Gaussian Process for Surrogate Modelling: Application in Non-Linear Mechanics

(Final report)

Tianyi LI[*], Joseph MORLIER[†], Simone CONIGLIO[‡] and Pierre-jean BARJHUX[§]
[*]Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO), Université de Toulouse, 31055 Toulouse, FRANCE
Email: Tianyi.LI@student.isae-supaero.fr
[†]Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO), Université de Toulouse, 31055 Toulouse, FRANCE
Email: joseph.morlier@isae-supaero.fr
[‡]Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO), Université de Toulouse, 31055 Toulouse, FRANCE
Email: simone.coniglio91@gmail.com
[§]Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO), Université de Toulouse, 31055 Toulouse, FRANCE
Email: pierre-jean.barjhoux@irt-saintexupery.com

***Abstract*—In engineering design problems, advanced simulations are time consuming and computationally expensive. As a cheap replacement of the expensive simulations, surrogate models have been widely accepted as an efficient tool of computational engineering design. Gaussian Process Regression (GPR) method is used in this article as a replacement of Finite Element Simulation to predict the maximum principle stress and its location of a non-linear model. The results show that GPR method can give accurate prediction of maximum principle stress and its location of a nonlinear model.**

Keywords: Gaussian Process Regression; Surrogate Modelling; Non-Linear Mechanics; Finite Element Method; Maximum Principle Stress

I. Introduction

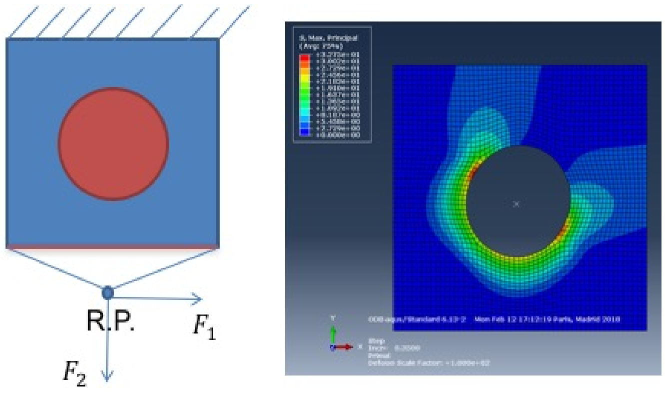
A. Motivation

Engineering design requires engineers to precise decisions out of numerous analysis. These engaged analysis could spend month time by dedicated teams. The 'Surrogate model' approach is proposed as an approximate model that can greatly reduce both time and cost in engineering design problems [1].

Using part of the results obtained from simulations or experiments as a training set, a surrogate model is trained and built. The established model is then optimized and tested to find the best possible function explaining the data. Once the surrogate model is well established, it can be used to predict the output of new data.

B. Problem statement

This project aims to construct a surrogate model to predict maximum principle stress and its location of a non-linear model. As shown in Figure 1, the life cycle of a simple non-linear component can be reduced to a linear combination of loads,F_1,F_2. The model is non linear due to contact with the red pin in the middle of the structure. Each load combination should be executed to analyze the life cycle due to the non-linearity. Since the Finite Element Analysis can be very expensive and the number of load combination can be of 5000 different loads, this approach can be prohibitive. As a replacement of finite element simulation, surrogate models are constructed so as to bridge the gap between accuracy and complexity.



a. Non-linear Model

b. Principal Stress Distribution

Fig. 1. Non-linear model of a small aircraft component

II. Methods

The type of surrogate model used in this project is Gaussain Process Regression (also called Kriging model), which is a subset of the Bayesian Inference algorithms. Before going into the principle of Gaussian Process, it is necessary to have a look at the Bayesian linear regression algorithm [2].

