An Active Learning Strategy for Joint Surrogate Models Construction with Compatibility Conditions: Application to VPP

Malo Pocheau, Olivier Le Maître, Renaud Bañuls

Journal of Sailing Technology 2023, volume 8, issue 1, pp. 76 – 95.
The Society of Naval Architects and Marine Engineers.

An Active Learning Strategy for Joint Surrogate Models Construction with Compatibility Conditions: Application to VPP

Malo Pocheau
Bañulsdesign, Centre de Mathématiques Appliquées, École Polytechnique, France,
malo@banulsdesign.com.

Olivier Le Maître
CNRS, Centre de Mathématiques Appliquées, École Polytechnique, France.

Renaud Bañuls
Bañulsdesign, France.

Manuscript received June 23, 2023; revision received July 17, 2023; accepted July 18, 2023.

Abstract. Static Velocity Prediction Programs (VPP) are standard tools in sailing yachts' design and performance assessment. Predicting the maximal steady velocity of a yacht involves resolving constrained optimization problems. These problems have a prohibitive computational cost when using high-fidelity global modeling of the yacht. This difficulty has motivated the introduction of modular approaches, decomposing the global model into subsystems modeled independently and approximated by surrogate models (response surfaces). The maximum boat speed for prescribed conditions solves an optimization problem for the trimming parameters of the model constrained by compatibility conditions between the subsystems' surrogate solution (e.g., the yacht equilibrium). The accuracy of the surrogates is then critical for the quality of the resulting VPP. This paper relies on Gaussian Process (GP) models of the subsystems and introduces an original sequential Active Learning Method (ALM) for their joint construction. Our ALM exploits the probabilistic nature of the GP models to decide the enrichment of the training sets using an infilling criterion that combines the predictive uncertainty of the surrogate models and the likelihood of equilibrium at every input point. The resulting strategy enables the concentration of the computational effort around the manifolds where equilibrium is satisfied. The results presented compare ALM with a standard (uninformed) Quasi-Monte Carlo method, which samples the input space of the subsystems uniformly. ALM surrogates have higher accuracy in the equilibrium regions for equal construction cost, with improved mean prediction and reduced prediction uncertainty. We further investigate the effect of the prediction uncertainty on the numerical VPP and in a routing problem.

Keywords: Surrogate Model; Gaussian Process; VPP; Routing; Uncertainty Quantification

NOMENCLATURE

A_{w}	Area of flotation plane [m^2]
B_{wl}	Beam at the waterline [m]
C	Co-variance function of a GP [-]
C_{mid}	Midship coefficient [-]
C_{p}	Prismatic coefficient [-]
D	Displacement [kg]
f	Flat parameter of aerodynamic model [-]
g	Acceleration of gravity [$\mathrm{m.s}^{-2}$]

G_{M}	Initial meta-centric height [m]
$\mathcal{GP}_{(a)}$	GP of aerodynamic model $M_{(a)}$ [-]
$\mathcal{GP}_{(i)}$	GP model of model $M_{(i)}$ [-]
$\mathcal{GP}_{(h)}$	GP of hydrodynamical model $M_{(h)}$ [-]
H	Vector of hyperparameters [-]
IC	Infilling criterion [-]
$\mathcal{L}$	Likelihood function [-]
L_{cb}	Longitudinal center of buoyancy (from forward perpendicular) [m]
L_{cf}	Longitudinal center of flotation (from forward perpendicular) [m]
L_{wl}	Length at the waterline [m]
M	Model of system S [-]
$M_{(a)}$	Model of aerodynamic subsystem [-]
$M_{(i)}$	Model of subsystem $S_{(i)}$ [-]
$M_{(h)}$	Model of hydrodynamic subsystem [-]
m	Number of subsystems [-]
n_o	Number of observations in GP construction [-]
p	Probability [-]
S	Parametrised system [-]
$S_{(i)}$	Parametrised subsystem (i) [-]
S_{dagg}	Daggerboard surface [m^2]
S_{sail}	Sail area [m^2]
T_{c}	Draft of canoe body [m]
T_{t}	Total draft [m]
$\mathrm{Tr}(\,)$	Trace of a matrix [-]
V	Boat speed [$\mathrm{m.s}^{-1}$]
V_a	Apparent wind speed [kt]
V_b	Boat speed expressed [kt]
W_{SA}	Wetted surface area [m^2]
$x_{(i)}$	Input of subsystem (i) [-]
X_o	Vector of observation points [-]
$x_{(i)}^{+}$	New observation point for $M_{(i)}$ [-]
X_*	Vector of prediction points [-]
Y_o	Vector of model predictions at X_o [-]
Y_*	GP predictions at X_* [-]
Z	Vector of compatibility constraints [-]
$\tilde{Z}$	Approximation of Z from the subsystem of GPs [-]
Z_{a}	Aerodynamic vertical center of effort [m]
Z_{dagg}	Daggerboard vertical center of effort [m]
β_a	True wind angle [°]
β_t	True wind speed [kt]
γ_a	Apparent wind angle [°]

θ	Heel angle [°]
λ	Leeway angle [°]
Λ^{dagg}	Daggerboard aspect ratio [-]
$\mu_{(i)}$	Mean value of $\mathcal{GP}_{(i)}$ [-]
ν_w	Viscosity of sea water [$\mathrm{m^2 s^{-1}}$]
ρ_a	Density of air [$\mathrm{kg\,m^{-3}}$]
ρ_w	Density of sea water [$\mathrm{kg\,m^{-3}}$]
$\Sigma^2_{(i)}$	Prediction co-variance of $\mathcal{GP}_{(i)}$ [-]
Ω	Input space of M [-]
$\Omega_{(i)}$	Input space of $M_{(i)}$ [-]

ALM	Adaptative Learning Method
CFD	Computational Fluid Dynamics
ETA	Estimated Time of Arrival
GFS	Global Forecast System
GMT	Greenwhich Mean Time
GP	Gaussian Process
LHS	Latin Hypercube Sampling
QMC	Quasi Monte Carlo
QTVLM	Routing software QtVlm
VPP	Velocity Prediction Program

1. INTRODUCTION

Velocity Prediction Programs (VPP), introduced by Davidson (1936) and again by Kerwin (1975) are widely used in modern naval architecture to predict the performance of a yacht, as attested by their widespread use in the published literature, see Horel (2022), and are currently still a topic of research as shown by the recent developments presented in Melis *et al*. (2022), and Reche-Vilanova *et al.* (2021). Classically, the VPP provides the maximum boat speed for some weather conditions (wind and possibly sea states) by resolving an optimization problem for several "free" parameters defined by the designer, such as sails and appendage trimming. The prediction cost of the VPP can vary greatly depending on the nature and fidelity of its underlying yacht model. Ideally, one should incorporate detailed CFD simulations, see Lindstand *et al*. (2017) and Persson *et al.* (2021), with large displacement elastic analysis for the sails and appendages. However, this level of modeling remains computationally too expensive to be applied at the whole yacht scale, embedded in optimization loops, and queried for multiple conditions.

To circumvent these limitations, VPP models typically divide the yacht model into subsystems (the sails, the hulls, the appendages, …), neglecting their interactions. The subsystems can then be simulated independently, possibly in parallel. Subsequently, one combines the subsystems to check that the whole system is in equilibrium (static VPP) and, if needed, adapt the "free" parameters of the subsystems to reach equilibrium. In practice, ensuring the equilibrium demands many iterations, and solving the subsystems many times induces a high computational cost. This situation has motivated the introduction of surrogate models for the subsystems. Classically, these surrogates interpolate the responses of the subsystems evaluated at a grid of input values, resulting in the often-called response surface models.

The present paper tackles the problem of building surrogate models of the yacht's subsystems most efficiently concerning the numerical cost and accuracy of the model's prediction. The main contribution of the present work is the definition of a joint sequential surrogates construction strategy motivated by the fact that the models of the subsystems need be accurate only for input values that satisfy some compatibility conditions, namely over equilibrium manifolds. The proposed method, called hereafter Active Learning Method (ALM), has the following specificities. First, the surrogates' construction does not use a priori grids in their input spaces. Instead, it relies on evaluation points sequentially selected following an infilling criterion that exploits the current knowledge of the models. Second, the surrogates are not constructed independently: selecting the following evaluation points combines predictions from all subsystems and aims to obtain highly accurate surrogates for input values achieving equilibrium. Third, the proposed ALM uses Gaussian Process surrogates and exploits their probabilistic nature to derive a robust infilling criterion. The final surrogate uncertainty can be propagated into the VPP to assess its predictive qualities.

The organization of the paper is as follows. Section 2 presents the ALM method, with the Gaussian Process construction in Section 2.1, the derivation of the infilling criterion in Section 2.2, and a brief overview of ALM implementation in Section 2.3. We illustrate the ALM in Section 3, for a yacht model detailed in Appendix A with surrogates' settings described in Section 3.1. Section 3.2 compares the convergence of the submodels and their uncertainty for the ALM and a standard QMC sampling method. We analyze the resulting VPP (yacht's polars) and the optimal solution of a routing problem in Section 4. These application examples demonstrate the efficiency of the proposed ALM approach in terms of accuracy for a given number of model evaluations. Finally, major findings of the present work and future extensions of ALM are briefly discussed in Section 5.

