# Thermal Model Construction for Delfi-PQ

Ziyou Li, Ziqi Zhang
Supervisor: Stefano Speretta

*TU Delft, The Netherlands*

**Traditional satellite thermal model construction methods require accurate attitude determination results, leading to a limited application situation. The aim of this paper is to *explore the viability of using machine learning methods to construct thermal models for Delfi-PQ*. This was achieved by analysing Two-Line Elements (TLEs) and temperature measurements with three methods: least-squares exponential fitting, Support Vector Regression (SVR) and Prophet. To evaluate the performances and implementation complexity of the different methods, the statistical characteristics as well as the computing resources required by the methods were derived. It was concluded that both exponential curve fitting and Prophet can provide accurate prediction results while exponential curve fitting required minimal computing resources.**

## Nomenclature

| | | |
|---|---|---|
| MCU | = | Microcontroller Unit |
| RBF | = | Radial Basis Function |
| RMSE | = | Root-Mean-Square Error |
| SVM | = | Support Vector Machine |
| SVR | = | Support Vector Regression |
| TLEs | = | Two-Line Elements |
| TT&C | = | Telemetry, Tracking and Command |

## I. Introduction

The PocketQube program, which was proposed by California State Polytechnic University San Luis Obispo[1], can allow for a further decrease in the cost and size of the satellites. Because of the great prospects, Delfi-PQ was then designed by Delft University of Technology as a PocketQube platform with the goal of demonstrating a reliable core bus and outer structure for a three unit PocketQube, as introduced in [1]. After several years of preparation, it was launched to Sun-Synchronous Orbit at a nominal altitude of 525 km on January 13, 2022.[2].

For Delfi-PQ, the thermal control subsystem is one of the essential ones to ensure its proper functioning. It aims to keep the overall temperature within an acceptable range while also achieving the most

[1] `https://web.archive.org/web/20160303185449/http://mstl.atl.calpoly.edu/~bklofas/Presentations/DevelopersWorkshop2009/7_CubeSat_Alt/1_Twiggs-PocketQub.pdf`, retrieved on 24/6/2022

[2] `https://www.n2yo.com/satellite/?s=51074`, retrieved on 24/6/2022

adequate temperature distribution inside the spacecraft[3]. However, due to unforeseen problems, Delfi-PQ is experiencing internal temperatures lower than expected. For the battery subsystem, which was designed based on an operational temperature range from 5 °C to 40 °C, an actual temperature range from −10 °C to 25 °C is measured in orbit. This leads to a very limited amount of current provided to the satellite, preventing its normal functioning. Therefore, it is of vital importance to construct the orbital temperature model for Delfi-PQ, which can help not only solve the current problems, but also perform potential problem detection by comparing observed temperatures with the model predicted values.

Theoretically, thermal model construction requires many parameters to be accurately measured, which could prove difficult in practice. Data-driven approaches however, can be used to reconstruct a model from measurements and also make future predictions by means of machine learning, which is one of the fastest-growing areas of computer science with far-reaching applications [2]. Various machine learning methods have been developed, enabling humans to deal with different situations. For instance, supervised learning learns from desired input-output data and has shown the ability to predict with unseen input data [3]. Reinforcement learning differs from supervised learning in not requiring input-output pairs and deciding on actions based on the environment. Deep learning trains deep neural networks, producing outstanding results in computer vision, natural language processing and machine translation.

In particular, the applications of machine learning in space aspects progressively become popular, where its ability to deal with scarce and incomplete data has been demonstrated [4]. [5] provides machine learning-based satellite thermal power consumption prediction, while [6] considers machine learning applications in spacecraft control. Based on the fact that traditional satellite thermal model construction requires accurate determinations of the satellite attitudes, which limits the applications, *this article focuses on constructing the thermal models for Delfi-PQ with machine learning methods.* This can not only be used to solve the problems experienced by Delfi-PQ, but also constitute the first step for more complicated systems where anomalies can be detected by comparing measurements with a model representing the expected states.

From the Delfi-PQ in-orbit temperature measurements, this article firstly introduces a sequence of data preprocessing procedures, aiming to combine temperature records from multiple orbits together based on the derived relationship between temperature and the angle between the projection of the solar vector and the Earth-spacecraft vector. Thereafter, three methods were performed to achieve thermal model construction, namely physics-oriented least-squares exponential curve fitting, data-oriented SVR and data-oriented Prophet. Finally, the models constructed through different approaches will be compared in prediction accuracy and implementation complexity.

