Drone Flight Time Estimation Under Epistemic Uncertainty

Conference Paper · January 2023

DOI: 10.3850/978-981-18-8071-1_P595-cd

5 authors, including:

Tathagata Basu
University of Strathclyde
23 PUBLICATIONS **7** CITATIONS
SEE PROFILE

Edoardo Patelli
University of Strathclyde
357 PUBLICATIONS **3,473** CITATIONS
SEE PROFILE

Massimiliano Vasile
University of Strathclyde
492 PUBLICATIONS **4,831** CITATIONS
SEE PROFILE

Drone flight time estimation under epistemic uncertainty

Tathagata Basu
Civil and Environmental Engineering, University of Strathclyde, UK. E-mail: tathagata.basu@strath.ac.uk

Gianluca Filippi
Mechanical and Aerospace Engineering, University of Strathclyde, UK. E-mail: g.filippi@strath.ac.uk

Edoardo Patelli
Civil and Environmental Engineering, University of Strathclyde, UK.

Marco Fossati
Mechanical and Aerospace Engineering, University of Strathclyde, UK.

Massimiliano Vasile
Mechanical and Aerospace Engineering, University of Strathclyde, UK.

Drone Logistic Network (or simply, DLN) is an emerging topic in the sector of transportation networks with applications in goods delivery, postal shipping, healthcare networks etc. It is a rather complex system which have different types of drones and ground facilities and it also requires a robust design of the network to ensure optimal time for delivery, efficiency, resilience, risk and cost efficiency along with different other optimizations of 'Key Performance Indicators'. Moreover, in sectors like healthcare networks, we need to be extra cautious whilst modeling the network as the consequence of failure is severe. Besides these, we also need to work with real-time telemetry data which can be very noisy at times. To deal with the above mentioned technicalities, we propose a robust surrogate modeling strategy through propagation of interval information from the observed data. We are interested in using this surrogate model to simulate contingency scenarios or simply to construct a Digital Twin (DT). For this particular contribution, we are specifically interested in estimating the drone flight time in uncertain conditions. With our proposed method, we obtain interval estimates for our quantities of interest, which can be interpreted as the set of possible values in between the optimistic and pessimistic bounds.

Keywords: Drone Logistic Network, Gaussian process regression, Epistemic Uncertainty, Interval Probability.

DLN Drone Logistic Network
DT Digital Twin
IGP Imprecise Gaussian Process
ImGP Interval-mean Gaussian Process
NIGP noisy input Gaussian process
VRP Vehicle Routing Problem

1. Introduction

In the recent years, we realized that a distributed healthcare network can improve the efficiency of a healthcare system significantly. Drone logistic network for delivering medical goods is one such application of distributed healthcare network. A trial for such network was performed near Rome by Leonardo and Telespazio (Li et al. (2020)), where the medical objects were delivered in 25 minutes by drone while the road journey along the coast took about 45-60 minutes. The effect of drone transportation on biological samples has also been analyzed by F. V. Daalen and Geerlings (2017) to investigate the benefits of a drone logistic system and it has been observed that there are no negative effects on the objects for a turn around time of less than 4 hours. Several other research works have also been carried out for Drone Logistic Network (DLN). Matternet (2020) announced a collaboration with lab facilities in Berlin to transport patients' samples from hospitals by drone; Amukele et al. (2016) investigated whether the medical specimens are affected by drone transport; Quintanilla García et al. (2021) successfully conducted flight tests for medical de-

Proceedings of the 33rd European Safety and Reliability Conference.
Edited by Mário P. Brito, Terje Aven, Piero Baraldi, Marko Čepin, Enrico Zio
 Published by Research Publishing, Singapore
ISBN: 981-973-0000-00-0 :: doi: 10.3850/981-973-0000-00-0_output

livery in Spain etc.

Following this direction, the UK government is currently focusing on the aspects of an autonomous DLN that will allow the delivering of medical equipment and assistance to remote areas. The CAELUS project, financed by the UK Industrial Strategy Future Flight Challenge Fund, has the aim of exploring the usage of drone delivery systems for dispatching of medical items. This paper presents part of the analyses done during the second phase of the project. The main objective is to create a digital blueprint - a combination of a DT models of the complex network and a set of optimization tools - of the DLN with a twofold applicability. To achieve a robust framework, we explore the possibility of using a interval valued surrogate modeling scheme for which we adapt the model proposed by Mangili (2016).