2. JOINT SURROGATES CONSTRUCTION

This work focuses on predicting the response $M(x) = y$ of a parametrized system S when its parameters x vary in $\Omega \subset \mathbb{R}^d$, $d \geq 1$. The system's model $M(x)$ is too costly to be evaluated repeatedly. For instance, computing $M(x)$ for some given x may require solving a complex fluid-structure interaction problem. Therefore learning directly the mapping $M(x) = y$ over the whole input domain Ω is not feasible. To tackle this difficulty, a "divide and simplify" approach is used, which consists in splitting the system S into a set of m subsystems $S_{(i)}$, each with a model $M_{(i)}$ that maps inputs $x_{(i)} \in \Omega_{(i)}$ to output $y_{(i)}$:

$$M_{(i)}\left(x_{(i)}\right) = y_{(i)}. \tag{1}$$

Note that the input space $\Omega_{(i)}$ of $S_{(i)}$ is generally not Ω and may have a different dimensionality $d_{(i)}$. This situation occurs when internal variables are needed to describe interactions between the subsystems or when some inputs (say wind characteristics) do not impact a submodel (wave resistance). The domain $\Omega_{(i)}$ may also be unknown, and it is assumed that reasonable bounds can be defined a priori. In this case, the $M_{(i)}(x)$ surrogate models can be constructed independently from the other. We remark that the evaluation of $M(x)$ from the submodels $M_{(i)}(x_{(i)})$ remains complex and calls for iterative techniques to determine the inputs $x_{(i)}$ not specified by x that fulfill the equilibrium and coupling conditions. However, the surrogate models make this task computationally much less expensive.

As stated above, while their construction can be carried out independently, the submodels $M_{(i)}$ have dependent inputs and outputs. Consequently, it is not necessary to build accurate surrogates over the whole $\Omega_{(i)}$, but only over the sub-manifolds where the subsystems satisfy coupling constraints, called hereafter compatibility conditions and introduced in Section 2.2. The main contribution of this paper is a joint infilling strategy to build accurate surrogates over the sub-manifolds where the constraints are satisfied.

2.1. Gaussian Process Model

Splitting the system into subsystems $S_{(i)}$ with simpler numerical models $M_{(i)}$ makes it possible to derive surrogate models. In this work, a Gaussian Processes (GP) model is used,

$$M_{(i)}(x_{(i)}) \approx \mathcal{GP}_{(i)}(\mu_{(i)}, \Sigma^2_{(i)}), \tag{2}$$

where $\mu_{(i)}$ and $\Sigma^2_{(i)}$ are the mean and covariance matrix of the GP model of $M_{(i)}$. This GP model is determined from submodels' observations as described below.

For the sake of readability, the subsystem's index is dropped in this subsection and we restrict ourselves to the case of models with scalar output ($y \in \mathbb{R}$); for a generalization to vector output, see Alvarez *et al.* (2012). Prior to the observations, the GP model assumes zero mean and a covariance function C. The construction of the posterior distribution of the GP model uses n_o observations in the form of couples $(x_k, y_k = M(x_k))$. The matrix of observation points is $\mathrm{X}_o \doteq \left[x_1 \cdots x_{n_o}\right]$ and the vector of observation values is $\mathrm{Y}_o = \left[y_1 \cdots y_{n_o}\right]^T$. Considering new prediction points, with matrix X_* and vector of value Y_*, the observed and predicted values have the *a priori* Gaussian distribution

$$\begin{bmatrix} \mathrm{Y}_o \\ \mathrm{Y}_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathrm{K}_o & \mathrm{K}_{o,*} \\ \mathrm{K}_{o,*}^T & \mathrm{K}_* \end{bmatrix}\right), \tag{3}$$

with the a priori covariance matrices between observation and prediction points $\mathrm{K}_o = C(\mathrm{X}_o, \mathrm{X}_o)$, $\mathrm{K}_* = C(\mathrm{X}_*, \mathrm{X}_*)$ and $\mathrm{K}_{o,*} = C(\mathrm{X}_o, \mathrm{X}_*)$. The posterior distribution of the vector of predicted values Y_*, conditioned on the observations, is given in Rasmussen and Williams (2006)

$$\mathrm{Y}_* | \mathrm{Y}_o, \mathrm{X}_o \sim \mathcal{N}\left(\mathrm{K}_{o,*}^T {\mathrm{K}_o}^{-1} \mathrm{Y}_o, \mathrm{K}_* - \mathrm{K}_{o,*}^T {\mathrm{K}_o}^{-1} \mathrm{K}_{o,*}\right). \tag{4}$$

From this result, the GP model $\mathcal{GP}_{(i)}$ in Eq.(2) has for mean and variance

$$\mu_{(i)}(x) = C(x, \mathrm{X}_o) {\mathrm{K}_o}^{-1} \mathrm{Y}_o, \quad \Sigma^2_{(i)}(x) = C(x, x) - C(x, \mathrm{X}_o)^T {\mathrm{K}_o}^{-1} C(\mathrm{X}_o, x). \tag{5}$$

The mean prediction $\mu_{(i)}(x)$ is the best prediction of $M_{(i)}(x)$, and the posterior variance $\Sigma^2_{(i)}(x)$ characterizes the Gaussian predictive uncertainty. The GP model depends on the prior covariance function C. In this work, the standard squared exponential form for C is used:

$$C(x, x') = \sigma^2 \exp\left(-(x - x')^T \left(\operatorname{diag} L\right)^{-1} (x - x')\right) + \delta(x - x')\sigma_\epsilon^2, \tag{6}$$

where σ^2 is the prior's variance, and L is the vector of correlation lengths along the dimensions of model input x. The last contribution, $\delta(x - x')\sigma_\epsilon^2$, accounts for the observation noise and using $\sigma_\epsilon^2 > 0$ ensures that K_o is invertible. The covariance parameters (σ, L, and σ_ϵ) are collected in the vector of hyper-parameters H. The choice of H is critical for the predictions' quality. In practice, H is learned from the observations. In the present work, the hyper-parameters are set by maximizing the prior likelihood of the observations:

$$H_* \doteq \operatorname{argmax} \mathcal{L}(\mathrm{Y}_o | H), \quad \mathcal{L}(\mathrm{Y}_o | H) = \frac{1}{(2\pi)^{\frac{n_o}{2}} \sqrt{|\mathrm{K}_o|}} \exp\left(-\frac{\mathrm{Y}_o^T \mathrm{K}_o^{-1} \mathrm{Y}_o}{2}\right). \tag{7}$$

Figure 1 illustrates the GP model construction. Figure 1 a) depicts the quantiles of the prior distribution and three random realizations of the GP model drawn from the prior distribution. The plot

in Figure 1 b) depicts quantiles and three realizations of the posterior distribution conditioned on the observations indicated with the crosses. The plot also shows the exact model (dash line). This posterior of the GP model uses the same hyper-parameters as the prior in Figure1 a). It shows that the prediction uncertainty increases as one moves away from the observations. The plot in Figure 1 c) also depicts the posterior quantiles and three realizations but for the hyper-parameters H_* that maximize the likelihood of the observations. Compared to Figure 1 b) it shows the improvement of the prediction and the reduction of the prediction uncertainty resulting from optimizing the hyper-parameters.

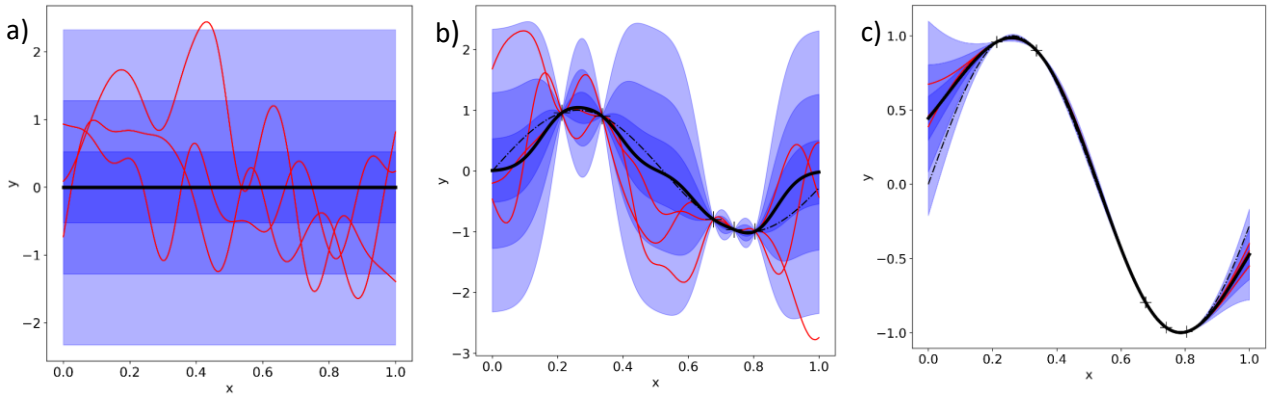

Figure 1. Illustration of the GP model construction. a) Prior model with mean value (solid black line), quantiles in blue contours, and 3 realizations (red lines); b) Posterior model updated from the observations (crosses) and true function (black dashed line); c) Optimized posterior GP model updated from the observations (crosses) and true function (black dashed line).