## II. Methodology

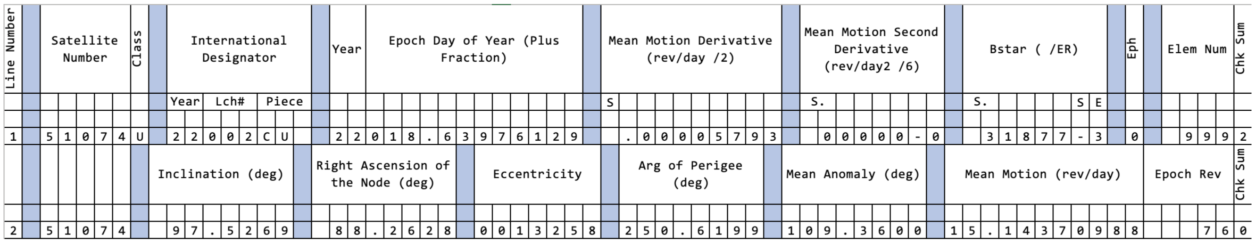
This section gives a detailed explanation of the three methods used to construct orbital temperature models, of which the results will be compared in prediction accuracy and implementation complexity. Firstly, the data used will be described in II.A. Afterwards, the feasibility of combining records from multiple orbits will be discussed and a series of orbital parameters will be derived in II.B. Finally, three methods are proposed in II.C. At first, a physics-oriented least-squares curve fitting method is used, which is a simple regression method based on a pre-determined piecewise exponential function. It can provide a straight overview of the collected data. Then, two data-oriented machine learning approaches are proposed. The first approach is SVR, which is a supervised learning model robust to outliers and requires lower computational power than other regression

[3] `https://www.esa.int/Enabling_Support/Space_Engineering_Technology/Thermal_Control#:~:text=The%20heat%20is%20rejected%20from,temperature%20distribution%20within%20the%20satellite.`, retrieved on 24/6/2022

methods. The second approach is Prophet, which shows strong performances in trend forecasting and is robust to a gap in the data [7].

## A. Data description

The primary raw data consists of two parts: TLEs and time series solar panel temperature data. As the name suggests, TLEs provide a set of Kepler orbit parameters for a particular satellite at a specific epoch time in the form of two lines of data, as shown in Fig. 1. It allows a rapid, moderately accurate orbital position prediction of the satellite [8].



| Line Number | Satellite Number | Class | International Designator | | | Year | Epoch Day of Year (Plus Fraction) | Mean Motion Derivative (rev/day /2) | Mean Motion Second Derivative (rev/day2 /6) | Bstar ( /ER) | Eph | Elem Num | Chk Sum |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Year | Lch# | Piece | | | S | S. | S. S E | | | |
| 1 | 51074 | U | 22 | 002 | CU | 22 | 018.63976129 | .00005793 | 00000-0 | 31877-3 | 0 | 999 | 2 |

| Line Number | Satellite Number | Inclination (deg) | Right Ascension of the Node (deg) | Eccentricity | Arg of Perigee (deg) | Mean Anomaly (deg) | Mean Motion (rev/day) | Epoch Rev | Chk Sum |
|---|---|---|---|---|---|---|---|---|---|
| 2 | 51074 | 97.5269 | 88.2628 | 0013258 | 250.6199 | 109.3600 | 15.14370988 | 76 | 0 |

**Figure 1. Two-Line Elements format.**

In addition to that, by the proper functioning of the TT&C subsystem, the satellite temperature data at various timestamps can be received on the ground. Fig. 2 shows a six-day temperature variation for Delfi-PQ, of which four solar panels' temperatures, battery temperatures and MCU temperatures are measured. It can be observed that the data is highly sparse, represented by the long straight lines in the figure. This is due to the limited satellite functionality caused by the low operational temperatures. Therefore, only a limited amount of on-board data can be transmitted. Moreover, the received temperatures present a periodic pattern with a period of approximate 96 min, which is equal to the orbit period of Delfi-PQ.

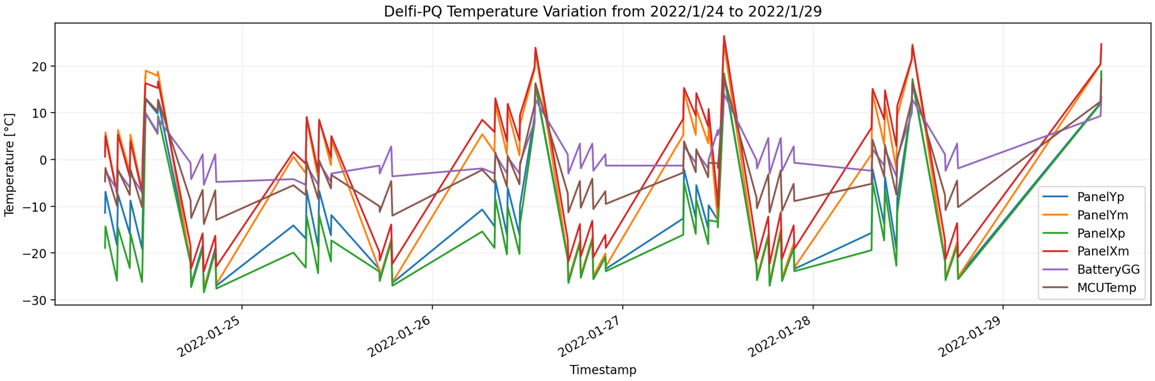


**Figure 2. Delfi-PQ temperature variation from 2022/1/24 to 2022/1/29.**

## B. Data preprocessing

II.A shows the sparsity of the available temperature data for a single orbit, which makes it difficult to perform model construction. Therefore, the idea of orbital overlaying, which aims to combine temperature records from several orbits together to make the data less sparse is proposed. The feasibility of this method is discussed in II.B.1. Moreover, the relationship between temperature and the angle between the projection of the solar vector and the Earth-spacecraft vector is considered as a key parameter of the satellite thermal system, and thus will be derived in II.B.2.

