surrogate-assisted evolutionary computation: Recent advances and future challenges.” *Swarm and Evolutionary Computation* 1 (2): 61–70. http://dx.doi.org/10.1016/j.swevo.2011.05.001.

Johanning, L, G Smith, and J Wolfram. 2006. “Mooring design approach for wave energy converters.” *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment* 220 (4): 159–174. http://dx.doi.org/10.1243/14750902JEME54.

JRC Ocean. 2016. “DT Ocean Suite - DTOcean Database 1.0.0.” https://setis.ec.europa.eu/dt-ocean/download/file/fid/82.

Kourakos, George, and Aristotelis Mantoglou. 2009. “Pumping optimization of coastal aquifers based on evolutionary algorithms and surrogate modular neural network models.” *Advances in Water Resources* 32 (4): 507–521. http://dx.doi.org/10.1016/j.advwatres.2009.01.001.

Kwan, C T, and F J Bruen. 1991. “Mooring Line Dynamics: Comparison of Time Domain, Frequency Domain, and Quasi-Static Analyses.” *Proceedings of the 23rd Annual Offshore Technology Conference Houstoun, Texas* .

Mitchell, Melanie. 1998. *An Introduction to Genetic Algorithms.* 1st ed. Cambridge, Mass.: The MIT Press. http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20{\&}path=ASIN/0262631857.

Müller, Andreas C., and Sarah Guido. 2016. *Introduction to Machine Learning with Python.* Sebastopol: O'Reilly Media.

Murphy, Kevin P. 2012. *Machine Learning: A Probabilistic Perspective.* Cambridge, Massachussetts: The MIT Press. http://link.springer.com/chapter/10.1007/978-94-011-3532-0{_}2.

Olaya-Marín, E.J., F. Martínez-Capel, and P. Vezza. 2013. “A comparison of artificial neural networks and random forests to predict native fish species richness in Mediterranean rivers.” *Knowledge and Management of Aquatic Ecosystems* (409): 07. http://www.kmae-journal.org/10.1051/kmae/2013052.

Ong, Yew S., Prasanth B. Nair, and Andrew J. Keane. 2003. “Evolutionary Optimization of Computationally Expensive Problems via Surrogate Modeling.” *AIAA Journal* 41 (4).

Oshiro, Thais Mayumi, Pedro Santoro Perez, and Jose Augusto Baranauskas. 2012. “How Many Trees in a Random Forest?.” *International Workshop on Machine Learning and Data*

Mining in Pattern Recognition (July): 154–168. http://www.springerlink.com/content/mm0aegv576cf8wqe/.

Pillai, Ajit C., Philipp R. Thies, and Lars Johanning. 2017. "Multi-Objective Optimization of Mooring Systems for Offshore Renewable Energy." In *Proceedings of the 12th European Wave and Tidal Energy Conference, Cork, Ireland,* .

Pillai, Ajit C., Philipp R. Thies, and Lars Johanning. 2018a. "Comparing Frequency and Time Domain Simulations for Geometry Optimization of a Floating Offshore Wind Turbine Mooring System." *Proceedings of the ASME 2018 International Offshre Wind Technical Conference (IOWTC2018) San Francisco, USA* 1.

Pillai, Ajit C., Philipp R. Thies, and Lars Johanning. 2018b. "Development of a Multi-Objective Genetic Algorithm for the Design of Offshore Renewable Energy Systems." In *Advances in Structural and Multidisciplinary Optimization (WCSMO12),* edited by Axel Schumacher, Thomas Vietor, Sierk Fiebig, Kai-Uwe Bletzinger, and Kurt Maute. 2013–2026. Cham: Springer International Publishing.

Pitt, E.G., A. Saulter, and H. Smith. 2006. *The wave power climate at the Wave Hub site*. Tech. Rep. November. Applied Wave Research report to SWRDA.

Rao, Singiresu S. 2009. *Engineering Optimization: Theory and Practice*. 4th ed. Hoboken, New Jersey: John Wiley & Sons.

Robertson, A., J. Jonkman, and M. Masciola. 2014. "Definition of the Semisubmersible Floating System for Phase II of OC4." *Golden, CO* (September): 38. http://www.nrel.gov/docs/fy14osti/60601.pdf.

Rychlik, I. 1987. "A new definition of the rainflow cycle counting method." *International Journal of Fatigue* 9 (2): 1190121.