A. Bayesian Linear Regression

Suppose that we have already got a training set with observations (or outputs): $y = (y(x_1), ..., y(x_i), ..., y(x_N))^T$ evaluated at inputs $X = (x_1, ..., x_i, ..., x_N)^T$. Our goal is to predict $y(x_*)$ at a test input x_*. The process of Bayesian linear regression consists of three steps: assuming a functional form of f (representation), minimizing the error between the observed outputs (y) and the predicted outputs f(x)(evaluation) and optimizing parameters of f (optimization). The function is written in terms of linear basis functions $\phi(x) = \{1, x\}^T$.

$$f(x) = \{1 \quad x\}\{w_0 \quad w_1\}^T \tag{1}$$

w_0,w_1 are the parameters of the function. The above equations can be combined to result in the likelihood $Pr[y|X, w]$.

$$Pr[y|X, w] = \mathcal{N}(X^T\omega, \sigma_n^2 I_N) \tag{2}$$

$Pr[y|X, w]$ is probability distribution of the outputs y at the inputs X. The $\mathcal{N}[\mu, \Sigma]$ symbolizes a multi-variate Gaussian distribution with mean vector μ and covariance matrix σ.

A prior is a probability distribution before looking at the observations. Since uncertainty about the mean function can be taken into account by adding an extra term to the covariance function, a zero mean Gaussian prior was put on weights.

$$Pr[w] = \mathcal{N}(0, \sigma_{prior}) \tag{3}$$

Once we have the prior distribution, Bayes rule was used to get a posterior distribution of parameters. Combining the equation 1, 2 and 3, the posterior distribution of weights can be derived:

$$Pr[w|y, X] = \mathcal{N}(\frac{1}{\delta_n^2}A^{-1}Xy, A^{-1}) \tag{4}$$

In above equation, $A = \delta_n^{-2}XX^T + \sigma_p rior^{-2}$. Then we can get the posterior distribution for function f at test point x_*:

$$Pr[f|x_*, X, y] = \mathcal{N}(\frac{1}{\delta_n^2}x_*A^{-1}Xy, x_*^T A^{-1}x_*) \tag{5}$$

It is worth noting that in equation 5, the mean $\frac{1}{\delta_n^2}x_*A^{-1}Xy$ can be used as a prediction at the test point x_* and variance $x_*^T A^{-1}x_*$ can be the confidence interval for the prediction [3].

B. Gaussian Process Regression Principle

Gaussian Process is a distribution over functions. Compared with Bayesian linear regression, GPs enable us to encode prior knowledge directly in the functional space.

A GP model can be completely parameterized by its mean and its covariance function before introducing training data. The mean function represents the trend of functions, while the covariance function defines the spatial dependency between input-points [4] [5].

$$Pr[f(x)] = GP(m(x), k(x_1, x_2)) \tag{6}$$

$Pr(f(x))$ is probability distribution of function f at point x. The mean function m(x) is often assumed to be zero because uncertainty about the mean function can be taken into account by adding an extra term to the covariance function. In this way, the GP model can be completely parametrized by the covariance function. Therefor GP learning problem is exactly the problem of finding proper properties of the covariance function.

Once the prior is well defined, we wish to make use of the information of the training data-set. Same as Bayesian rules, a posterior is defined as the probability distribution after updating information of training data into prior distribution.

The measured outputs with noise can be written as:

$$y(x) = f(x) + \varepsilon \tag{7}$$

In equation 7, ε represents an independent random noise sampled from $\mathcal{N}(0, \sigma_n^2)$. We can thus obtain its prior GP:

$$Pr[y(x)] = GP(0, k(x_1, x_2) + \sigma_n^2\delta_{x_1,x_2}) \tag{8}$$

δ_{x_1,x_2} is a Kronecker delta function. According to above equations, the joint distribution of the training outputs $y(X)$ and true value $f(X)$ is shown below:

$$Pr\begin{bmatrix} y(x) \\ f(x_*) \end{bmatrix} = \mathcal{N}(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K(X,X) + \sigma_n^2 I_N & K(X, X_*) \\ K(X_*, X) & K(X_*, X_*) \end{bmatrix}) \tag{9}$$

The posterior distribution of $f(X_*)$ can be calculated as follows:

$$\begin{aligned} &Pr[f(x_*)|X_*, X, y(X)] = \\ &\quad GP(K(X_*, X)(K(X,X) + \sigma_n^2 I_N)^{-1}f(X), \\ &K(X_*, X_*) - K(X, X_*)(K(X,X) + \sigma_n^2 I_N)^{-1}K(X, X_*)) \end{aligned} \tag{10}$$

We can see from eqution 10 that the predictive mean is a linear combination of the observations and the predicted variance in not dependent on the observations y. Same as Bayesian rules, the mean here is a prediction at the test point and predicted variance can be used as an efficient method to track uncertainties arising from the prior assumption and observations [6].