### *1. Orbital overlaying*

The prerequisite for this method to be feasible is that the orbits are very similar to each other in solar conditions. This can be justified by computing the eclipse fraction for each orbit following Eq. 1 and Eq. 2 [9], where $Re$ is the Earth radius and $h$ is the height of the spacecraft. $\beta$ is the angle between the solar vector and its projection onto the orbital plane, which can be derived from Eq. 3 with known the ecliptic true solar longitude ($\Gamma$), the right ascension of ascending node ($\Omega$), the orbital inclination angle ($i$) and the obliquity of the ecliptic for Earth ($\epsilon = 23.45°$ at present)[4] [9].

$$\beta^* = \sin^{-1}(\frac{Re}{Re + h}) \tag{1}$$

$$f_e = \begin{cases} \frac{1}{180°} \cos^{-1}[\frac{\sqrt{h^2+2R_e h}}{(R_e+h)\cos(\beta)}] & |\beta| < \beta^* \\ 0 & |\beta| \geq \beta^* \end{cases} \tag{2}$$

$$\beta = \sin^{-1}[\cos(\Gamma)\sin(\Omega)\sin(i) - \sin(\Gamma)\cos(\epsilon)\cos(\Omega)\sin(i) + \sin(\Gamma)\sin(\epsilon)\cos(i)] \tag{3}$$

The results are shown in Fig. 3. It can be calculated that in a period of 46 days, the eclipse fractions vary with the mean of 0.3468 and with the standard deviation of 0.0008243. Moreover, for the orbit with the lowest eclipse fraction, it will experience only an additional solar radiation of 23 s compared with the orbit with the highest eclipse fraction, based on an approximation of 96 min orbit period for Delfi-PQ. Therefore, it is reasonable to overlay multiple orbits as the illumination variation over this period was minimal.

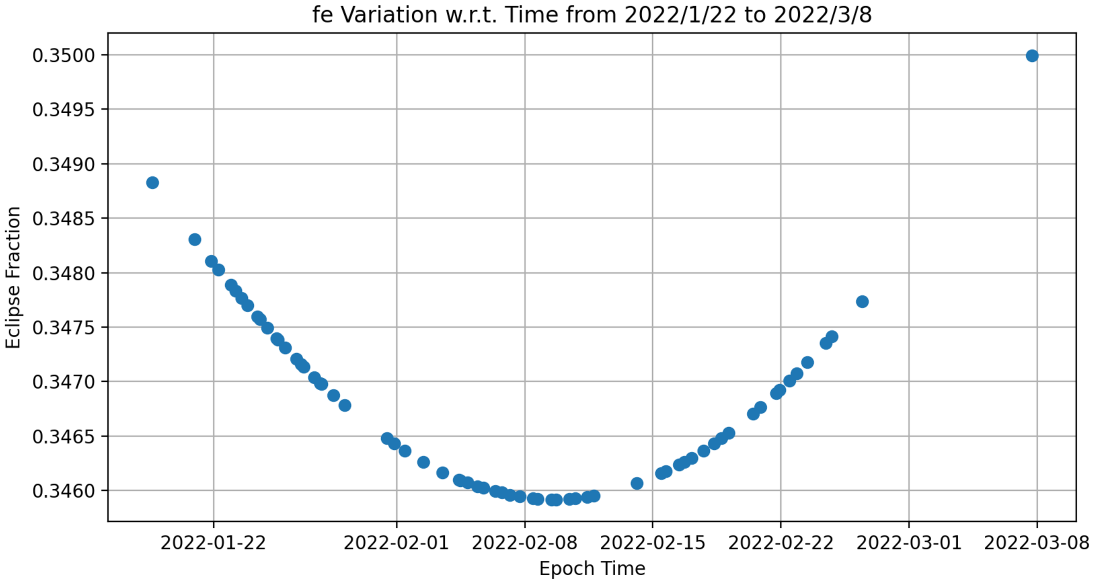


**Figure 3. $f_e$ variation with respect to time from 2022/1/22 to 2022/3/8.**

[4] `https://nescacademy.nasa.gov/?file=OnOrbitThermalEnvironmentsPart3.pptx&id=25&distr=Public`, retrieved on 24/6/2022

### 2. *θ angle calculation*

In order to perform the orbital overlay, the angle between the projection of the solar vector and the Earth-spacecraft vector will be derived in this section. As discussed in II.A, a group of orbital parameters at each epoch time **et**, as well as a set of recorded temperatures at each telemetry collection time **tt** are collected. However, since the telemetry collection time is not necessarily equal to the epoch time, the satellite's orbital parameters at each telemetry collection time need to be propagated from TLEs. Based on the fact that TLEs are valid from the time when they are released till the time at which new TLEs are released, telemetry collection times are grouped so that they can fall in the validity period of each TLE. For example, Eq. 4 gives two groups with epoch times $et_i$ and $et_{i+1}$

$$\begin{aligned} \ldots, [et_i, tt_j, tt_{j+1}, tt_{j+2}, \ldots], [et_{i+1}, tt_{j+n}, tt_{j+n+1}, tt_{j+n+2}, \ldots], \ldots \\ et_i \le tt_j, \ldots tt_{j+n-1} < et_{i+1} \end{aligned} \tag{4}$$

After completing this, the key parameters calculation inside each TLE time validity window will be performed. Fig. 4 illustrates the orbital parameters to be used in this section.