2.2. Infilling Strategy

The aim of this work is to construct the most accurate GP model $\mathcal{GP}_{(i)}$ of the submodels $M_{(i)}$ for the lowest cost, measured here by the number of submodel evaluations (e.g., model observations n_o). As shown in the previous example, the GP model $\mathcal{GP}_{(i)}$ has lower prediction uncertainty in the neighborhood of the observation points and higher variance far away. This suggests that distributing the observations over the domain evenly can reduce the overall variance of the model on Ω. Uniform grids are not an option except for low dimensional input spaces ($d \leq 4{,}5$) which is why this work relies on random sampling methods such as Latin-Hypercube-Sampling (LHS) and Quasi Monte-Carlo (QMC). This work uses QMC, specifically Halton's sequences (Halton, 1960), as a baseline approach.

Alternatively, one can exploit the current GP model to decide the next observation point sequentially and update the GP model after each new observation is available. The next point should be selected to reduce the GP model's prediction uncertainty effectively. One wants to choose the next observation point without evaluating the model (i.e., making the observation). However, it is only possible to compute the resulting uncertainty reduction by evaluating the model. This difficulty calls for approximations or heuristics. One possibility consists in assuming that the following observation will not affect the optimal hyper-parameters of $\mathcal{GP}_{(i)}$. As seen from the expression of the prediction's variance in Eq.(5), the variance of the GP model depends only on the location of the observation points for a fixed value of the hyper-parameters. It can be easily updated for a new observation point through rank-one updates (Sherman–Morrison–Woodbury formula (Sherman and Morrison, 1950)). The following observation point is thus chosen to maximize the predictive variance reduction over the input domain. A more straightforward approach involves selecting the next observation point at the maximum of the current predictive variance.

A priori methods (LHS, QMC) present the advantage of being easier to implement and are embarrassingly parallel. However, they are less efficient than sequential methods and usually need more model evaluations (observations) to achieve prediction with the same Mean Squared Error (MSE). In the following, the probabilistic nature of the Gaussian processes is leveraged to introduce a new infilling strategy that accounts for the compatibility conditions. Specifically, the set of submodels' input $x_{(i)}^{+}$ for the following observations is defined through the generic optimization problem

$$\left(x_{(1)},\cdots,x_{(m)}\right)^{+} = \underset{\left(x_{(1)},\cdots,x_{(m)}\right)\in \Omega_{(1)}\times \ldots \times \Omega_{(m)}}{\operatorname{argmax}} IC\left(x_{(1)},\ldots,x_{(m)}\right), \tag{8}$$

where the acquisition function, or Infilling Criterion, IC defines the specific sequential strategy. In the following, we denote $X = \left(x_{(1)},\cdots,x_{(m)}\right)$ the vector of submodel inputs and Ω_X its range, such that $IC\colon \Omega_X \mapsto \mathbb{R}$. With this structure of IC, all submodels' next evaluation points are generally interdependent and determined simultaneously. The definition of IC is chosen to select input points whose evaluations reduce the models' variance after the update and satisfy the compatibility conditions. It suggests a composite structure for IC, with a variance and compatibility parts.

Starting with the compatibility part, we denote Z the vector gathering all the compatibility constraints. The submodel inputs are compatible when $\| Z(X) \| = 0$, that is, when $X \in \Omega^{\mathrm{eq}}$ where $\Omega^{\mathrm{eq}} \subseteq \Omega_X$ is the equilibrium manifold. Note that from Ω^{eq} one can derive the equilibrium manifolds $\Omega_{(i)}^{\mathrm{eq}}$ of the submodels. The vector of constraints $Z(X)$ would be too costly to evaluate for given $X \in \Omega_X$; which is why $\tilde{Z}(X)$ its approximation based on the GP models $\mathcal{GP}_{(i)}$ is introduced. The main idea of this work is to leverage the probabilistic nature of the GP-based predictions to derive the probability distribution of $\tilde{Z}(X)$. In this work, only the case of compatibility constraints linear in the submodels is tackled. More general situations can be handled via Monte-Carlo sampling or Taylor expansions (local linearisations); see, for instance, Sanson *et al.* (2019). The GP models' construction is separate, so $\mathcal{GP}_{(i)}$ and $\mathcal{GP}_{(j)}$ are independent when $i \neq j$. Therefore, in the linear case, the vector $\tilde{Z}$ is Gaussian, and one can explicitly derive its mean value $\mu_Z(X)$ and covariance matrix of $\Sigma_Z^2(X)$ from the means $\mu_{(i)}(x_{(i)})$ and covariance functions $\Sigma_{(i)}^2(x_{(i)})$ of the GP models. Specifically, for weight matrices $A_{(i)}$ of the linear combination and $Z(X) = \sum_i A_{(i)}\, \mathcal{GP}_{(i)}(x_{(i)})$, it comes

$$\mu_Z(X) = \sum_i A_{(i)}\, \mu_{(i)}\left(x_{(i)}\right), \quad \Sigma_Z^2(X) = \sum_i A_{(i)}\, \Sigma_{(i)}^2\left(x_{(i)}\right) A_{(i)}^T. \tag{9}$$

The approximate vector of constraints thus follows the multivariate normal distribution,

$$\tilde{Z}(X) \sim \mathcal{N}\left(\mu_Z(X), \Sigma_Z^2(X)\right), \tag{10}$$

and the density of $\tilde{Z}$ given the input values $X \in \Omega_X$ becomes

$$p_{\tilde{Z}}(z|X) \propto \frac{1}{|\Sigma_Z^2(X)|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mu_Z(X)-z)^T \left(\Sigma_Z^2(X)\right)^{-1} (\mu_Z(X)-z)\right). \tag{11}$$

The density at zero of $\tilde{Z}$ measures the likelihood of the GP model to satisfy the compatibility conditions at the input point $X = \left(x_{(1)},\cdots,x_{(m)}\right)$

$$\mathcal{L}_{Z=0}(X) = p_{\tilde{Z}}(0|X) \propto \frac{1}{|\Sigma_Z^2(X)|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\mu_Z(X)^T \left(\Sigma_Z^2(X)\right)^{-1} \mu_Z(X)\right). \tag{12}$$

Now focusing on the variance part of the definition of IC, the model's variance reduction over $\Omega_{(i)}$ induced by a new observation at a given point $x_{(i)}$ is computable, assuming that the hyper-

parameters remain unchanged. The selection of the new points could involve the integral models' variance reduction *weighted* by the likelihood of satisfying the compatibility conditions. A Monte Carlo method could estimate such an integral, but the procedure would be computationally expensive when the likelihood concentrates on the compatibility manifolds. Therefore, a local criterion based on the models' variance at new observation points was chosen. The formulation we chose is to sum the (independent) models' uncertainties to get

$$IC(X) = \left[\sum_{i=1}^{m} \mathrm{Tr}\left(\Sigma_{(i)}^2\left(x_{(i)}\right)\right)\right] \mathcal{L}_{Z=0}(X), \tag{13}$$

where $\mathrm{Tr}\left(\Sigma_{(i)}^2\right)$ is the trace of the predictions' covariance of $\mathcal{GP}_{(i)}$. The first term of IC drives the next sampling points in areas where the GP models prediction uncertainties are the highest. The second term focuses the sampling in places where the compatibility constraints have a high chance of being satisfied.

2.3. Infilling Algorithm

The construction of the GP models of the subsystems begins with an initialization step, where some input points are drawn randomly in the $\Omega_{(i)}$; the submodels $S_{(i)}$ are evaluated at these points and the GP models $\mathcal{GP}_{(i)}$ are constructed. This step includes the optimization of the hyperparameters $H_{(i)}$. In this work the DIRECT algorithm (Jones *et al.*, 1993) from the NLopt library (Johnson 2023) is used.

After the initialization step, the infilling strategy proceeds to solve

$$\left(x_{(1)}, \cdots, x_{(m)}\right)^+ = \underset{\left(x_{(1)}, \cdots, x_{(m)}\right) \in \Omega_{(1)} \times \ldots \times \Omega_{(m)}}{\mathrm{argmax}} \left[\sum_{i=1}^{m} \mathrm{Tr}\left(\Sigma_{(i)}^2\left(x_{(i)}\right)\right)\right] \mathcal{L}_{Z=0}\left(x_{(1)}, \cdots, x_{(m)}\right). \tag{14}$$

In practice, some submodels share some of their inputs $x_{(i)}$ with other submodels, and the optimization must account for these structures in the inputs. One possibility is to add equality constraints to the optimization problem. Alternatively, one can reduce the vector X of inputs to remove redundancies. The work uses this second approach, solving the optimization problem with the NLopt library. Note that any other optimization utility could be used for that purpose.