C. Covariance Functions

As we mentioned before, the mean function is zero. Therefor the properties of functions under a GP are controlled by the functional form of the covariance and its hyper-parameters [7]. In order to give a accurate prediction, functional form selection and hyper-parameters optimization are very important.

1) Covariance Functions: As most common used covariance functions, SE(squared exponentia) kernel, Matérn kernel and RQ(rational quadratic) kernel are used in this project.

Squared Exponential Kernel:

$$k_{SE}(d, \theta) = \theta_{amplitude}^2 exp(-\frac{d^2}{2\theta_{lengthscale}}) \tag{11}$$

Matérn Kernel:

$$\begin{aligned} &k_{Matern}(\upsilon = \frac{3}{2}, d, \theta) = \\ &\quad \theta_{amplitude}^2(1 + \frac{\sqrt{3}d}{\theta_{lengthscale}})exp(\frac{\sqrt{3}d}{\theta_{lengthscale}}) \end{aligned} \tag{12}$$

Rational Quadratic Kernel:

$$k_{RQ}(\alpha, k, d) = (1 + \frac{d^2}{2\alpha k^2})^{-\alpha} \tag{13}$$

2) Hyperparameters: We can see from above equations that these covariance functions are specified by a set of parameters. In GP, they are called hyper-parameters. In this project, optimal hyper-parameters are found by calculating and maximizing the marginal likelihood [3]:

$$log(Pr[y|X,\theta]) = -\frac{1}{2}y^T[K(X,X)+\sigma_n^2 I_N]^{-1}y \\ - log|K(X,X)+\sigma_n^2 I_N| - \frac{N}{2}log(2\pi) \quad (14)$$

There are three terms in marginal likelihood [7]: a data-fit term $-\frac{1}{2}y^T[K(X,X)+\sigma_n^2 I_N]^{-1}y$, a model complexity term $-log|K(X,X)+\sigma_n^2 I_N|$, and a normalization term $-\frac{N}{2}log(2\pi)$.

The optimal hyper-parameters we get from maximizing the marginal likelihood are the local maximums. In order to obtain global optimum, Leave-One-Out-Cross-Validation will be used in this project [8] [9].

D. Error Analysis Method

1) Leave-One-Out-Cross-Validation: As mentioned above, Leave-One-Out-Cross-Validation is used to find the global optimum of hyper-parameters. The main idea of CV is to split the experimental design set into two disjoint sets, one is used for training and the other one is used to testing the surrogate model. Leave-One-Out (LOO) is a special case of CV where test sets are obtained by removing one observation at-a-time.

2) Error Measurement Method: In order to test the performance of the established surrogate model, we need to measure the error between the observed outputs and the predicted outputs. Root-mean-square Error($RMSE$) and R^2 are used in this manuscript.

Root-mean-square Error:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{n}} \quad (15)$$

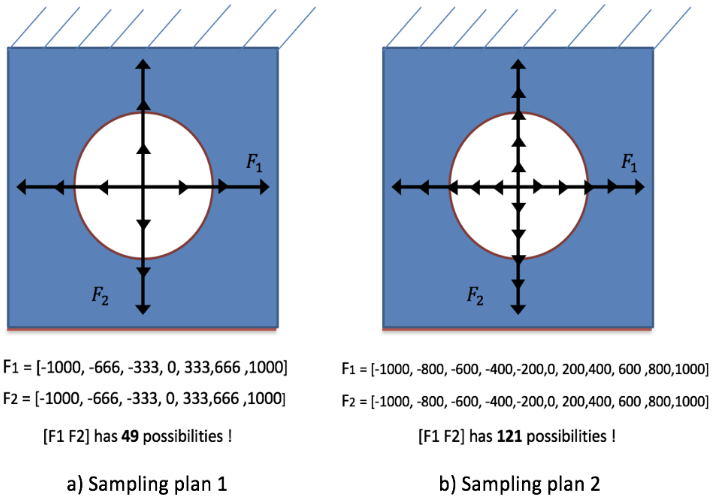
R^2 Error:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - \overline{Y}_i)^2} \quad (16)$$

In above equations, Y_i is observed outputs, $\hat{Y}_i$ is predicted output and $\overline{Y}_i$ is the mean of observed outputs. It is worth noting that our prediction is more accurate when the $RMSE$ is closer to zero. For R^2, the closer its value is to one, the more accurate our prediction is.