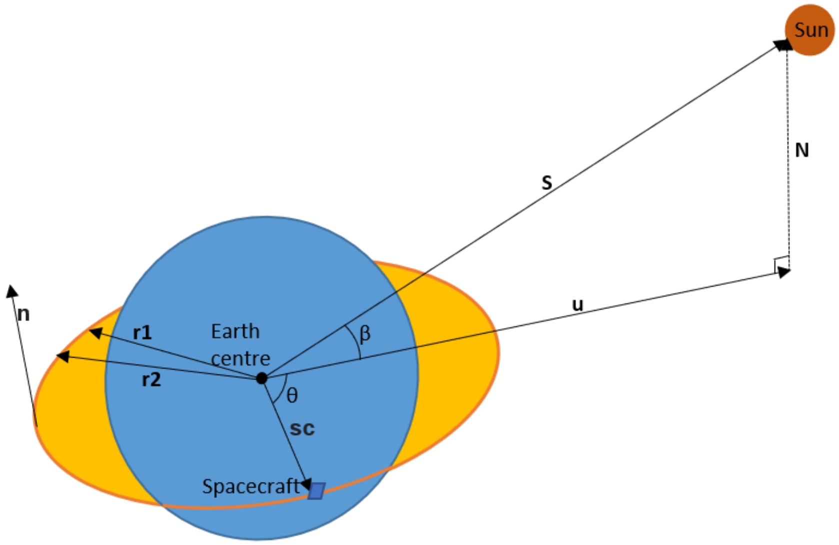


**Figure 4. Orbital parameters illustration.**

where:

- $\mathbf{r_1}$: Earth-spacecraft vector at a epoch time
- $\mathbf{r_2}$: Earth-spacecraft vector at 5 min after the epoch time
- **sc**: Earth-spacecraft vector at certain telemetry collection time in the validity window of the epoch time
- **S**: Solar vector at the epoch time
- **N**: Vector from the orbital plane to Sun and is perpendicular to the orbital plane
- **n**: Normal unit vector of spacecraft orbital plane
- **u**: Earth-Sun vector projection onto the spacecraft orbital plane
- $\beta$: Angle between the solar vector and its projection onto the spacecraft orbital plane
- $\theta$: Angle between the projection of the solar vector and the Earth-spacecraft vector

Firstly, the length of **N** can be obtained by following the trigonometric function shown in Eq. 5 with the calculated $\beta$ angle in II.B.1.

$$|\mathbf{N}| = |\mathbf{S}| \sin(\beta) \tag{5}$$

Moreover, since both $\mathbf{r_1}$ and $\mathbf{r_2}$ are on the spacecraft orbital plane, the normal unit vector of the orbital plane $\mathbf{n}$, which is parallel to $\mathbf{N}$, can be derived by the following Eq. 6. However, it is still not sure that the direction of $\mathbf{N}$ is the same as $\mathbf{n}$ or in the opposite. The angle $\alpha$, which is determined as the angle between $\mathbf{S}$ and $\mathbf{n}$, can help and is calculated in Eq. 7. It can be found that if the angle between $\mathbf{S}$ and $\mathbf{n}$ is smaller than 90°, $\mathbf{n}$ and $\mathbf{N}$ are in the same direction (Sun is above the orbital plane). Otherwise, $\mathbf{n}$ and $\mathbf{N}$ are in the opposite direction. This can be represented by Eq. 8.

$$\mathbf{n} = \frac{\mathbf{r_1} \times \mathbf{r_2}}{|\mathbf{r_1} \times \mathbf{r_2}|} \tag{6}$$

$$\alpha = \arccos(\frac{\mathbf{n} \cdot \mathbf{S}}{|\mathbf{n}||\mathbf{S}|}) \tag{7}$$

$$\mathbf{N} = \begin{cases} |\mathbf{N}| \cdot \mathbf{n} & 0^\circ < \alpha < 90^\circ \\ -|\mathbf{N}| \cdot \mathbf{n} & \alpha \geq 90^\circ \end{cases} \tag{8}$$

Finally, $\mathbf{u}$ can be obtained by vector subtraction following Eq. 9, and the $\theta$ angle for each telemetry collection time can be computed by Eq. 10, where the direction of $\mathbf{n}$ and that of the cross product of $\mathbf{u}$ and $\mathbf{sc}$ are used to ensure that the $\theta$ angle is determined following the satellite rotation direction.

$$\mathbf{u} = \mathbf{S} - \mathbf{N} \tag{9}$$

$$\theta = \begin{cases} \arccos \frac{\mathbf{u} \cdot \mathbf{sc}}{|\mathbf{u}||\mathbf{sc}|} & \arccos(\frac{\mathbf{n} \cdot (\mathbf{u} \times \mathbf{sc})}{|\mathbf{n}||(\mathbf{u} \times \mathbf{sc})|}) = 0^\circ \\ 360^\circ - \arccos \frac{\mathbf{u} \cdot \mathbf{sc}}{|\mathbf{u}||\mathbf{sc}|} & \arccos(\frac{\mathbf{n} \cdot (\mathbf{u} \times \mathbf{sc})}{|\mathbf{n}||(\mathbf{u} \times \mathbf{sc})|}) = 180^\circ \end{cases} \tag{10}$$

Fig. 5 shows an approximately linear relationship between $\theta$ angle and the time, caused by the circular orbit of Delfi-PQ. It can also be found that there are significant gaps in the data.