Ryu, Sam, Arun S. Duggal, Caspar N. Heyl, and Zong Woo Geem. 2016. "Cost-Optimized FPSO Mooring Design Via Harmony Search." *Journal of Offshore Mechanics and Arctic Engineering* 138 (6): 061303. http://offshoremechanics.asmedigitalcollection.asme.org/article.aspx?doi=10.1115/1.4034374.

Ryu, Sam, Caspar N. Heyl, Arun S. Duggal, and Zong Woo Geem. 2007. "Mooring Cost Optimization via Harmony Search." *26th International Conference on Offshore Mechanics and Arctic Engineering* 1–8.

Schijve, Jaap. 2009. *Fatigue of Structures and Materials*. 2nd ed. Delft: Springer Science+Business Media. http://link.springer.com/content/pdf/10.1007/0-306-48396-3.pdfhttp://www.springerlink.com/index/10.1007/0-306-48396-3.

Shafieefar, Mehdi, and Aidin Rezvani. 2007. "Mooring optimization of floating platforms using a genetic algorithm." *Ocean Engineering* 34 (10): 1413–1421.

Shankar Bhattacharjee, Kalyan, Hemant Kumar Singh, and Tapabrata Ray. 2016. "Multi-Objective Optimization With Multiple Spatially Distributed Surrogates." *Journal of Mechanical Design* 138 (9): 091401. http://mechanicaldesign.asmedigitalcollection.asme.org/article.aspx?doi=10.1115/1.4034035.

Sidarta, Djoni E., Johyun Kyoung, Jim O'Sullivan, and Kostas F. Lambrakos. 2017. "Prediction of offshore platform mooring line tensions using artificial neural network." *Proceedings of the ASME 2017 36th International Conference on Ocean, Offshore, and Arctic Engineering (OMAE 2017) June 25-30 Trondhei, Norway* 1–11.

Spears, William M, and Kenneth A De Jong. 1995. *On the Virtues of Parameterized Uniform Crossover*. Tech. rep.. Washington DC: Naval Research Lab.

Statnikov, Alexander, Lily Wang, and Constantin F. Aliferis. 2008. "A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification." *BMC Bioinformatics* 9: 1–10.

Thies, Philipp R., Lars Johanning, Violette Harnois, Helen C M Smith, and David N. Parish. 2014. "Mooring line fatigue damage evaluation for floating marine energy converters: Field measurements and prediction." *Renewable Energy* 63: 133–144. http://dx.doi.org/10.1016/j.renene.2013.08.050.

Thomsen, Jonas, Francesco Ferri, Jens Kofoed, and Kevin Black. 2018. "Cost Optimization of Mooring Solutions for Large Floating Wave Energy Converters." *Energies* 11 (1): 159. http://www.mdpi.com/1996-1073/11/1/159.

Thomsen, Jonas Bjerg, Claes Eskilsson, and Francesco Ferri. 2017. *Assessment of Available Nu-*

merical Tools for Dynamic Mooring Analysis. Tech. rep.. Aalborg University.

Voutchkov, I, and A J Keane. 2006. "Multi-objective optimization using surrogates." *Engineering Sciences* 14 (2): 155–175. http://eprints.soton.ac.uk/37984/.

Weller, Sam, Jon Hardwick, Lars Johanning, Madjid Karimirad, Boris Teillant, Alex Raventos, Stephen Banfield, et al. 2014. "A comprehensive assessment of the applicability of available and proposed offshore mooring and foundation technologies and design tools for array applications." 68. http://www.dtocean.eu/Deliverables-Documentation/Deliverable-4.1.

Weller, Sam D, Philipp R Thies, Tessa Gordelier, and Lars Johanning. 2015. "Reducing Reliability Uncertainties for Marine Renewable Energy." *Journal of Marine Science and Engineering* 3: 1349–1361.

Wolpert, Dh. 1995. "No free lunch theorems for search." *Technical Report SFI-TR-95-02-010* 1–38. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.47.7505{\&}rep=rep1{\&}type=pdf.

Wolpert, D H, and W G Macready. 1997. "No Free Lunch Theorems for Optimisation." *IEEE Trans. on Evolutionary Computation* 1 (1): 67–82.

Won, Kok Sung, and Tapabrata Ray. 2005. "A framework for design optimization using surrogates." *Engineering Optimization* 37 (7): 685–703.