III. Modeling process

GPR models are first constructed to predict the maximum principle stress for each load combination. We then wish to give the prediction of the location of maximum stress. All the code is written in the basis of the code library provided in the book [7] of C.E.Rasmussen



F1 = [-1000, -666, -333, 0, 333,666 ,1000]
F2 = [-1000, -666, -333, 0, 333,666 ,1000]
[F1 F2] has **49** possibilities !

a) Sampling plan 1

F1 = [-1000, -800, -600, -400,-200,0, 200,400, 600 ,800,1000]
F2 = [-1000, -800, -600, -400,-200,0, 200,400, 600 ,800,1000]
[F1 F2] has **121** possibilities !

b) Sampling plan 2

Fig. 2. Sampling plan

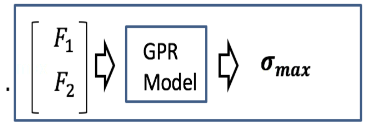


Fig. 3. Input-Output for Maximum Prediction

A. Sampling Design

As shown in Figure 2, two different sampling plans are used in this project.

For sampling plan 1, both $F1$ and $F2$ have seven different values being evenly taken from $[-1000, 1000]$. So there are 49 different load combination in total. We then obtain the maximum principle stress and its location for each load by ABAQUS simulation. Sampling plan 2 are obtained by evenly taking 11 values for both $F1$ and $F2$. Same as plan 1, their maximum principle stress and locations of maximums are also simulated by ABAQUS.

B. Maximum Principle Stress

1) Hyper-parameter Optimization: Only the sampling plan 1 is used in predicting maximum stress. In order to build our surrogate model, we first determine inputs and outputs in the problem of predicting the maximum principle stress. As shown in Figure 3, this is a multiple inputs(F_1 and F_2) with single output(σ_{max}) GPR problem. We first select an empty mean function and a SE covariance function as prior information. Before predicting in the new data, we need to find the optimal hyper-parameters. As mentioned before, LOO-Cross-Validation is used here to find the global optimum.

2) Prediction in New Data: Once our surrogate model is well constructed and optimized, we wish to predict the maximum principle stress for new load combinations. 49 different loads and their corresponding maximum stress in sampling plan 1 are used as training set. 14 randomly generated loads

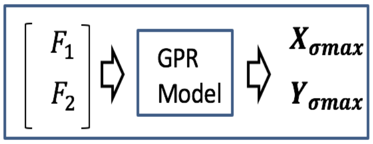


Fig. 4. Input-Output for Maximum Prediction

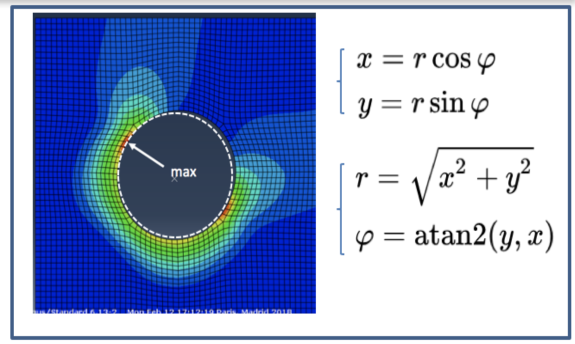


Fig. 5. Location of Maximum Stress

and their corresponding maximum stress are used as testing set.

3) Comparison Between Different Covariance Functions: The prediction of maximum stress for new load combinations are also given by using RQ and Matérn covariance function. Different surrogate models are constructed using different covariance functions.

C. Location of Maximum Principle Stress

Unlike the prediction of maximum principle stress, the GPR learning of its location is a multiple-output problem(shown in figure 4).

1) Training X Coordinate and Y Coordinate Separately: Without considering the relationship between the x and y coordinates, we first divide this problem into two single-output training problem. The x and y coordinates are trained as output separately and two surrogates models are constructed.

2) Training in Polar Coordinate: As loads are applied on the center of the model, the maximum stress occur in the inner circle(Figure 5). In this way, this problem could be simplified into a single output problem by changing Cartesian coordinate(x,y) into Polar coordinate(ϕ,r=cste).

3) Training with 121 training samples : Finally, Sampling plan 2 with more sample points is used as training set to predict the position of the maximum stress.