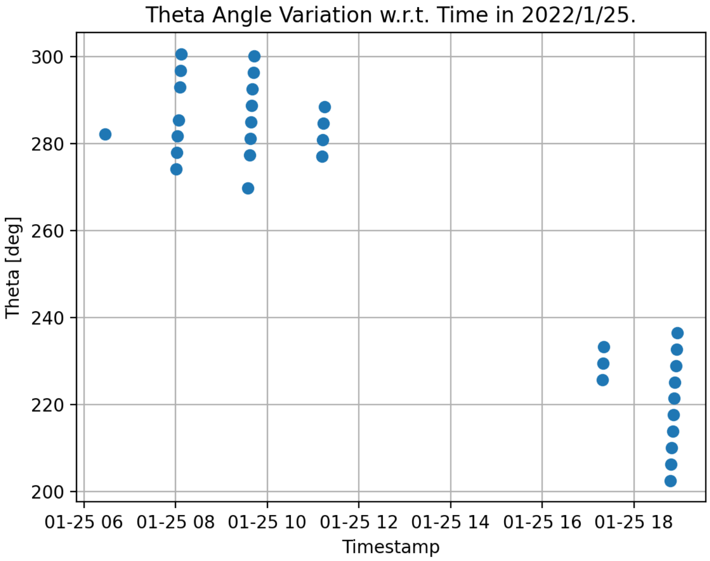


**Figure 5. $\theta$ angle variation with respect to time in 2022/1/25.**

One thing needs to be mentioned here is that the $\theta$ angle zero, which is aligned with the satellite being the closest to the Sun, can be moved to the eclipse-sunlight transition point by the following transformation introduced in Eq. 11.

$$\theta_t = \begin{cases} \theta + 180^\circ \cdot (1 - f_{e_{mean}}) & 0^\circ \leq \theta < 180^\circ \cdot (1 + f_{e_{mean}}) \\ \theta - 180^\circ \cdot (1 + f_{e_{mean}}) & 180^\circ \cdot (1 + f_{e_{mean}}) \leq \theta \end{cases} \tag{11}$$

Fig. 6 illustrates the temperature variation of panel Yp with respect to transformed $\theta$ angle, of which the zero value represents the eclipse-sunlight transition. Therefore, the measured temperature will increase in $[0, 360^\circ \cdot (1 - f_e)]$ and decrease in the rest of the orbit.

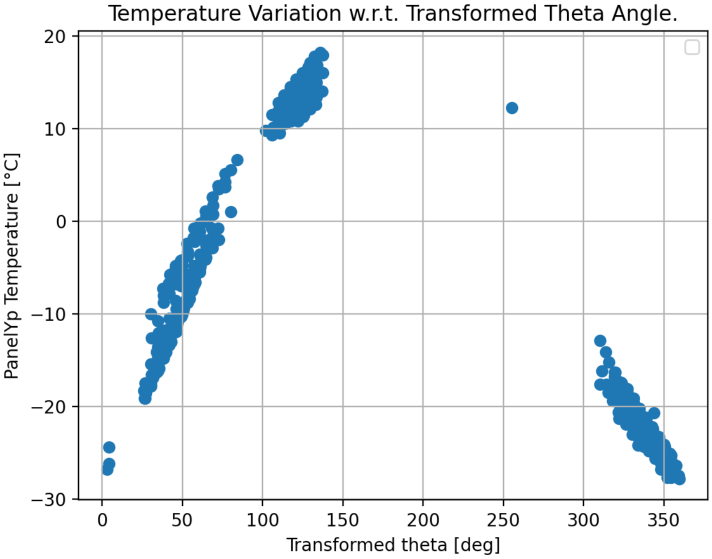


**Figure 6. PanelYp temperature variation w.r.t. transformed $\theta$ angle.**

### C. Model construction

With the completion of the data preprocessing procedures, this section then focuses on the model construction with the use of three different methods. The construction process uses the Panel Yp temperatures as a function of the transformed theta angles of overlaid orbits from January 24th, 2022 to March 18th, 2022. Besides, the eclipse fraction can be assumed constant and equal to the mean value, as its variation is minimal. The temperature prediction results will also be illustrated in this section.

#### *1. Exponential curve fitting*

Based on the fact that the satellite thermal system can be simplified as a lumped thermal system which could be solved by exponential functions, a simple least-squares exponential curve fitting is then performed to give a straight overview of the temperature variation with respect to the transformed $\theta$ angle. More specifically, two exponential functions with six free parameters are generated within $[0, 360^\circ \cdot (1 - f_e)]$ and $[360^\circ \cdot (1 - f_e), 360^\circ]$ separately. To ensure continuity, the parameters of the two functions should be chosen so that the two equations can reach the same value at the $\theta$ angle equal to $360^\circ \cdot (1 - f_e)$. Moreover, since both 0° and 360° represent the end of eclipse and the start of sunlight, they should also have the same value. These two constraints reduce the number of free parameters to four and finally give the following regression functions with $f_e$ being equal to 0.3468, as displayed in Eq. 12. The fitting result is also plotted in Fig. 7, where it can be found that a sharp drop is presented after the maximum temperature point. This is due to the non-derivable property of the pre-determined piecewise function.