The first application of the digital blueprint corresponds to the design process of the whole DLN, which has to be optimal for the given key performance indicators as defined by the stakeholders. One such key performance indicator is to deliver the medical goods within a certain time interval. To do so, we need to estimate the time a drone of specific type takes. However, we notice that the flight time of a drone can be influenced by two major uncertain parameters: wind speed and wind direction. As a result the estimated time can be unreliable at times and we want to predict a reliable interval within which the flight time is supposed to lie.

As hinted earlier, we will adapt the notion Imprecise Gaussian Process (IGP) to quantify the associated uncertainty in the drone flight time. But using a complete prior ignorance can be restrictive for practical purposes, so we initiate our model by considering an interval for the location parameter of the Gaussian process with non-zero center. That is, we are interested in incorporating some prior information efficiently so that our posterior estimate of the Gaussian process is not too wide. This can be interpreted as an approximate Bayesian computation similar to noisy input Gaussian process (NIGP) proposed by McHutchon and Rasmussen (2011).

The rest of the paper is organized as follows, we will first give a short introduction to Gaussian process in Section 2, followed by our proposed method of Interval-mean Gaussian Process (ImGP) in Section 3. In this section, we provide different theoretical aspects of our method and give the posterior bounds for mean and variance of the ImGP. We give an algorithmic formulation to train ImGP in Section 4 and show our results with a synthetic dataset in Section 5. Finally, we illustrate our method for the drone flight estimation in Section 6 for three different types of drones and conclude this paper in Section 7.

2. Gaussian Process

Let y be an output and x be a p-dimensional input which satisfies the following functional relationship

$$y = f(x) + \epsilon \tag{1}$$

where ϵ is the associated noise following a normal distribution with mean 0 and variance σ^2. Our main objective is to estimate this unknown function f with the help of observational data and a natural choice for that is to use a Gaussian process prior for the unknown function f.

Let $\mu := \mu(x) : \mathbb{R}^p \to \mathbb{R}$ be a mean function and $k_\theta := k_\theta(x, x) : \mathbb{R}^p \to \mathbb{R}$ be a covariance function with hyper-parameter θ. Now, for a vector of n inputs $\mathbf{x}$ and corresponding noisy observation of outputs $\mathbf{y}$, we have

$$\mathbf{y} \sim GP\left(\tilde{\mu}, K + \sigma^2 \mathbf{I}_n\right) \tag{2}$$

where $\tilde{\mu} :=((\mu(\mathbf{x}_1), \ldots, \mu(\mathbf{x}_n))$ and K is an $n \times n$ matrix such that $[K]_{ij} = k_\theta(\mathbf{x}_i, \mathbf{x}_j)$.

Now for a new vector of m inputs, the joint distribution is given by

$$\begin{bmatrix} \mathbf{y} \\ f^* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \tilde{\mu} \\ \tilde{\mu}^* \end{bmatrix}, \begin{bmatrix} K + \sigma^2 \mathbf{I}_n & K^{*T} \\ K^* & K^{**} \end{bmatrix}\right) \tag{3}$$

where K^* is an $n \times m$ matrix such that $[K^*]_{ij} = k_\theta(\mathbf{x}_i^*, \mathbf{x}_j)$ and K^{**} is an $m \times m$ matrix such that $[K^{**}]_{ij} = k_\theta(\mathbf{x}_i^*, \mathbf{x}_j^*)$. Then, the posterior mean and variance of f^* is given by:

$$\hat{\mu}^* = \tilde{\mu}^* + K^* K_n^{-1}(\mathbf{y} - \tilde{\mu}) \tag{4}$$

$$\hat{K}^{**} = K^{**} - K^* K_n^{-1} K^{*T} \tag{5}$$

where $K_n = K + \sigma^2 \mathbf{I}_n$.

2.1. *Information about Input Variables*

Gaussian Process defined above is a well investigated method, especially, in the field of engineering where we often need to use surrogate methods to find approximate solutions of complex problems. However, in many cases the inputs can be heavily influenced by underlying uncertainty of system and quantifying that can be a rather challenging task.

McHutchon and Rasmussen (2011) proposed NIGP to deal with inputs under severe uncertainty with the assumption that these inputs follow a Gaussian distribution. In general, NIGP performs very well in reducing the variance of the Gaussian Process. However, this can be limiting, as in reality the noise of the input variable does not need to be in nature. More specifically, we may not have any distributional knowledge on the input variable. This motivates us to look into the possibility of using interval information for Gaussian process.