When the following input values $x_{(i)}^+$ are determined, $M_{(i)}(x_{(i)}^+)$ is evaluated to obtain a new observation and update $\mathcal{GP}_{(i)}$ accordingly. After the update of all GP models, the procedure is repeated until the computational budget is exhausted or some stopping criteria are met. This algorithm presents the advantage of determining the new inputs simultaneously, enabling the parallel evaluation of the submodels. In some situations, updating only a few or just one submodel may be preferred before reconsidering Eq. (14). For instance, one could decide to update individual models using their respective variances $\mathrm{Tr}\left(\Sigma_{(i)}^2\right)$. Alternatively, one could consider the numerical cost of the models by adapting the multi-fidelity strategy proposed in Pellegrini, *et al.* (2016).

3. APPLICATION TO A SAILING YACHT MODEL

The proposed method is applied to a sailing yacht model to illustrate the efficiency of separated surrogates' construction with the proposed infilling strategy. The assessment of the method includes the surrogates error analysis, in Section 3.2, and the induced errors when using these models in a Velocity Prediction Program (VPP) and a routing problem, in Section 4. In order to assess these errors, a simple and relatively low-cost yacht model that permits the exact VPP computation for reference is chosen.

3.1. Yacht Model

The yacht characteristics and its model is fully described in Appendix A. It is split into two submodels $M_{(a)}$ and $M_{(h)}$ (i.e. $m = 2$), predicting respectively the aerodynamic and hydrodynamic forces and moments acting on the yacht. Two Gaussian-Processes, $\mathcal{GP}_{(a)}$ and $\mathcal{GP}_{(h)}$ subsequently approximate the submodels. As previously discussed, one can construct the submodels' surrogates separately, but it is advantageous to concentrate the construction effort on inputs value that satisfy a compatibility condition. In the present yacht model, the compatibility condition expresses the static equilibrium of the yacht. For simplicity, only the yacht's three principal degrees of freedom (see Larsson, 1999)), namely the propulsion and drift forces and the heeling moment, are considered in this work such that $\mathcal{GP}_{(a,h)} \in \mathbb{R}^3$ and the compatibility condition writes

$$\tilde{Z}(X) = \mathcal{GP}_{(a)}(X) + \mathcal{GP}_{(h)}(X) = 0. \tag{15}$$

Overall, the yacht model uses six input variables ($d = 6$) to determine the forces (see Appendix A): the boat speed (V_b), the heel angle (θ), the leeway angle (λ), the flat (f) parameter of the sails, the true wind angle (β_a) and the true wind speed (β_t). The inputs of the hydrodynamic model are only the first three inputs, whereas the aerodynamic model uses all the inputs. Table 1 reports the (a priori) inputs' range for constructing the surrogates.

These ranges are deliberately large and involve unrealistic situations that challenge the physical validity of some loads' model. The heel range, for instance, goes from -20 to 90 degrees, when some force models implicitly assume small heel angles and negative heel angles may not be feasible at equilibrium. Our goal when choosing these large ranges is to demonstrate the robustness of the ALM algorithm and its capacity to focus on the equilibrium manifold.

Table 1. Inputs' range.

Input	V_b [kt]	θ [˚]	λ [˚]	f [-]	β_t [˚]	β_a [˚]
Lower bound	0.1	-20	-7	0	2	0
Upper bound	10	90	7	1	22	180

3.2. Sequential Surrogates Construction

The method described in Section 2 is then applied to this yacht mode, constructing GPs of the aerodynamic and hydrodynamic models iteratively. The construction differs slightly from the uni-dimensional case exposed in Section 2.1, as the models are vector values with components being the three loads of interest. At each step, we apply affine transformations to standardize componentwise the data (Principal Component Analysis) before the GPs construction. These transformations preserve the constraints' Gaussian structure while balancing the models' magnitude and easing the prescription of hyperparameter ranges. Other strategies were also tested, such as those presented in Alvarez *et al.* (2012). However, we found that, in this problem, these methods were not noticeably improving the accuracy while demanding the optimization of significantly more hyperparameters.

In this section and the following, we contrast the predictions of two methods to demonstrate the performance of the proposed method. To serve as a reference, we build the GP models on data points sampled with the Quasi-Monte Carlo method over the whole domain of the inputs. More precisely, we rely on Halton sequences. Besides its sampling qualities, the QMC method enables the prolongation of the sequence, a feature exploited to monitor the convergence. Our approach, in

contrast, uses the current models to select the new sample point at each step from the infilling criteria given by Eq. (14); it then evaluates the aerodynamic and hydrodynamic models at the new point before proceeding with the update of the GP models.

For a quantitative measure of the error, we define the subset of the input space $\Omega_\epsilon \subset \Omega_X$ as the set of inputs satisfying the constraints within a tolerance ϵ:

$$\Omega_\epsilon = \{X \in \Omega_X, \| Z(X) \| \leq \epsilon\}. \tag{16}$$

We then measure the GP-based prediction (mean value) of the loads and constraints errors using the $L_2(\Omega_\epsilon)$ error. For the constraints, for instance, we define

$$\| Z(X) - \mu_Z(X) \|^2_{L_2(\Omega_\epsilon)} = \int_{\Omega_\epsilon} \| Z(X) - \mu_Z(X) \|^2 \, dX. \tag{17}$$

In practice, evaluating the integral over the non-explicit domain Ω_ϵ uses a uniform Monte-Carlo sample set drawn by a standard rejection method over the whole domain Ω_X. Since the rejection rate increases when ϵ decreases, we have progressively lowered ϵ until the averaged error $\| Z(X) - \mu_Z(X) \|^2_{L_2(\Omega_\epsilon)} / |\Omega_\epsilon|$ becomes ϵ-independent. The results presented hereafter use $\epsilon = 1{,}000$.

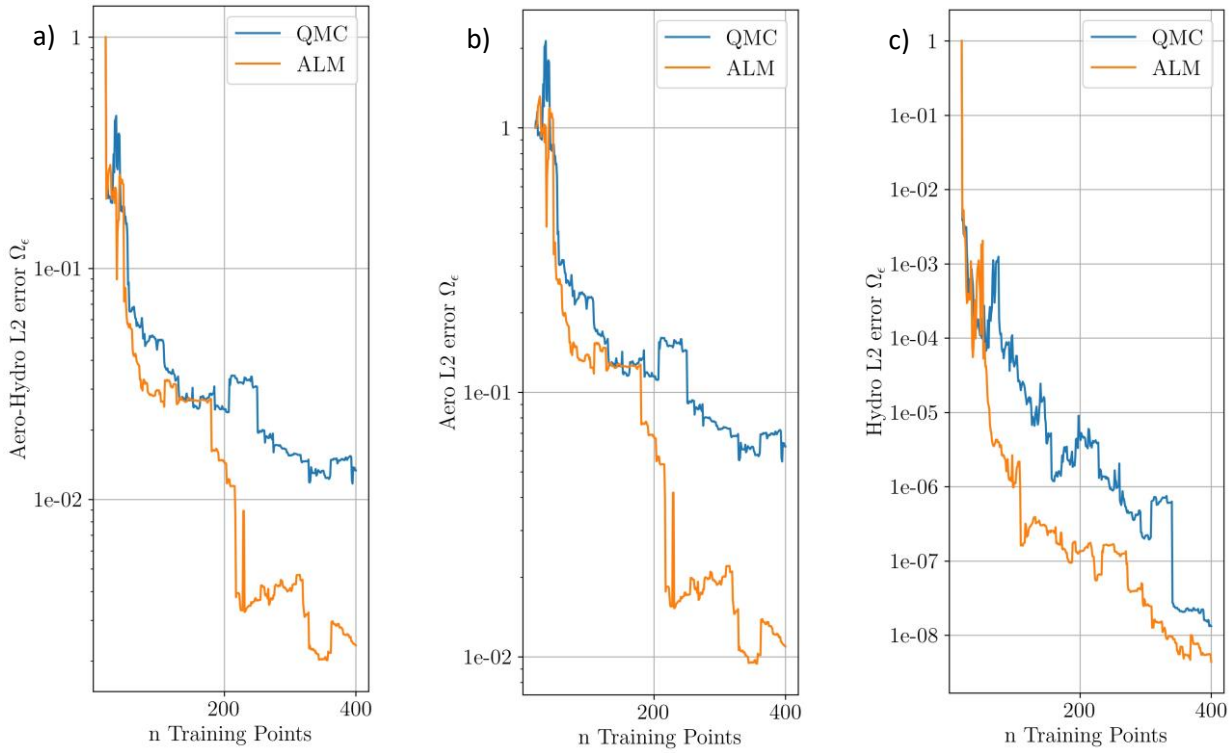

Figure 2. Evolutions with the training set dimension of the $L_2(\Omega_\epsilon)$ errors on the predicted constraints a), aerodynamic forces b), and hydrodynamic forces c). The plots are for the proposed method (ALM) and the QMC sampling.