$$T = \begin{cases} 97.84 \cdot (1 - e^{-0.004600\theta}) - 27.71 & 0 \leq \theta \leq 235.152 \\ 75.47 \cdot e^{-0.01479(\theta - 235.152)} - 39.62 & \theta > 235.152 \end{cases} \tag{12}$$

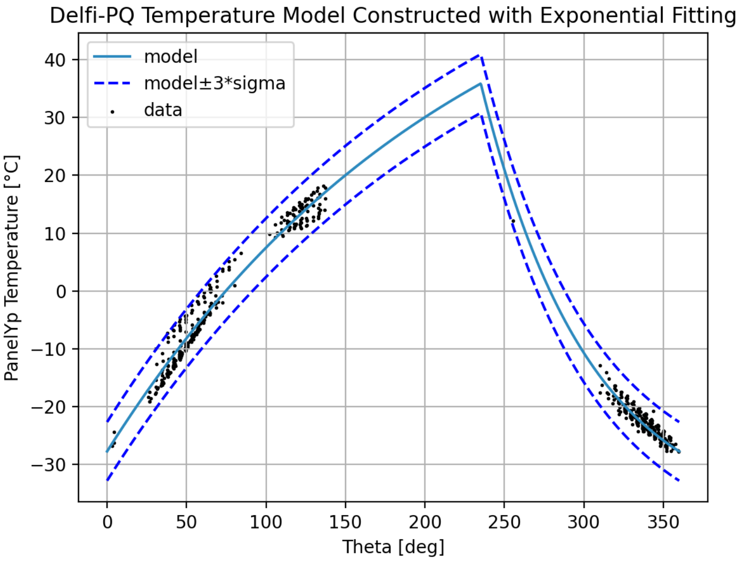


**Figure 7. Delfi-PQ temperature model constructed with exponential fitting.**

*2. SVM & SVR*

SVMs are supervised machine learning models with affiliated algorithms that analyse data for classification and regression analysis, originally proposed by V. N. Vapnik and A. Ya. Chervonenkis in 1964 as a linear classifier [10]. Combined with the later proposed kernel method [11], using SVM for non-linear function classification became feasible. A version of SVM for regression is also proposed [12], which can often be referred to as SVR. The method is chosen because of its ability of performing non-linear regression. Moreover, SVR is memory efficient, allowing the potential on-board implementation of the method.

In this article, the implementation of SVR of the Python package Scikit-learn version 1.0.0 is used [13]. The kernel method is set to the Scikit-learn default RBF and default settings of Scikit-learn is applied to other parameters. The learning result is presented in Fig. 8. It can be observed that the model built through SVR changed its trend at transformed $\theta$ being between 100° to 200°, which should be equal to 235.152° under constant eclipse fraction assumption, as shown in Fig. 7. This is mainly due to the lack of data in a large $\theta$ range where the temperature of the satellite starts decreasing.

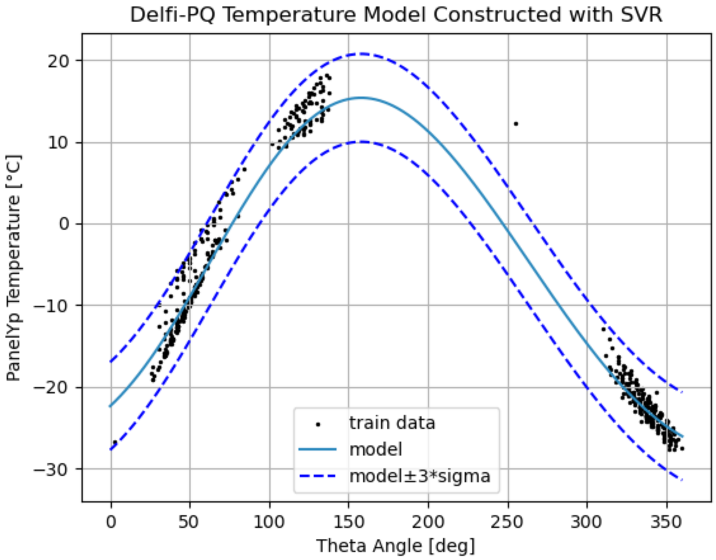


**Figure 8. Delfi-PQ Temperature model constructed with SVR**

*3. Prophet*

Prophet is a data forecasting procedure developed by Sean J. Taylor and Benjamin Letham at Meta[5]. It includes many parameters for the user to adjust the model, allowing improvements to the forecast by adding the domain knowledge. Therefore, it is especially robust to missing data and trend shifts. As discussed before, a temperature trend shift is expected to happen when $\theta$ angle data is missing, which can be specified by using Changepoint parameter in Prophet.

In this paper, the python implementation of Prophet version 1.0.1 is used. Also, as the temperature is expected to decrease from transformed $\theta$ being equal to 235.152°, a Changepoint at that point is added to the model manually as a model parameter. The prediction results are shown in Fig. 9, which can be found that the trend of the constructed model follows the solar conditions of the orbit smoothly.