Let x be a p-dimensional input and y be the corresponding noisy observation of $f(x)$. Now, let $\underline{x}_j \leq x_j \leq \overline{x}_j$ for each $j = 1, \cdots, p$. Then, we have

$$y = f\left(x_1^c + x_1^r h, \cdots, x_p^c + x_p^r h\right) + \epsilon \tag{6}$$

where $x_j^c = \frac{\overline{x}_j + \underline{x}_j}{2}, 0 \leq x_j^r \leq \frac{\overline{x}_j - \underline{x}_j}{2}$ and $h = \pm 1$.

Now, by applying Taylor's series expansion, we have

$$\begin{aligned} &f\left(x_1^c + x_1^r h, \cdots, x_p^c + x_p^r h\right) \\ &= f(x^c) + h x^r \nabla f(x^c) + \cdots \end{aligned} \tag{7}$$

where $x^c{:=}(x_1^c, \ \ldots, \ x_p^c)$ and $x^r{:=}(x_1^r, \ \ldots, \ x_p^r)$. Clearly, this expansion is convergent when $\|x^r\| < 1$. For $\|x^r\| > 1$, we can assume it to be divergent to get rid of computational difficulties. In both the cases, we can approximate $f(x)$ as a combination of center and radius so that,

$$M_c = f(x^c) \tag{8}$$

and

$$r \in \mathcal{R} := \begin{cases} [0, x^r \nabla f(x^c)], & \|x^r\| < 1 \\ [0, \infty), & \|x^r\| \geq 1 \end{cases} \tag{9}$$

Note that, it is not possible to compute the exact values of M_c or r, as it involves the unknown function f. Instead, we follow the approach of McHutchon and Rasmussen (2011) and obtain a standard Gaussian process for the function f then we calculate the derivative of the mean. We can do that as the derivative process of a Gaussian process is also a Gaussian process as shown by Solak et al. (2002).

3. Interval-mean Gaussian Process

As hinted earlier, we are interested in using the interval information of the input variables. So, we exploit the notion of IGP Mangili (2016) to propose ImGP which allows us to vary the mean of the Gaussian process within a bounded interval. This particular modification of IGP allows us to incorporate prior information efficiently as well as learn from the data. This is particularly useful as in our Vehicle Routing Problem (VRP), majority of the quantities have a positive compact support and it is reasonable to use that information to reduce the imprecision.

Definition 3.1 (ImGP). *Given a base covariance kernel $k_\theta(x, x)$ and a constant $\Delta > 0$, we define the ImGP as the set of Gaussian processes, such that*

$$\mathcal{G}_\Delta = \left\{ GP\left(M_c + rh, k_\theta(x, x) + \frac{1 + r}{\Delta}\right) : h = \pm 1, r \in \mathcal{R} \right\} \tag{10}$$

where M_c is the center and $\mathcal{R}$ is the set of all possible values of the radius of the interval mean respectively.

Theorem 3.1. *Let $f(x)$ follows an ImGP as defined in Eq. 10 and let $\mathbf{k}_x = [k_\theta(x, x_1), \ \ldots, \ k_\theta(x, x_n)]^T$. Then the posterior of $f(x)$ is a Gaussian process with mean*

$$\hat{\mu}(x) = \mathbf{k}_x^T K_n^{-1} \left(\mathbf{y} - \hat{y}\mathbf{1}_n\right) + \hat{y} \tag{11}$$

and variance

$$\hat{k}(x,x) = k_\theta(x,x) - \mathbf{k}_x^T K_n^{-1}\mathbf{k}_x + \frac{(r+1)(1-\mathbf{s}_k^T\mathbf{k}_x)^T(1-\mathbf{s}_k^T\mathbf{k}_x)}{\Delta+(r+1)S_k} \tag{12}$$

where

$$\hat{y} = \frac{(r+1)\mathbf{s}_k^T\mathbf{y} + \Delta(M_c+rh)}{\Delta+(r+1)S_k}, \tag{13}$$

$\mathbf{s}_k = K_n^{-1}\mathbf{1}_n$, $S_k = \mathbf{1}_n^T K_n^{-1}\mathbf{1}_n$ *and* $\mathbf{1}_n$ *is the* n*-dimensional vectors with entries being equal to 1.*

Proof. The proof of the above theorem is similar to the one provided by Mangili (2016) and can be obtained very easily with simple algebraic manipulation. Therefore, we omit the proof for brevity of the space. □