Figure 2 compares these errors for the QMC and our method, labeled ALM. Figure 2 a) shows the errors of the constraints in $L_2(\Omega_\epsilon)$ norms (normalized by their initial values) as functions of the number of sample points in the data basis. The initial models for the two methods use the same (QMC) sample set of 20 points. Figure 2 b) shows the evolution of the aerodynamic model errors, and Figure 2 c) concerns the hydrodynamic model. The plot of the errors on the constraint shows that after an initial stage where the QMC and ALM methods yield comparable errors, the ALM method produces much more precise estimations of the constraint when the number of training

points exceeds 175. Specifically, an error reduction of roughly one order of magnitude is reported when $n > 300$. Also, the plots of the errors on the loads highlight that the aerodynamic (Figure 2 b)) is the most challenging to construct and quickly exceeds the hydrodynamic error (Figure 2 c)) by several orders of magnitude. It is, therefore, the main contributor to the equilibrium error. This behavior is explained by more complex dependencies and the highest dimensionality of the aerodynamic model, with six input dimensions compared to three for the hydrodynamic model. Consistently, the error of the ALM and QMC hydrodynamic models are much closer than for their aerodynamic counterparts, denoting that the ALM focuses on reducing the constraints error through its principal contributor.

Figure 3 illustrates the differences in the structure of the training sets generated with the QMC and ALM methods. The plot shows the projection of the training points in the (V_b, λ) plan. Also reported with red crosses is the projection of points uniformly sampled from Ω_ϵ and used to estimate the $L_2(\Omega_\epsilon)$ norms. We see that when the distribution of the QMC points (blue) is uniform in the plan, the distribution of the ALM points (orange) principally concentrates in areas of low constraint values (Ω_ϵ), explaining the improved accuracy in these regions. The plot also highlights that ALM can accommodate situations combining very narrow and substantial domains of quasi-equilibrium. For instance, we observe a quasi-uniform ALM sampling in the λ direction at low V_b. The uniform sampling is due to low hydrodynamic and aerodynamic forces at low V_b and β_t, which require exploring the whole parametric space to sufficiently reduce the prediction variance and confidently decide whether the equilibrium is likely. In contrast, when the loads are important (higher V_b), a comparable prediction variance will not be as critical in the equilibrium assessment far from the manifold. ALM then concentrates the samples within narrow domains.

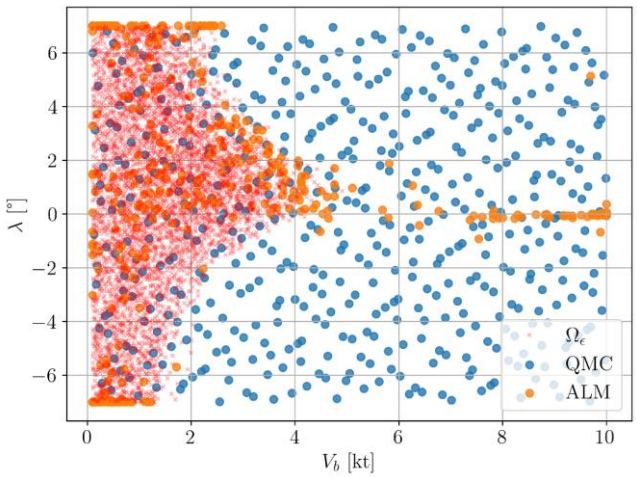

Figure 3. Projection of the sampling points on the (V_b, λ) plan. Small red crosses are points in Ω_ϵ, blue dots correspond to QMC sample points, and orange dots are the sampling points for the ALM method.

4. APPLICATION TO VPP AND ROUTING

In this section we exploit the models build in the previous section in two applications: a Velocity Prediction Program and routing.

4.1. Surrogates-based Velocity Prediction Program

We formulate the VPP problem as follows: given some wind conditions, (β_a, β_t), find the maximum boat speed V_b and associated trimming parameters (θ, λ, f) within the input domain that satisfy the equilibrium of the yacht. Solving the VPP problem for several combinations of (β_a, β_t) yields the so-called yacht's polar, which characterizes the yacht's performance in the industry.

There are several ways to solve this constrained optimization problem. In the present work, a hybrid approach is employed. It combines a pseudo-transient method that computes the terminal (stationary) $(\mathrm{V}_b, \theta, \lambda)$ values for given f, with an external optimization loop that determines the FL value leading to the highest V_b. This approach, like others, requires many evaluations of the aerodynamic and hydrodynamic models. For the simple yacht model considered here, it is possible to solve the VPP problem directly, i.e., using the exact models. We can then assess the impact on the VPP of the GP models when they replace the exact models.

In Figure 4, we report the yacht's polar for four values of β_t (5, 10, 15 and 20 kt) and several β_a, using the exact models and GP surrogates based on 300 training points with the ALM and QMC methods. These experiments use the best GP prediction, i.e., their mean, to replace the exact model. We observe that for ALM, the approximation is consistently close to the exact polar. In contrast, the QMC-based GP models yield more significant deviations, particularly at low β_t values (most inner polar) and for large β_a. One can better appreciate the differences in the surrogate methods in Figure 5, which shows the evolution of the mean squared error (MSE) of the VPP with the size of the training set. The ALM has an MSE reduced by roughly two orders of magnitude compared to the naive QMC method when the training set size exceeds 300.

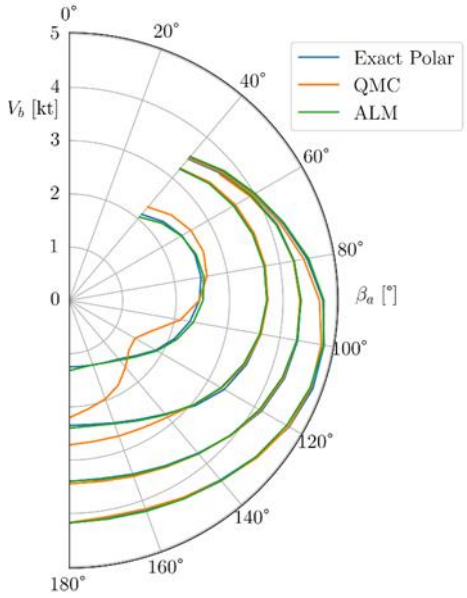

Figure 4. comparison of yacht polars computed with the exact (blue), QMC (orange) and ALM (green) models, and for β_t = 5, 10, 15 and 20 kt (the higher the true wind speed, the higher the V_b). The GP-based polars use the mean prediction from 300 training points.

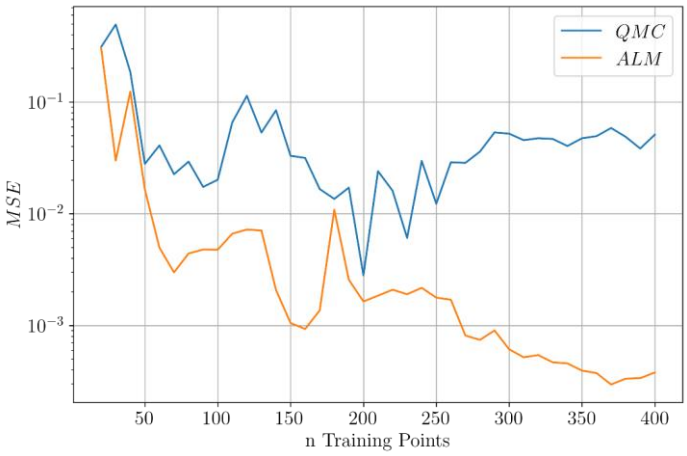

Figure 5. Evolution with the training set size of the polars' MSE.

Figure 4 showed that the Adaptive Learning method produces more accurate polars than the QMC sampling method. It was evidenced by reporting the distance between the mean-based polars and their exact model counterparts. However, another critical aspect of the ALM method is that it provides a much lower uncertainty level in the prediction. Figure 6 illustrates this characteristic for the ALM and QMC methods, respectively; they show 50 random samples of the polars computed using 50 independent realizations of the aerodynamic and hydrodynamic GP models. These polars are generated by solving the VPP problem using correlated samples of the GP models' posterior distributions. One can observe how correlations between the models' input values translate into smooth polar samples. Further, the plots highlight the reduced uncertainty in the polars for the ALM approach (Figure 6 a)) compared to the QMC sampling (Figure 6 b)): the sample dispersion is significantly less in the former case. Characterizing this spread in the polar prediction is essential to estimate the confidence in the constructed model and eventually decide to refine the GP model further.