IV. Results

A. Maximum Principle Stress Prediction

1) Hyper-parameter Optimization: Before predicting maximum stress for new loads, LOO-Cross-Validation is used to get the global optimal hyper-parameters. The optimum hyper-parameters found in SE kernel are $[d = 7.17 \quad \theta = 3.50]$ (equation 11).

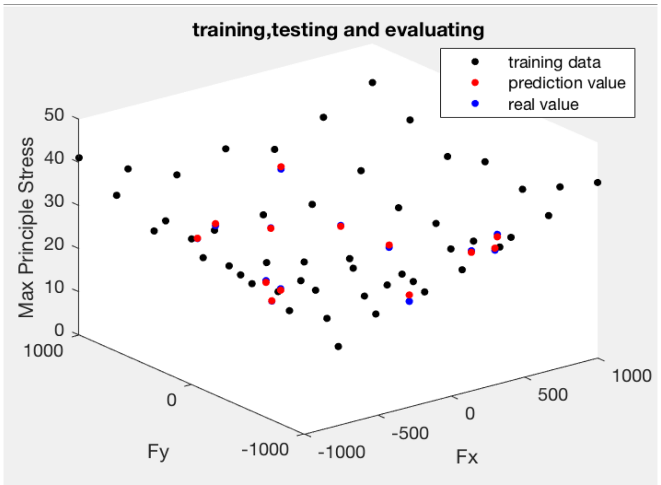


Fig. 6. Evaluation of GPR Model

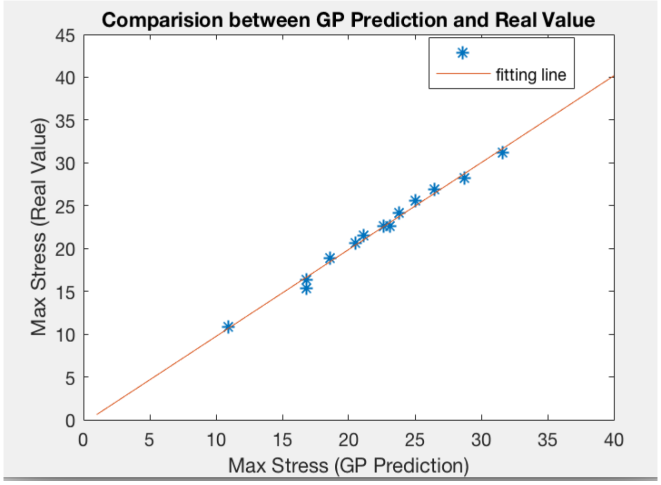


Fig. 7. Fitted line

2) Prediction in New Data: Once our surrogate model is well constructed, the predictive accuracy is tested.

In figure 6, x-axis and y-axis represent inputs F_1 and F_2, respectively. The z-axis represents the maximum principle stress. The black dots represent the training data, the red dots represent the predicted output, and the blue dots represent the observed output. We can see from Figure 6 that the red dots and blue dots are either very close or coincident. Therefore, intuitively our GP model can give accurate predictions for maximum principle stress.

As shown in figure 7, the x-axis represents the predicted output and the y-axis represents the observed output. Therefore, the position of points on the graph can represent the difference between the observed value and the predicted value. When the points are distributed along the line $y = x$, it means that the predictions are close to the observations. However, when the points are distributed around the x-axis and y-axis, our GPR model fails to give accurate predictions.

A straight line is fitted according to the points in the figure

7. The equation for this line is $y = 1.01x - 0.41$. As its slope is approximately equal to 1 and the intercept is close to 0, it can be assumed that the prediction results are close to the observations. In addition, the RMSE between the predicted value and the true value is 0.5431, which also indicates that our GP model has a very high prediction accuracy.

3) Comparison Between Different Covariance Functions: SE kernel, RQ kernel, Matérn kernel and their combinations are also used as covariance functions to predict the maximum stress. We wish to explore the influence of the covariance function on the prediction results. The results are shown in table 1. RMSE and R^2 error are calculated for each covariance function. At the same time, the corresponding code running time is also given. The code running time is taken as the average of ten runs and computations were done on a Intel Core i5-2410M CPU 3.1 GHz.