Theorem 3.2. *Let* $r \in \mathcal{R} := [0,\overline{r})$. *Then the posterior bounds for mean and variance of ImGP is given by:*

$$\underline{\hat{\mu}}(x) = \mathbf{k}_x^T K_n^{-1}\mathbf{y} + \left(1-\mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta M_c + (\overline{r}+1)\mathbf{s}_k^T\mathbf{y}}{\Delta+(\overline{r}+1)S_k} - \left|1-\mathbf{k}_x^T\mathbf{s}_k\right|\frac{\Delta\overline{r}}{\Delta+(\overline{r}+1)S_k}; \tag{14}$$

$$\hat{\overline{\mu}}(x) = \mathbf{k}_x^T K_n^{-1}\mathbf{y} + \left(1-\mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta M_c + (\overline{r}+1)\mathbf{s}_k^T\mathbf{y}}{\Delta+(\overline{r}+1)S_k} + \left|1-\mathbf{k}_x^T\mathbf{s}_k\right|\frac{\Delta\overline{r}}{\Delta+(\overline{r}+1)S_k} \tag{15}$$

and

$$\underline{\hat{k}}(x,x) = k_\theta(x,x) - \mathbf{k}_x^T K_n^{-1}\mathbf{k}_x + \frac{(1-\mathbf{s}_k^T\mathbf{k}_x)^T(1-\mathbf{s}_k^T\mathbf{k}_x)}{\Delta+S_k} \tag{16}$$

$$\hat{\overline{k}}(x,x) = k_\theta(x,x) - \mathbf{k}_x^T K_n^{-1}\mathbf{k}_x + \frac{(1-\mathbf{s}_k^T\mathbf{k}_x)^T(1-\mathbf{s}_k^T\mathbf{k}_x)}{\frac{\Delta}{\overline{r}+1}+S_k}. \tag{17}$$

Note that $\mathcal{R}$ is defined as an open set as we may not have a finite radius. Even in that case, we can simply find the bounds very easily using the limiting case.

We would also like to mention here that these bounds are much simpler to deal with than the one given by Mangili (2016) for IGP. However, this can be slightly wider in some cases. However, from the practical point of view, our bounds are easier to implement we do not need to use intermediate conditions for finding tighter bounds.

Proof. To obtain the posterior bounds of the mean, we first simplify the expression of the mean. From (11) we have,

$$\begin{aligned}
&\hat{\mu}(x)\\
&= \mathbf{k}_x^T K_n^{-1}\left(\mathbf{y}-\hat{y}\mathbf{1}_n\right)+\hat{y} && (18)\\
&= \mathbf{k}_x^T K_n^{-1}\mathbf{y} + \left(1-\mathbf{k}_x^T K_n^{-1}\mathbf{1}_n\right)\hat{y} && (19)\\
&= \mathbf{k}_x^T K_n^{-1}\mathbf{y} + \left(1-\mathbf{k}_x^T\mathbf{s}_k\right)\hat{y} && (20)
\end{aligned}$$

Now, since $\hat{y}$ can be written as

$$\hat{y} = \frac{\frac{\Delta(M_c+rh)}{r+1}+\mathbf{s}_k^T\mathbf{y}}{\frac{\Delta}{r+1}+S_k} \tag{21}$$

we get

$$\frac{\frac{\Delta(M_c-\overline{r})}{\overline{r}+1}+\mathbf{s}_k^T\mathbf{y}}{\frac{\Delta}{\overline{r}+1}+S_k} \le \hat{y} \le \frac{\frac{\Delta(M_c+\overline{r})}{\overline{r}+1}+\mathbf{s}_k^T\mathbf{y}}{\frac{\Delta}{\overline{r}+1}+S_k}. \tag{22}$$

Therefore, we need to check two different conditions based on the sign of $\left(1-\mathbf{k}_x^T\mathbf{s}_k\right)$.

a) $\left(1-\mathbf{k}_x^T\mathbf{s}_k\right) > 0$: In this case, the posterior mean is increasing w.r.t $\hat{y}$. That is the lower bound of the posterior mean is obtained for the lower bound of $\hat{y}$. Therefore the lower bound is given by:

$$\begin{aligned}
\underline{\hat{\mu}}(x) &= \mathbf{k}_x^T K_n^{-1}\mathbf{y} + \left(1-\mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta(M_c-\overline{r})+(\overline{r}+1)\mathbf{s}_k^T\mathbf{y}}{\Delta+(\overline{r}+1)S_k} && (23)\\
&= \mathbf{k}_x^T K_n^{-1}\mathbf{y} + \left(1-\mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta M_c+(\overline{r}+1)\mathbf{s}_k^T\mathbf{y}}{\Delta+(\overline{r}+1)S_k} - \left(1-\mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta\overline{r}}{\Delta+(\overline{r}+1)S_k} && (24)\\
&= \mathbf{k}_x^T K_n^{-1}\mathbf{y} + \left(1-\mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta M_c+(\overline{r}+1)\mathbf{s}_k^T\mathbf{y}}{\Delta+(\overline{r}+1)S_k} - \left|1-\mathbf{k}_x^T\mathbf{s}_k\right|\frac{\Delta\overline{r}}{\Delta+(\overline{r}+1)S_k}. && (25)
\end{aligned}$$