[5] `https://research.facebook.com/blog/2017/2/prophet-forecasting-at-scale/`, , retrieved on 04/12/2022

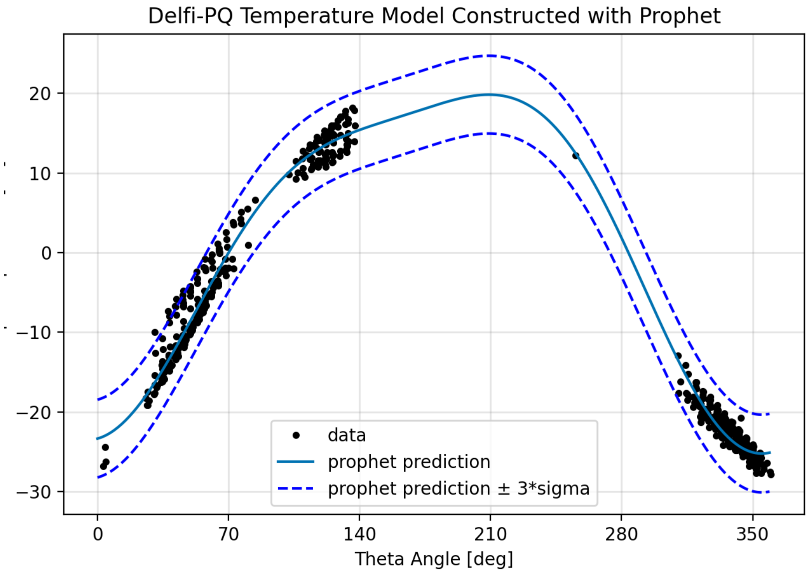


**Figure 9. Delfi-PQ Temperature model constructed with Prophet.**

## III. Results

After finalising the model construction procedures using three different methods, this section focuses on the comparison of the methods. Firstly, the prediction accuracy will be compared in III.A by calculating the representative statistical characteristics. Afterwards, the computational resources required by each method will be discussed in III.B.

### A. Prediction accuracy

Two criteria, R2 score and RMSE, are used as the representative statistical characteristics to evaluate the results of the aforementioned three methods.

R2 score represents the proportion of variance and is calculated as Eq. 13, where $y_i$ is the $i$th observed value, $\hat{y}_i$ is the corresponding $i$th predicted value, and $\bar{y}$ is the mean of all observed values. The best possible score is 1, representing that the predicted values exactly match the observed values, and the worst is 0. The R2 score can provide an indication of the quality of the fit and can hence measure how well unseen samples may be predicted by the model [6].

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{13}$$

RMSE is a commonly used measure of prediction error, which is the square root of the average of squared errors and is defined as Eq. 14. It can hence provide a typical difference between the predicted values and the true values. In general, a lower RMSE indicates a better fit. The best possible RMSE for a model is 0, meaning that there exists no error between the predicted values and the true values.

[6] https://scikit-learn.org/stable/modules/model_evaluation.html#r2-score, retrieved on 26/6/2022

$$\mathrm{RMSE}(y, \hat{y}) = \sqrt{\frac{1}{n_{\mathrm{samples}}} \sum_{i=0}^{n_{\mathrm{samples}}-1} (y_i - \hat{y}_i)^2} \tag{14}$$

The R2 score and RMSE of the three earlier constructed models are shown in Table 1, which can be observed highly similar. The first method produced comparably accurate results to the other two complicated algorithms by simply fitting two exponential functions. This is mainly because that the satellite thermal system can be approximately considered as a lumped thermal problem which can be solved by an exponential function. Moreover, the model predicted through SVR does not satisfy the fact that the transformed $\theta$ should be equal to $235.152°$ where the temperature reaches its maximum value , as in the other two models. This indicates that SVR failed to predict the accurate maximum value, which is due to the lack of observed temperatures.

**Table 1. R2 Score and RMSE for Different Methods.**

| Method | R2 Score | RMSE |
|---|---|---|
| Exponential Curve Fitting | 0.9826 | 1.6889 |
| SVR | 0.9807 | 1.7927 |
| Prophet | 0.9838 | 1.6285 |

**B. Computational resources**

Memory usage and computation time are chosen as the criteria for the measurement of the required computational resources because of their intuitiveness and breadth of use. As the computational resources of a satellite are generally restricted, it is expected that the model construction would require as little memory space as possible, and would require as short computation time as possible.

Due to the restricted access to the actual satellite and due to that the measurement is only for the purpose of relative comparison, the computational resources measurement is executed on a computer with technical specifications shown in Table 2. The memory usage is measured with the heap analysis tool $Guppy3$ version 3.1.2. while the computation time is measured with the $time$ Python module.

**Table 2. Technical Specifications Used for Computational Resources Measurement.**

| Component | Type |
|---|---|
| Operating System | macOS Big Sur 11.6.6 |
| CPU | Intel Core i5 8259U @ 2.3GHz |
| Memory | 16GB 2133 MHz LPDDR3 |

Table 3 shows the average memory usage and computation time for each method to perform data preprocessing and model constructing. Exponential curve fitting is the most economical in terms of memory usage. It requires only 1.64 MB of memory space, nearly one tenth of the other two methods. It also only requires the least computation time of 5.8 seconds. In general, exponential curve fitting is the most computationally economical method and Prophet require the most resources among the three methods.