We first focus on three simple covariance function. It is not difficult to see that their corresponding learning time is almost the same, while Matérn kernel has the highest prediction accuracy, with a RMSE of 0.6385 and a R^2 of 0.9855. On the whole, the combined covariance functions have higher prediction accuracy than single kernel functions, but their running time also increases. The combination of SE, RQ and Matérn has the most accurate prediction and the longest running time. Therefore, if there are only tens of samples in training set, the combination of SE, RQ and Matérn can be used to get the best performance. But when there are hundreds of samples in training set, it is more appropriate to use a simpler covariance function to improve training efficiency.

B. Maximum Principle Stress Location Prediction

After predicting the maximum stress, we then construct surrogate models to predict its location.

1) Training X Coordinate and Y Coordinate Separately: The X coordinate and Y coordinate are first trained as output separately. Same principle as figure 7, Figures 8 and 9 show the fitting curves for the x and y coordinates, respectively. The equations of their fitted lines are $y = 2.5x + 27.8$ and $y = 0.93x - 0.74$. Besides, the RMSE of X coordinate is 5.18 and RMSE of Y coordinate is 3.65.

It is not difficult to see that the predicted results are far from what we expected. There are two possible reasons. First of all, the relationship between the x-coordinate and the y-coordinate of the maximum location is not considered. As discussed above, the maximum principle stress occurs on the inner ring of our model. Moreover, 49 training samples are too few to predict the location of maximum principle stress.

2) Training in Polar Coordinate: As explained before, we can simplify the problem by using angle in Polar coordinate as the only output. Once we obtain the predicted angle, we convert it to the Cartesian coordinate and compare with observations.

Figures 10 and 11 show the fitting curves for the x and y coordinates, respectively. The equations of their fitted lines are $y = 0.52x - 9.9$ and $y = 0.70x + 0.02$. Besides, the RMSE of X coordinate is 5.61 and RMSE of Y coordinate is 3.90. Although the problem is simplified and the running time is halved, the accuracy of the prediction does not improve. In this case, the number of training samples may be the decisive factor in determining the prediction accuracy.

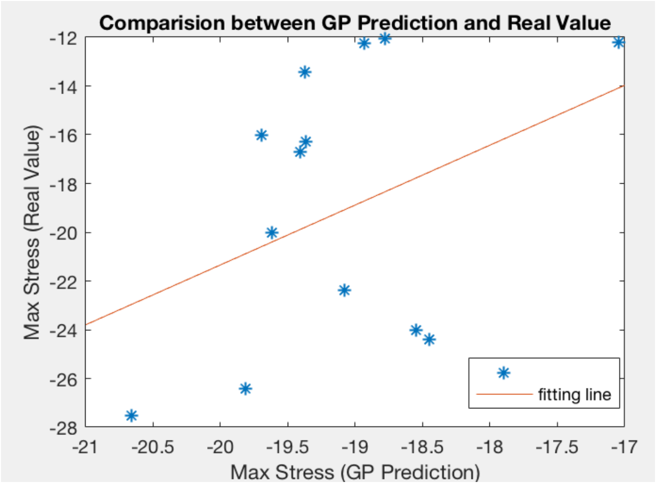


Fig. 8. Fitted Line for X Coordinate When Training X Y Separately

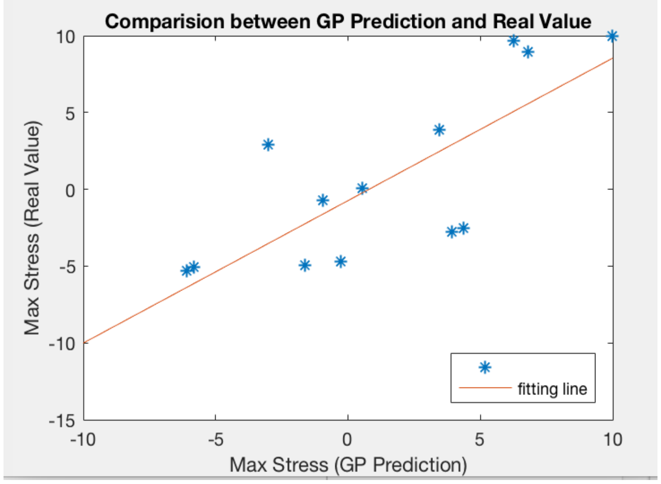


Fig. 9. Fitted Line for Y Coordinate When Training X Y Separately

3) Training with 121 training samples : Number of training samples increases from 49 to 121 by using Sample Plan 2. The results are shown in figure 12 and figure 13. The equations of fitted lines for the x and y coordinates are $y = 0.88x - 3$ and $y = 0.80x$, respectively. The RMSE of X coordinate is 3.7 and RMSE of Y coordinate is 3.0. By increasing the number of training samples, the prediction output gets closer to the observation output. Unlike the prediction of the maximum principle stress, more training data are needed to establish the GPR model to predict its location.