Similarly the upper bound is given by:

$$\begin{aligned}\hat{\bar{\mu}}(x) &= \mathbf{k}_x^T K_n^{-1}\mathbf{y} \\ &+ \left(1 - \mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta M_c + (\bar{r}+1)\mathbf{s}_k^T\mathbf{y}}{\Delta + (\bar{r}+1)S_k} \\ &+ \left|1 - \mathbf{k}_x^T\mathbf{s}_k\right|\frac{\Delta\bar{r}}{\Delta + (\bar{r}+1)S_k}. \end{aligned} \tag{26}$$

b) $\left(1 - \mathbf{k}_x^T\mathbf{s}_k\right) < 0$: In this case, the posterior mean is decreasing w.r.t $\hat{y}$. That is the lower bound is obtained for the upper bound of $\hat{y}$ and vice versa. Therefore proceeding like before, we get the following lower bound:

$$\begin{aligned}\hat{\underline{\mu}}(x) &= \mathbf{k}_x^T K_n^{-1}\mathbf{y} \\ &+ \left(1 - \mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta M_c + (\bar{r}+1)\mathbf{s}_k^T\mathbf{y}}{\Delta + (\bar{r}+1)S_k} \\ &+ \left(1 - \mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta\bar{r}}{\Delta + (\bar{r}+1)S_k} \end{aligned} \tag{27}$$

$$\begin{aligned}&= \mathbf{k}_x^T K_n^{-1}\mathbf{y} \\ &+ \left(1 - \mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta M_c + (\bar{r}+1)\mathbf{s}_k^T\mathbf{y}}{\Delta + (\bar{r}+1)S_k} \\ &- \left|1 - \mathbf{k}_x^T\mathbf{s}_k\right|\frac{\Delta\bar{r}}{\Delta + (\bar{r}+1)S_k}. \end{aligned} \tag{28}$$

Similarly the upper bound is given by:

$$\begin{aligned}\hat{\bar{\mu}}(x) &= \mathbf{k}_x^T K_n^{-1}\mathbf{y} \\ &+ \left(1 - \mathbf{k}_x^T\mathbf{s}_k\right)\frac{\Delta M_c + (\bar{r}+1)\mathbf{s}_k^T\mathbf{y}}{\Delta + (\bar{r}+1)S_k} \\ &+ \left|1 - \mathbf{k}_x^T\mathbf{s}_k\right|\frac{\Delta\bar{r}}{\Delta + (\bar{r}+1)S_k}. \end{aligned} \tag{29}$$

For the bounds of the posterior variance we need to check the term containing the imprecision factor Δ. From (12) we have

$$\begin{aligned}&\frac{(r+1)(1 - \mathbf{s}_k^T\mathbf{k}_x)^T(1 - \mathbf{s}_k^T\mathbf{k}_x)}{\Delta + (r+1)S_k} \\ &= \frac{(1 - \mathbf{s}_k^T\mathbf{k}_x)^T(1 - \mathbf{s}_k^T\mathbf{k}_x)}{\frac{\Delta}{r+1} + S_k}. \end{aligned} \tag{30}$$

Since $(1 - \mathbf{s}_k^T\mathbf{k}_x)^T(1 - \mathbf{s}_k^T\mathbf{k}_x) > 0$ the posterior variance is monotonically increasing w.r.t r. Therefore,

$$\begin{aligned}\hat{\underline{k}}(x,x) &= k_\theta(x,x) - \mathbf{k}_x^T K_n^{-1}\mathbf{k}_x \\ &+ \frac{(1 - \mathbf{s}_k^T\mathbf{k}_x)^T(1 - \mathbf{s}_k^T\mathbf{k}_x)}{\Delta + S_k} \end{aligned} \tag{31}$$

$$\begin{aligned}\hat{\bar{k}}(x,x) &= k_\theta(x,x) - \mathbf{k}_x^T K_n^{-1}\mathbf{k}_x \\ &+ \frac{(1 - \mathbf{s}_k^T\mathbf{k}_x)^T(1 - \mathbf{s}_k^T\mathbf{k}_x)}{\frac{\Delta}{r+1} + S_k} \end{aligned} \tag{32}$$

□

4. Training of ImGP

As mentioned earlier, we incorporate a bi-level approach similar to NIGP by McHutchon and Rasmussen (2011) to train our ImGP. In the primary level, we use the training data to obtain a standard Gaussian process. We then compute the posterior mean and its derivative at x^c and construct the interval of our ImGP.