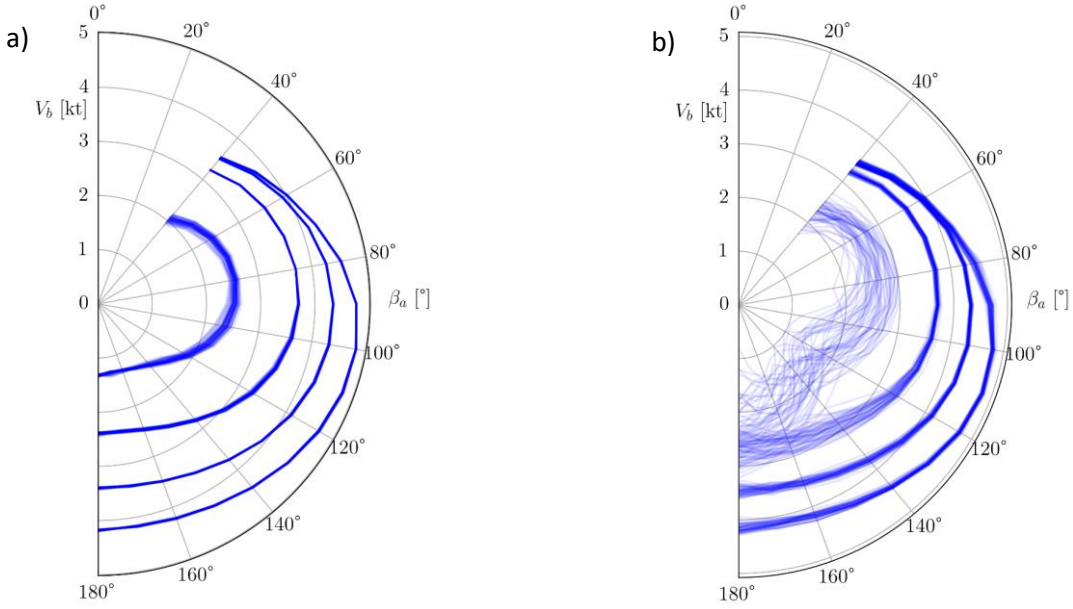

Figure 6. Fifty random samples of the polars for the surrogates constructed with 300 training point for a) ALM and b) QMC. Polars are shown for true wind speed values β_t = 5, 10, 15 and 20 kt corresponding to increasing V_b.

4.2. Routing with Surrogate Models Uncertainty

We compare the QMC and ALM GP construction on a routing application to complete the comparison. Specifically, we selected the course of the next edition of the Fastnet race, which departs in the Solent, rounds the Fastnet light-house (southeast of Ireland) and ends in Cherbourg. The routing utility is the QTVLM software (see QTVLM (2023)) which computes the isochrone from yacht polars and weather data to produce the route to complete the course in shortest time. The optimal route depends on the polars of the yacht which are fed to QTVLM. The routing uses the weather conditions predicted by the GFS forecast model on the 24th of March, 2023 at 0000 GMT for a start of the course on the same day at 1600 GMT. Note that the time resolution of the QTVLM software is 5 minutes.

We perform the routing for the 50 realizations of the polars shown in Figure 6. The resulting routes are reported in Figure 7. For ALM, see Figure 7 a), the spread of the routes is limited, except for the channel crossing to reach Cherbourg at the end of the course. In contrast, the spread of the routes for the QMC approach depicted in Figure 7 b) becomes significant much earlier. One may notice an outlier route induced by a very poor model realization in the QMC approach, with an early channel crossing at the start and a northern route in the first crossing of the Irish Sea to the Fastnet rock.

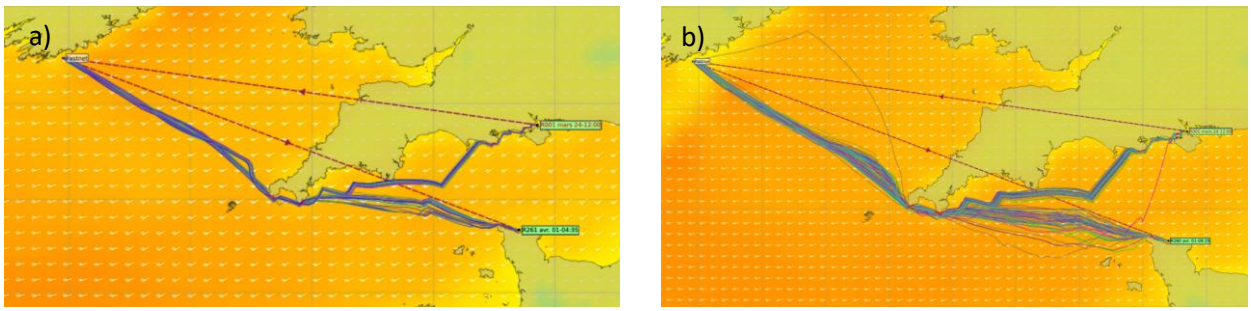

Figure 7. Route distribution for 50 polar samples using a) ALM (200 training points) and b) QMC (300 training points).

We further assess the impact of the models' uncertainty by reporting the induced statistics of the route's duration or estimated time of arrival (ETA) produced by the routing software.

Table 2 summarizes the statistic of the ETA for the two methods and two training set sizes. The second column provides the ETA based on the mean of the GP models (polars of the best model predictions). We observe that the mean QMC models overestimate the ETA of the exact model (last row of the table) by more than 1.5 h when the mean ALM models are in better agreement with less than 40 minutes error. The third and fourth columns provide the average and standard deviation of the ETA estimated from the 50 random samples of the polars. The sample averages of the ETA for ALM are within 15 minutes of the exact ETA, with a consistent standard deviation of roughly 20 minutes. In contrast, the sample-averaged ETA for the QMC construction varies from 1.5 to 2 hours, with larger standard deviations that are inconsistent. For instance, for QMC with 300 training points, an average ETA of 182 hours (short of roughly 2 hours) is predicted, with a standard deviation of only 36 minutes, when the ETA for the corresponding mean models is 185.6 hours. These results highlight the lack of robustness of the QMC method, whose significant GP models' uncertainty induces high polar variabilities and long-tailed ETA distributions.

Table 2. Estimated ETA for the ALS and QMC methods using two training set sizes (200 and 300 points). The Table reports the ETA of the mean model (2nd column), the sample average, and the standard deviation of the ETA (3rd and 4th column). The ETA for the exact model is also reported (last line).

Method	ETA for mean prediction	Sample average ETA	Standard Dev. ETA
QMC (200 pts)	185 h 20 min	185 h 27 min 36 sec	1 h 15 min 55 sec
QMC (300 pts)	185 h 40 min	182 h 00 min 20 sec	0 h 36 min 53 sec
ALM (200 pts)	183 h 35 min	184 h 04 min 30 sec	0 h 23 min 35 sec
ALM (300 pts)	183 h 10 min	184 h 02 min 06 sec	0 h 21 min 37 sec
Exact model	183 h 30 min	–	–

The spread of the ETA distribution for the different methods can be appreciated from Figure 8. In this plot, we have estimated the densities by applying a standard kernel density method on the sample sets of 50 ETAs associated with the random samples of the polars. The change in the densities for the QMC methods between 200 and 300 training points is striking, especially when compared to the two ALM densities.

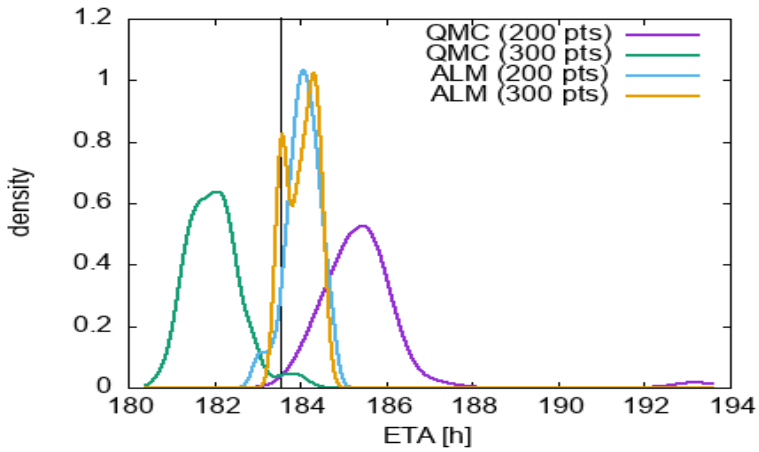

Figure 8. Distributions of ETA predicted by ALM and QMC methods using 200 and 300 training points, as indicated. The densities are estimated using 50 random samples of the GP models. The vertical line shows the ETA for the exact models (183.5 h).

5. CONCLUSIONS

This work presents a new method of constructing surrogate models of complex systems. The key ingredients of the proposed methods are

- partition of the system into simpler subsystems with compatibility conditions
- sequential construction of the subsystems' Gaussian Process surrogates, with a progressive enrichment of the training sets driven by the current system approximation
- Adaptive Learning Method aiming to reduce the surrogate errors for input values likely to satisfy the compatibility conditions

The Active Learning method (ALM) relies heavily on the probabilistic nature of Gaussian Process surrogates representing the subsystems. We derive an infilling criterion from the probabilistic information that targets input values with high prediction variance and a high likelihood of solving the compatibility conditions.