V. Conclusion

Gaussian Process is a very important machine-learning method. It enable us to encode prior information directly in the functional space.

Gaussian Process Regression is a very efficient tool to predict maximum stress for a non-linear model. After learn-

TABLE I
ERROR ANALYSIS

Error	SE	RQ	$Matern$	$SE+RQ$	$SE+Matern$	$RQ+Matern$	$SE+RQ+Matern$
RMSE	0.9582	0.7179	0.6285	0.6250	0.6176	0.6174	0.5431
R^2	0.9659	0.9809	0.9853	0.9855	0.9858	0.9858	0.9890
Time Cost/s	0.26	0.32	0.32	0.53	0.56	0.58	0.80

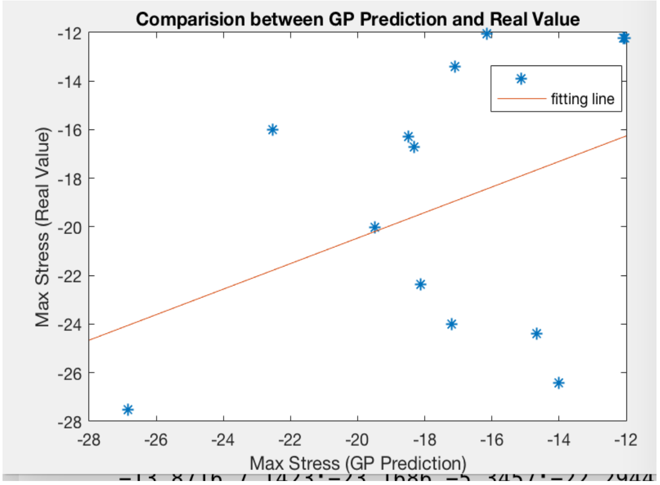


Fig. 10. Fitted Line for X Coordinate with 121 Training Samples

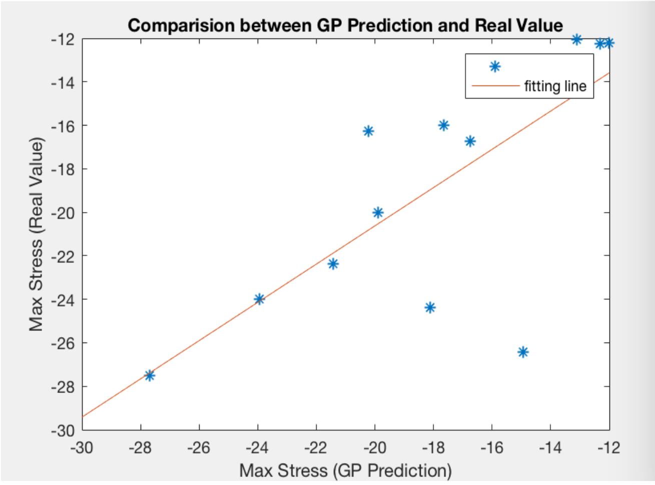


Fig. 12. Fitted Line for X Coordinate When Training in Polar Coordinate

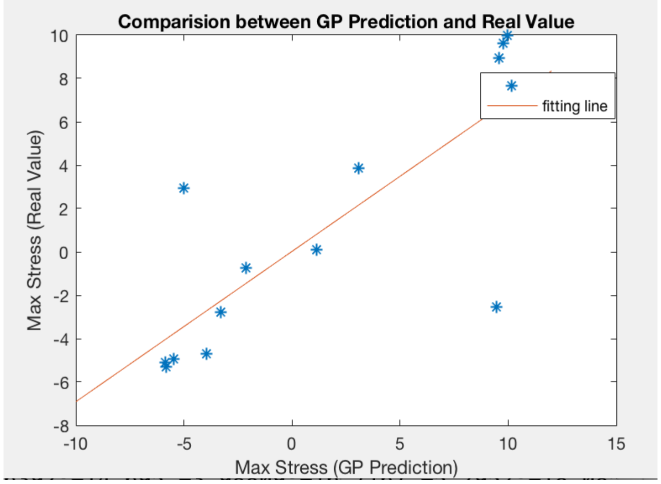


Fig. 11. Fitted Line for Y Coordinate with 121 Training Samples

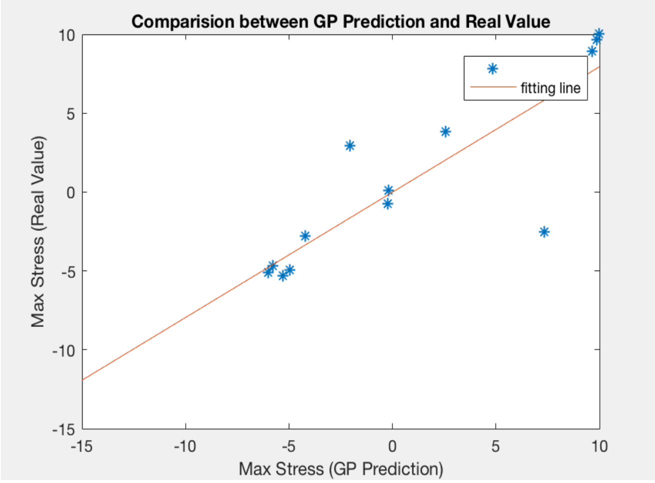


Fig. 13. Fitted Line for Y Coordinate When Training in Polar Coordinate

ing with 49 training samples and testing with 14 randomly generated samples, the RMSE between the observations and predictions is calculated as 0.5431 and the line equation is $y = 1.01x - 0.41$. These results show that GPR can give accurate predictions in estimating maximum principle stress.

The covariance function has a significant influence on the prediction result. Combined covariance functions have higher accuracy than simple covariance functions, but their running time also increases.

Unlike the prediction of maximum stress, the prediction of its location requires more training samples. The prediction accuracy improves significantly after increasing training data from 49 to 121.

VI. Future Work

1) Changing Sampling Method: Latin Hypercube Sampling can be used and replace Sampling Plan 1 and 2.

2) Using Proper Orthogonal Decomposition(POD) with large number of samples: We can increase the number of training samples to several hundreds when predicting the location of maximum principle stress. POD can be used to simplify large sample size problems.