Finally, we use the interval mean obtained by the primary level to initialize our ImGP. We train our model with the dataset and optimize the hyperparameter of the ImGP w.r.t. suitable loss function. We summarize this method in Algorithm 1.

In general, it might appear that we are using the data twice. However, the data is only being used for obtaining approximate bound for the prior mean of the Gaussian process. Therefore, it can be argued that this approach is somewhat similar to empirical Bayes methods.

5. Simulation Studies

For the simulation studies with synthetic dataset, we consider the following function

$$y = \sin(x) + \cos^2(x) + x^3 + \epsilon \tag{33}$$

where ϵ is a random noise following a normal distribution with mean 0 and variance 1. To generate the training dataset x, we sample 20 observations from a normal distribution with mean 0 and variance 3.

To perform our analyses, we consider three different levels of imprecision: $0.1, 1$ and 10. We show our analyses in Figure 1. In the figure, the green curves are used to show the bounds of the posterior mean; the red curves are used to show confidence region corresponding to the upper bound of the covariance; the black curves are used to show the exact functional value at the

Algorithm 1 Interval-mean Gaussian Process

(1) Fit standard GP (mean 0) with the training dataset $[\mathbf{y}, \mathbf{x}]$ and obtain the posterior mean given by:

$$\hat{\mu}(x) = \mathbf{k}_x \left(K + \sigma^2 \mathbf{I}_n\right)^{-1} \mathbf{y}$$

(2) Optimize the hyperparameter of the kernel function using log marginal likelihood.

(3) Find the derivative of the posterior mean at x^c

$$\nabla\hat{\mu}(x) \mid_{x^c} = \nabla\mathbf{k}_x \mid_{x^c} \left(K + \sigma^2 \mathbf{I}_n\right)^{-1} \mathbf{y}$$

(4) Compute the mean and radius of the location parameter of ImGP

(a)

$$M_c = f(x^c) \approx \hat{\mu}(x^c)$$

(b)

$$r \in \mathcal{R} \approx \begin{cases} \left[0, x^r \nabla\hat{\mu}(x) \mid_{x^c}\right], & \|x^r\| < 1 \\ [0, \infty), & \|x^r\| > 1 \end{cases}$$

(5) Fit ImGP with interval mean specified by center M_c and radius r.

test points. The black circles show the training points of the ImGP.

Clearly, as we should expect, for smaller values of Δ ie. the level of imprecision, the bounds are close to each other. As we increase the value of Δ, the bounds become wider. We also notice that for the first case ($\Delta = 0.1$), the exact functional value at test points is not contained within the confidence region for higher values of x. However, as we increase the level of imprecision in the other two cases, this functional value remains within the confidence region. This leads to the question of choosing the level of imprecision of ImGP. For instance, we can consider a weighted utility average of the lower and upper bounds of the posterior mean and use that to compute the root mean squared error. This way, we can get a single value for the posterior estimate and use method like cross-validation to tune the parameter Δ. In a more generalized setting, we can also use bi-level cross-validation with the mixing weight of the posterior bounds.

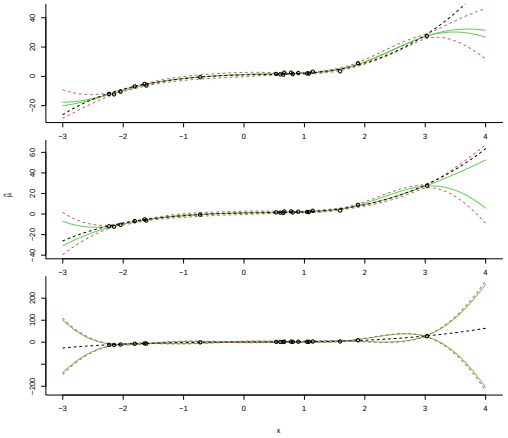

Fig. 1. Prediction using ImGP for synthetic dataset with different levels of imprecision: $\Delta = 0.1$ (top), $\Delta = 1$ (middle) and $\Delta = 10$ (bottom).