We apply the ALM to a simple yacht model. We show that ALM results in GP surrogates of the aerodynamic and hydrodynamic forces that are more accurate in the neighborhood of the compatible domain than for non-informed construction methods, such as Quasi-Monte Carlo (QMC) sampling. Our tests show that ALM requires significantly fewer submodel evaluations to achieve a prescribed prediction accuracy compared to a standard QMC sampling method. This reduction of submodel evaluations will directly translate into computational savings, provided that the CPU cost of the submodels' evaluation dominates the cost of solving the sequence of optimization problems for the infilling points. ALM targets these situations, which should correspond to most applications involving complex CFD codes and structural models.

The resulting surrogates were applied to solve a VPP problem and compute polars. The polars predicted by the mean GP surrogates built with the active learning method can be up to ten times more accurate than the polars obtained without the adaptive strategy (QMC). The improvement will be even more impressive for more complex yacht models and higher dimensional input spaces. However, the improvement may be difficult to quantify if exact polars are unavailable.

Reductions in the GP surrogates' prediction uncertainty accompany the gain in accuracy of the mean prediction for ALM. This point is particularly relevant as the prediction uncertainty is crucial when assessing the surrogate model quality and the confidence in the model prediction. The paper has illustrated these aspects by reporting posterior uncertainties in the VPP (polars). We also draw

models from their posterior distribution to conduct an ensemble routing analysis and estimate the statistical spread of course duration due to model uncertainty.

The ALM can be improved in several aspects. First, we would like to extend the infilling criterion to account for possibly very heterogeneous computational costs of the submodels, along the lines of the cost-informed infilling strategies proposed in Pellegrini *et al.*, 2016 and Sacher *et al.* (2021). Similarly, the proposed approach is purely sequential with just one infilling point determined at each iteration, and it could be extended to generate a batch of infilling points in order to exploit parallelism better.

6. ACKNOWLEDGMENTS

This work is supported by the French Agence Nationale Recherche Technologie (ANRT) through the CIFRE grant n°2019/0680 and by the naval architecture bureau Bañulsdesign. The authors wish to thank QTVLM for its assistance in modifying their software to accommodate the use of polar distributions in routing problems. The authors thank the anonymous reviewers for their constructive comments, which helped improve this article.

7. REFERENCES

Alvarez, M., Rosasco, L. and Lawrence, N. (2012). Kernels for Vector-Valued Functions: a Review. *Foundation and Trends in Machine Learning,* 4, 195–266.

Claughton, A., Fossati, F., Battistin, D. and Muggiasca, M. (2008). Changes and Development to Sail Aerodynamics in the ORC International Handicap Rule. *20th Int. Symp. on Yacht Design and Yacht Construction.*

Davidson, K. S. M. (1936). Some Experimental Studies of the Sailing Yacht. *Stevens Institute of Technology, Experimental Towing Tank, Hoboken, New Yersey, Presented at The Society of Naval Architects and Marine Engineers, SNAME.*

Duchon, J. (1977). *Splines Minimizing Rotation-Invariant Semi-Norms in Sobolev Spaces.* Vol. 571, in *Constructive Theory of Functions of Several Variables*, édité par A. Dold, B. Eckmann, W. Schempp et K. Zeller, Springer, 85–100.

Halton, J. H. (1960). On the Efficiency of Certain Quasi-Random Sequences of Points in Evaluating Multi-Dimensional Integrals. *Numerische Mathematik,* 2, 84–90.

Horel, B. (2022). Review of Existing Benchmarks and Databases for Sailing Vessels. *Journal of Sailing Technology, 7:1, 52-87.*

International Towing Tank Conference (1957). *8th International Towing Tank Conference.* Madrid, Spain.

Johnson, S. G. (2023). *The NLopt nonlinear-optimization package.* http://github.com/stevengj/nlopt.

Jones, D. R., Perttunen, C. D. and Stuckman, B. E. (1993). Lipschitzian Optimization without the Lipschitz Constant. *J. Optimization Theory and Applications,* 79, 157–181.

Kerwin, J. E. (1975). A Velocity Prediction Program for Ocean Racing Yachts. *The Society of Naval Architects and Marine Engineers, SNAME, Report 75-1, Massachusetts Institute of Technology, MIT, Department of Ocean Engineering, Ocean Race Handicapping Project.*

Keuning, J. A. and Katgert, M. (2008). A Bare Hull Resistance Prediction Method Derived From the Results of The Delft Systematic Yacht Hull Series Extended to Higher Speeds. *Proc. Int. Conf. on Innovation in High Performance Sailing Yacht,* 13–21.

Larsson, L. (1999). *Principles of Yacht Design.* 2nd ed. London: Adlard Coles Nautical.

Lindstand Levin, R. and Larsson, L. (2017). Sailing Yacht Performance Prediction Based on Coupled CFD and Rigid Body Dynamics in 6 Degrees of Freedom. *Ocean Engineering,* 144, 362-373.

Melis, M. F., Hansen, H. and Fischer, M. (2022). Velocity Prediction Program for a Hydrofoiling Lake Racer. *Journal of Sailing Technology,*7:1, 255-275*.*

Pellegrini, R., Lemma, U., Leotardi, C., Campana, E. F. and Diez, M. (2016). Multi-Fidelity Adaptive Global Metamodel of Expensive Computer Simulations. *2016 IEEE Congress on Evolutionary Computation (CEC),* IEEE. 4444–4451.

Persson, A., Larsson, L. and Finnsgård C. (2021). An Improved Procedure for Strongly Coupled Prediction of Sailing Yacht Performance. *Journal of Sailing Technology* 6, 133–150.

Prandtl, L. (1925). Bericht über Untersuchungen zur ausgebildeten Turbulenz. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik,* 5, 136–139.

QTVLM, 2023. https://www.meltemus.com/index.php/en/.

Rasmussen, C. E., and Williams, C. (2006). *Gaussian Processes for Machine Learning.* Cambridge, Mass, MIT Press.

Reche-Vilanova, M., Hansen, H. and Bingham, H. B. (2021). Performance Prediction Program for Wind-Assisted Cargo Ships. *Journal of Sailing Technology,* 6, 91–117.

Sacher, M., Le Maître, O., Duvigneau, R., Hauville, F., Durand, M. and Lothodé, C. (2021). A Non-Nested Infilling Strategy for Multifidelity Based Efficient Global Optimization. *International Journal for Uncertainty Quantification,* 11, 1–30.

Sanson, F., Le Maître, O. and Congedo, P.M. (2019). Systems of Gaussian Process Models for Directed Chains of Solvers. *Computer Methods in Applied Mechanics and Engineering,* 352, 32-55.

Sherman, J. and Morrison,W. J. (1950). Adjustment of an Inverse Matrix Corresponding to a Change in One Element of a Given Matrix. *The Annals of Mathematical Statistics,* 21, 124–127.

APPENDIX. SAILBOAT MODEL

In this appendix we detail the sailboat model used in the numerical experiments. This model was selected for its simplicity that enable fast computations, and does not aim at reflecting an actual yacht. In particular some part of the model may be used for conditions that may lead to non-physical prediction (e.g., detached flows) for the range of inputs considered for the construction of the surrogates.

A.1. Yacht Characteristics and Physical Properties

The input variables of θ and λ are defined in a coordinate system given in Figure 9, where the yacht is on starboard tack and its direction of travel is along the x_{ori} vector, the y_{ori} vector is pointing to leeward (port) and the z_{ori} is directed upwards and normal to the free-surface. The angle θ

represents a negative rotation around the x_{ori} axis, increasing the yacht's righting moment. The angle λ represents a negative rotation around the z_{ori} axis, yielding a drift of the yacht to port.

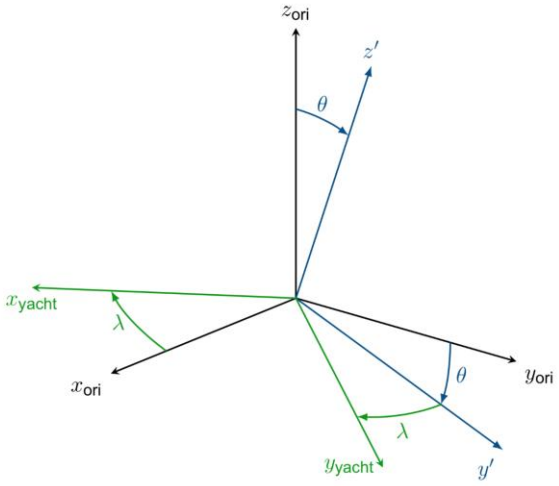

Figure 9. Reference frame.

The yacht characteristics are listed in Table 3, while Table 4 reports the physical properties used in the model.

Table 3. Yacht characteristics.