3) Infill Criteria: It is possible to enhance the accuracy of the model using infill points, in addition to the initial sampling plan. For example, we can find the point with maximum variance and add a training sample at this point.

Acknowledgment

I would like to thank my tutor Dr. Joseph Morlier, who gave me this opportunity to work on this project and also help me in the the academic side. My thanks also go to Dr. Simone Coniglio, who has presented the whole subject of this project to me and has told me how to finish my work step by step. Last but not least, I would like to express thanks to Dr. Pierre-jean Barjhoux with his kindly help with my questions in Matlab.

References

[1] Alexander Forrester, Andy Keane, et al. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.

[2] Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

[3] David JC MacKay and David JC Mac Kay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[4] David Duvenaud. *Automatic model construction with Gaussian processes*. PhD thesis, University of Cambridge, 2014.

[5] Andrew Gordon Wilson. Covariance kernels for fast automatic pattern discovery and extrapolation with gaussian processes. *University of Cambridge*, 2014.

[6] Amar Shah, Andrew Wilson, and Zoubin Ghahramani. Student-t processes as alternatives to gaussian processes. In *Artificial Intelligence and Statistics*, pages 877–885, 2014.

[7] Carl Edward Rasmussen. Gaussian processes in machine learning. In *Advanced lectures on machine learning*, pages 63–71. Springer, 2004.

[8] Loic Le Gratiet. *Multi-fidelity Gaussian process regression for computer experiments*. PhD thesis, Université Paris-Diderot-Paris VII, 2013.

[9] Mohamed Amine Bouhlel. *Optimisation auto-adaptative en environnement d'analyse multidisciplinaire via les modèles de krigeage combinés à la méthode PLS*. PhD thesis, Toulouse, ISAE, 2016.