6. Estimating Drone Flight Time with ImGP

For the DLN, we are currently using three different times of drone. We can use a high fidelity model of the drone flight to calculate the flight time between point 'A' and point 'B'. However, this high fidelity model is complex and computationally expensive. As a result, using this high fidelity model is nearly impossible for the purpose of the optimization of the DLN. Instead we use telemetry data to measure the different modeling variable of the flight time and construct a surrogate model for optimization. Specifically, there are 5 different variables, which are most important in computing the drone flight time: distance between 'A' and 'B', maximum velocity of the drone, wind speed, wind direction and payload. The last three variables are uncertain beyond the measurement noise. In Figure 2, we show these variables from the dataset with the first drone type. We can notice that they are spread in the entire region and therefore having a normality assumption can lead to unreliable estimates. Instead we work with the support of the variable. For instance, for wind direction it lies between $[0, \pi]$ and payload it lies between $[0, 5]$. For the wind speed we rely on the observational data.

Similar to our analyses with the synthetic data, we consider three different levels of imprecision

to perform our analysis, which 0.01, 0.1 and 1. We also use similar color scheme to illustrate the bounds of the posterior mean and the confidence bounds. Clearly, we do not have exact functional expression anymore to illustrate the goodness of fit. Instead we use black circles to show the value at test points.

We notice that for the first two types of the drones the result are somewhat similar in terms of containing the observed flight time within the confidence region for different values of Δ. For the third drone type, however, we notice that the associated uncertainty is extremely high and we fail to contain one of the points even for $\Delta = 1$. It can also be seen that for higher values of Δ, the lower bound of the posterior mean is usually negative. Of course, in reality, it is not possible. The negative values of the lower bound only represent the uncertainty in data. In practice, we can assume that the values can be very close to 0 and simply truncate the estimates at 0.

In general, we can report these intervals to represent the epistemic uncertainty associated with the data. But in many cases, we need a single estimate to perform additional tasks. For that, we can use a weight parameter and obtain a weighted estimate from this interval so that

$$\hat{\mu}_\omega(x) = \omega\underline{\hat{\mu}}(x) + (1-\omega)\overline{\hat{\mu}}(x). \qquad (34)$$

This way, we can interpret this weight parameter as a degree of confidence. Clearly, when the degree of confidence is zero, we get $\hat{\mu}_\omega(x) = \overline{\hat{\mu}}(x)$. Therefore, $\hat{\mu}_\omega(x)$ gives the maximum value for the expected time taken by a drone or simply the worst-case scenario.

7. Conclusion and Future Work

In this paper, we discussed a possibility of using an ImGP as a robust surrogate model for problems where the modeling variables highly uncertain in nature. We provide a simple training algorithm for this method and showed their benefits using a synthetic dataset. More importantly, we use this method to illustrate the problem of estimating the drone flight time in a DLN for healthcare purpose and we can use these bounds to plan drone scheduling for delivering medical goods. Besides

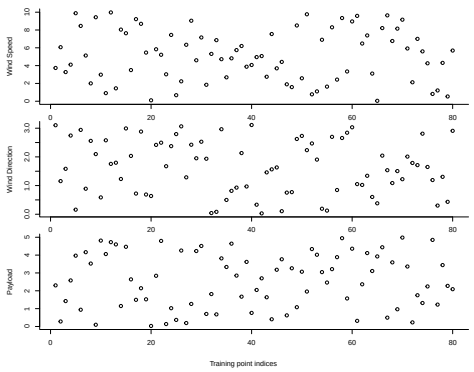

Fig. 2. Training points associated with the wind speed (top), wind direction (middle) and payload (bottom).

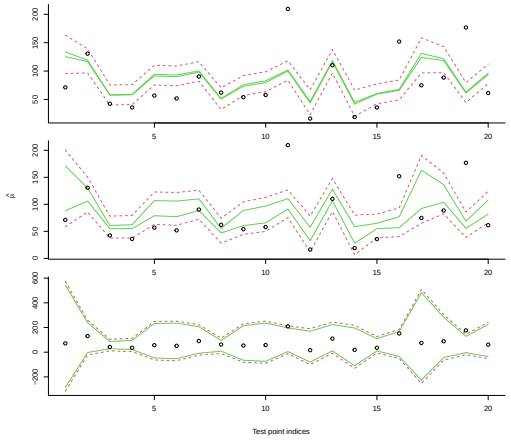

Fig. 3. Prediction using ImGP for drone type 1 with $\Delta = 0.01$ (top), $\Delta = 0.1$ (middle) and $\Delta = 1$ (bottom).

the practical aspects of ImGP, we also provide an alternative proof for obtaining the posterior means which simplifies the expressions of the posterior mean and removes several intermediate conditions suggested by Mangili (2016). This certainly, makes the posterior bound a little wider than the IGP. However, it reduces the computational costs and makes it easier to work with.