Characteristics	Symbol	Value	Unit
Length at the waterline	L_{wl}	9.14	[m]
Beam at the waterline	B_{wl}	3.25	[m]
Displacement	D	2.8	[m^3]
Longitudinal center of buoyancy (from forward perpendicular)	L_{cb}	4.81	[m]
Longitudinal center of flotation (from forward perpendicular)	L_{cf}	4.81	[m]
Draft of canoe body	T_{c}	0.4	[m]
Total draft	T_{t}	2.0	[m]
Prismatic coefficient	C_{p}	0.55	[−]
Midship coefficient	C_{mid}	0.64	[−]
Daggerboard surface	S_{dagg}	1.125	[m^2]
Daggerboard vertical center of effort	Z_{dagg}	-0.8	[m]
Daggerboard aspect ratio	Λ^{dagg}	2	[−]
Initial meta-centric height	G_{M}	0.8	[m]
Wetted surface area	W_{SA}	15	[m^2]
Area of flotation plane	A_{w}	15	[m^2]
Sail area	S_{sail}	55	[m^2]
Aerodynamic vertical center of effort	Z_{a}	-0.8	[m]

Table 4. Physical properties.

Quantity	Symbol	Value	Unit
Viscosity of sea water	ν_w	$1e^{-6}$	$[m^2 s^{-1}]$
Density of sea water	ρ_w	1,025	$[kg\ m^{-3}]$
Density of air	ρ_a	1.22	$[kg\ m^{-3}]$
Acceleration of gravity	g	9.81	$[m\ s^{-2}]$

A.2. Aerodynamic and Hydrodynamic Forces

The aerodynamic forces are modeled using the ORC VPP documentation (Claughton, et al. 2008). They depend on the apparent wind speed V_a and apparent wind angle γ_a defined by

$$V_a{}^2 = \left(\mathrm{V}_b + \beta_t \cos(\beta_a)\right)^2 + \left(\beta_t \sin(\beta_a)\right)^2, \quad \gamma_a = \arccos\left(\frac{\mathrm{V}_b + \beta_t \cos(\beta_a)}{V_a}\right). \tag{18}$$

The driving force, side force and heeling moment of the sail plan are computed using the following formulas (with V_a in m s^{-1}):

$$F_x^a = \frac{1}{2}\rho_a \mathrm{S}_{\mathrm{sail}} V_a{}^2 (C_l^a\ f \sin(\gamma_a) - C_d^a \cos(\gamma_a)), \tag{19}$$

$$F_y^a = \frac{1}{2}\rho_a \mathrm{S}_{\mathrm{sail}} V_a{}^2 (C_l^a\ f \cos(\gamma_a) + C_d^a \sin(\gamma_a)) \cos(\theta), \tag{20}$$

$$M_x^a = \frac{1}{2}\rho_a \mathrm{S}_{\mathrm{sail}} V_a{}^2 Z_{\mathrm{a}} (C_l^a f \cos(\gamma_a) + C_d^a \sin(\gamma_a)), \tag{21}$$

Where the aerodynamic lift and drag coefficients, C_l^a and C_d^a, are reported in Table 5 for several apparent wind angles. The coefficients are interpolated for other apparent wind angles using a Thin Plate Spline method (see Duchon (1977)). The flat parameter f takes values in the range [0, 1] to emulate strong de-powering (as the lift coefficient of the sail can effectively be reduced to 0) and simulate globally the easing and reefing of the sail.

Table 5. Aerodynamic force coefficients with apparent wind angle.

γ_a [°]	0	7	9	12	28	60	90	120	150	180
C_d^a	0.0431	0.0258	0.0232	0.0232	0.0325	0.1130	0.3825	0.9688	1.3157	1.3448
C_l^a	0.0	0.8620	1.0517	1.1637	1.3469	1.3534	1.2672	0.9310	0.3879	-0.1120

The viscous hydrodynamic drag coefficient of the hull is taken from the classical ITTC-57 formula (International Towing Tank Conference 1957),

$$C_f = \frac{0.075}{(\log(Re_h) - 2)^2}, \tag{22}$$

where $Re_h = \frac{VL_{\mathrm{wl}}}{\nu_w}$ is the hull's Reynolds number, V being the boat speed expressed in m s^{-1}. The wave resistance R_{wav} is estimated from the Delft-Series Experiments (Keuning and Katgert 2008), through the regression given by Eq. (23):

$$\frac{R_{\mathrm{wav}}}{\rho g D} = a_0 + \left(a_1 \frac{L_{\mathrm{cb}}}{\mathrm{L_{wl}}} + a_2 C_{\mathrm{p}} + a_3 \frac{D^{\frac{2}{3}}}{A_{\mathrm{w}}} + a_4 \frac{B_{\mathrm{wl}}}{L_{\mathrm{wl}}} + a_5 \frac{L_{\mathrm{cb}}}{L_{\mathrm{cf}}} + a_6 \frac{B_{\mathrm{wl}}}{T_{\mathrm{c}}} + a_7 \cdot C_{\mathrm{mid}} \right) \frac{D^{\frac{1}{3}}}{L_{\mathrm{wl}}}. \qquad (23)$$

The coefficients of the regression depend on the hull's Froude number $F_n = V/\sqrt{gL_{\mathrm{wl}}}$. Table 6 provides these regression coefficients for several Froude values; again a Thin Plate Spline interpolation is employed to determine their values at other Froude numbers.

The wave resistance is corrected for the heel, using an added resistance R_h computed as (Larsson 1999)

$$R_{\mathrm{h}} = \frac{\pi}{180} 10^{-3} \left(6.647 \frac{T_{\mathrm{c}}}{2} + 2.517 \frac{B_{\mathrm{wl}}}{T_{\mathrm{c}}} + 3.710 \frac{B_{\mathrm{wl}}}{T_{\mathrm{t}}} \right) \theta. \qquad (24)$$

The stability of the hull is based on the simple transverse stability model at small angles proposed by Larsson (1999), with a righting moment given by

$$M_{\mathrm{stab}} = \rho_w g D G_{\mathrm{M}} \sin(\theta). \qquad (25)$$

The forces acting on the daggerboard are based on the standard Prandlt theory (Prandtl 1925), with lift and drag coefficients expressed as

$$C_l^{\mathrm{dagg}} = 2\pi\lambda, \qquad C_d^{\mathrm{dagg}} = 0.008 + \frac{\left(C_l^{\mathrm{dagg}}\right)^2}{0.8\,\pi\Lambda^{\mathrm{dagg}}}, \qquad (26)$$

with the leeway angle, λ, in rad.

With these expressions, the hydrodynamic driving force, side force and heeling moment acting on the yacht are computed as

$$F_x^h = R_{\mathrm{wav}} + R_{\mathrm{h}} + \frac{1}{2}\rho_w V^2 \left(S_{\mathrm{dagg}} C_d^{\mathrm{dagg}} + W_{\mathrm{SA}} C_f \right), \qquad (27)$$

$$F_y^h = \frac{1}{2}\rho_w S_{\mathrm{dagg}} V^2 C_l^{\mathrm{dagg}}, \qquad (28)$$

$$M_x^h = \tfrac{1}{2}\rho_w S_{\mathrm{dagg}} C_l^{\mathrm{dagg}} Z_{\mathrm{dagg}} + M_{\mathrm{stab}}. \qquad (29)$$

Table 6. Wave resistance coefficients with the Froude number.

F_n	**0.15**	**0.20**	**0.25**	**0.30**	**0.35**	**0.40**	**0.45**	**0.50**	**0.55**	**0.6**	**0.65**	**0.70**	**0.75**
a_0	-0.0005	-0.0003	-0.0002	-0.0009	-0.0026	-0.0064	-0.0218	-0.0388	-0.0347	-0.0361	0.0008	0.0108	0.1023
a_1	0.0023	0.0059	-0.0156	0.0016	-0.0567	-0.4034	-0.5261	-0.5986	-0.4764	0.0037	0.3728	-0.1238	0.7726
a_2	-0.0086	-0.0064	0.0031	0.0337	0.0446	-0.1250	-0.2945	-0.3038	-0.2361	-0.2960	-0.3667	-0.2026	0.5040
a_3	-0.0015	0.0070	-0.0021	-0.0285	-0.1091	0.0273	0.2485	0.6033	0.8726	0.9661	1.3957	1.1282	1.7867
a_4	0.0061	0.0014	-0.0070	-0.0367	-0.0707	-0.1341	-0.2428	-0.0430	0.4219	0.6123	1.0343	1.1836	2.1934
a_5	0.0010	0.0013	0.1048	0.0218	0.0914	0.3578	0.6293	0.8332	0.8990	0.7534	0.3230	0.49763	-1.5479
a_6	0.0001	0.0005	0.0010	0.0015	0.0021	0.0045	0.0081	0.0106	0.0096	0.0100	0.0072	0.0038	-0.0115
a_7	0.0052	-0.0020	-0.0043	-0.0172	-0.0078	0.1115	0.2086	0.1336	-0.2272	-0.3352	-0.4632	-0.4477	-0.0977