**Table 3. Memory Usage and Computation Time for Different Methods**

| Method | Memory Usage [MB] | Computation Time [s] |
|---|---|---|
| Exponential Curve Fitting | 1.648253 | 5.808 |
| SVR | 15.153566 | 6.693 |
| Prophet | 15.661035 | 7.355 |

## IV. Conclusion

In this paper, the feasibility of using machine learning methods to construct orbital temperature models for Delfi-PQ was explored. More specifically, three thermal models were constructed by analysing TLEs and temperature records with least-squares exponential fitting, SVR and Prophet respectively.

Firstly, in order to deal with highly sparse raw data, an orbital overlaying method was proposed, of which the feasibility was carefully clarified with the eclipse fraction computation. Afterwards, the relationship between $\theta$ angles and temperatures from multiple orbits was derived, based on which the three models were built. They were finally compared in the prediction accuracy by means of R2 score and RMSE, and the computational resources requirements by the agency of memory usage and computation time.

It can be concluded that the R2 score and RMSE are quite similar among the three methods while exponential curve fitting shows significant advantages in the consumption of computational resources aspect. As one of the complicated algorithms, Prophet generates accurate predictions and shows its potentials on dealing with more complicated problems. SVR however, fails to keep the increasing trend when the satellite is still absorbing solar energy. Exponential function fitting also provides good results because that it is based on lumped thermal system assumption, but presents a sudden drop after transiting from sunlight to eclipse.

Due to limited data of TLEs and temperature records, this article only discusses the thermal models in a two-month period. Therefore, it is recommended to further investigate the feasibility of orbit overlaying as well as the model construction methods in a long period with potential significant changes in eclipse fraction. Moreover, in the second method, only RBF is used as the kernel function. Thus, it is also recommended to test SVR methods with more types of kernel functions.

## Acknowledgments

I would like to express my deepest gratitude and appreciation to Professor Stefano Speretta who made this work possible. Not only did he help acquire and refine the data, but also inspired me by proposing the idea of orbital overlaying. Moreover, his guidance, patience and encouragement are indispensable for me to complete this project.

# References

[1] Radu, S., Uludag, S., Speretta, S., Bouwmeester, J., Gill, E., and Foteinakis, N., "Delfi-PQ: The first pocketqube of Delft University of Technology," *69th International Astronautical Congress: Bremen, Germany [IAC-18-B4.6B. 5] International Astronautical Federation, IAF*, 2018.

[2] "Understanding Machine Learning: From Theory to Algorithms," ???? URL `http://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning`.

[3] Russell, S., and Norvig, P., *Artificial intelligence : a modern approach*, third edition. ed., Prentice Hall, 2010. doi: 10.1017/S0269888900007724.

[4] Chien, S., Doyle, R., Davies, A. G., Jónsson, A., and Lorenz, R., "The future of ai in space," *IEEE Intelligent Systems*, Vol. 21, 2006, pp. 64–69. doi: 10.1109/MIS.2006.79.

[5] Petković, M., Boumghar, R., Breskvar, M., Džeroski, S., Kocev, D., Levatić, J., Lucas, L., Osojnik, A., Ženko, B., and Simidjievski, N., "Machine Learning for Predicting Thermal Power Consumption of the Mars Express Spacecraft," *IEEE Aerospace and Electronic Systems Magazine*, Vol. 34, No. 7, 2019, pp. 46–60. doi: 10.1109/MAES.2019.2915456.

[6] Shirobokov, M., Trofimov, S., and Ovchinnikov, M., "Survey of machine learning techniques in spacecraft control design," *Acta Astronautica*, Vol. 186, 2021, pp. 87–97. doi: https://doi.org/10.1016/j.actaastro.2021.05.018, URL `https://www.sciencedirect.com/science/article/pii/S0094576521002514`.

[7] Taylor, S. J., and Letham, B., "Forecasting at scale," 2017. doi: 10.7287/PEERJ.PREPRINTS.3190V2, URL `https://peerj.com/preprints/3190`.

[8] Vallado, D., and Cefola, P., "Two-line element sets - Practice and use," *Proceedings of the International Astronautical Congress, IAC*, Vol. 7, 2012, pp. 5812–5825.

[9] M, S., "Computation of Eclipse Time for Low-Earth Orbiting Small Satellites," *International Journal of Aviation, Aeronautics, and Aerospace*, 2019. doi: 10.15394/ijaaa.2019.1412.

[10] Vapnik, V. N., and Chervonenkis, A. Y., "A class of algorithms for pattern recognition learning," *Avtomat. i Telemekh.*, Vol. 25, 1964, pp. 937–945.

[11] Aizerman, M. A., Braverman, E. A., and Rozonoer, L., "Theoretical foundations of the potential function method in pattern recognition learning." *Automation and Remote Control,*, 1964, pp. 821–837.

[12] "Support vector regression machines | Proceedings of the 9th International Conference on Neural Information Processing Systems," , ???? URL `https://dl.acm.org/doi/10.5555/2998981.2999003`.

[13] "1.4. Support Vector Machines — scikit-learn 1.1.0 documentation," , ???? URL `https://scikit-learn.org/stable/modules/svm.html`.