Currently we are only relying on the location parameter to express our lack of knowledge on the distributional properties of the modeling variables. Ideally, we should also use the variance term to represent our lack of knowledge and correct the final variance term accordingly. For

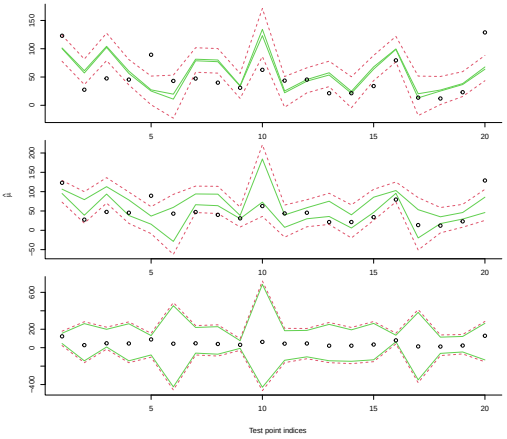

Fig. 4. Prediction using ImGP for drone type 2 with $\Delta = 0.01$ (top), $\Delta = 0.1$ (middle) and $\Delta = 1$ (bottom).

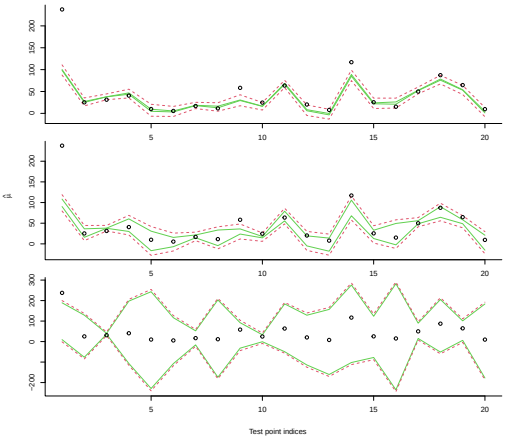

Fig. 5. Prediction using ImGP for drone type 3 with $\Delta = 0.01$ (top), $\Delta = 0.1$ (middle) and $\Delta = 1$ (bottom).

instance, in NIGP, McHutchon and Rasmussen (2011) use normality assumption to correct the variance term by using the variance of the input variables. This remains the missing block of the puzzle. Moreover, in many cases, we may need to consider the data in intervals. Especially, when multiple telemetries are present in the same location, their readings can be different. In such cases, we need to formulate the problem in a way where we can use interval data to train the ImGP.

Acknowledgement

This work is funded by the project CAELUS, UK Industrial Strategy Future Flight Challenge Fund.

References

Amukele, T. K., J. Street, K. Carroll, H. Miller, and S. X. Zhang (2016). Drone transport of microbes in blood and sputum laboratory specimens. *Journal of Clinical Microbiology 54*, 2622–2625.

F. V. Daalen, F. Holleman, C. V. and S. E. Geerlings (2017). Optimizing the process of blood culture collection. *European Congress of Clinical Microbiology and Infectious Diseases (ECCMID)*.

Li, G., Z. Liu, J. Hu, and H.-J. Li (2020). Leonardo, telespazio and bambino gesu' children's hospital test the use of drones for biomedical material delivery.

Mangili, F. (2016). A prior near-ignorance gaussian process model for nonparametric regression. *International Journal of Approximate Reasoning 78*, 153–171.

Matternet (2020). Matternet launches drone delivery operations at labor berlin in germany.

McHutchon, A. and C. E. Rasmussen (2011). Gaussian process training with input noise. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, NIPS'11, Red Hook, NY, USA, pp. 1341–1349. Curran Associates Inc.

Quintanilla García, I., N. Vera Vélez, P. Alcaraz Martínez, J. Vidal Ull, and B. Fernández Gallo (2021). A quickly deployed and uas-based logistics network for delivery of critical medical goods during healthcare system stress periods: A real use case in valencia (spain). *Drones 5*(1).

Solak, E., R. Murray-smith, W. Leithead, D. Leith, and C. Rasmussen (2002). Derivative observations in gaussian process models of dynamic systems. In S. Becker, S. Thrun, and K. Obermayer (Eds.), *Advances in Neural Information Processing Systems*, Volume 15. MIT Press.