

School of Surveying and Geospatial Engineering
Department of Geodesy and Hydrography

Spherical Spline Implementation

Author:
Aliakbar Zarkoob

Professor:
Dr. Abdolreza Safari

Winter 2025

1 Introduction

Spherical spline interpolation has emerged as a critical tool in addressing approximation problems on the sphere, particularly in fields such as geophysics, physical geodesy, and environmental sciences. The increasing availability of high-resolution, spatially distributed data, such as those generated by satellite-based techniques, has necessitated the development of robust mathematical methods to approximate functions defined on spherical domains. Such functions often represent physical quantities like temperature, pressure, ozone concentration, gravitational and magnetic forces, or elastic deformation. These quantities are typically sampled at discrete, irregularly spaced points on the Earth's surface, requiring interpolation methods that can accurately reconstruct them over the entire spherical domain.

Traditional Euclidean methods for localized approximation are often inadequate for global problems on the sphere, such as gravity field determination. The fundamental challenge lies in the lack of a differential mapping that can transform the entire sphere into a bounded planar region without distortion. This limitation has driven the development of approximation methods specifically tailored to spherical domains. Among these, spherical spline interpolation stands out for its ability to model global phenomena effectively while maintaining desirable mathematical properties, such as smoothness and stability.

This report is focused on spherical spline interpolation for gravity potential data, where three distinct kernels (Abel-Poisson, Singularity, and Logarithmic) are employed for interpolating data from a global 6-degree regular grid to a finer 3-degree grid.

2 Implementation Steps

The answer for spherical spline is linear combination of the kernel function, as shown in the equation 1. Where y is a desired point, x_i are the input points, a_i are the coefficients for spherical spline interpolation, and N is the number of input points.

$$S(y) = \sum_{i=1}^N a_i K(x_i, y) \quad (1)$$

The coefficients can be calculated from the linear system of equations shown in equation 2. The matrix equation is shown in equation 3. Where both x_i and y_j represent input points and $F(y_j)$ are the values at input points. Thus the coefficients can be calculated by inverting the design matrix (A).

$$F(y_j) = \sum_{i=1}^N a_i K(x_i, y_j) , \quad j = 1, 2, \dots, N \quad (2)$$

$$\begin{bmatrix} F(y_1) \\ F(y_2) \\ \vdots \\ F(y_N) \end{bmatrix} = \begin{bmatrix} K(x_1, y_1) & K(x_2, y_1) & \dots & K(x_N, y_1) \\ K(x_1, y_2) & K(x_2, y_2) & \dots & K(x_N, y_2) \\ \vdots & \vdots & \ddots & \vdots \\ K(x_1, y_N) & K(x_2, y_N) & \dots & K(x_N, y_N) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} \equiv l = Ax \quad (3)$$

$$\hat{x} = A^{-1}l \quad (4)$$

After the coefficients are calculated, the value at any desired point can be calculated from the equation 1, or the matrix equation of 5 for U Points.

$$\begin{bmatrix} S(y_1) \\ S(y_2) \\ \vdots \\ S(y_U) \end{bmatrix} = \begin{bmatrix} K(x_1, y_1) & K(x_2, y_1) & \dots & K(x_N, y_1) \\ K(x_1, y_2) & K(x_2, y_2) & \dots & K(x_N, y_2) \\ \vdots & \vdots & \ddots & \vdots \\ K(x_1, y_U) & K(x_2, y_U) & \dots & K(x_N, y_U) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} \quad (5)$$

As mentioned, three kernel functions of Abel-Poisson, Singularity, and Logarithmic are used in this project. These three functions are defined in the equations 6 to 8. Where x and y are Cartesian coordinate vectors of a point on the sphere, and h is the smoothing parameter of the kernel which it's value is in range of $(0, 1)$.

$$K_{\text{Abel-Poisson}}(x, y) = \frac{1}{4\pi} \frac{1 - h^2}{(L_h(x, y))^{\frac{3}{2}}} \quad (6)$$

$$K_{\text{Singularity}}(x, y) = \frac{1}{2\pi} \frac{1}{(L_h(x, y))^{\frac{1}{2}}} \quad (7)$$

$$K_{\text{Logarithmic}}(x, y) = \frac{1}{2\pi h} \ln \left(1 + \frac{2h}{(L_h(x, y))^{\frac{1}{2}} + 1 - h} \right) \quad (8)$$

$$L_h(x, y) = 1 + h^2 - 2h \langle x, y \rangle \quad (9)$$

As mentioned before, data used in this project is gravity potential. This data was acquired from ICGEM¹, and it's shown in the figure 1.

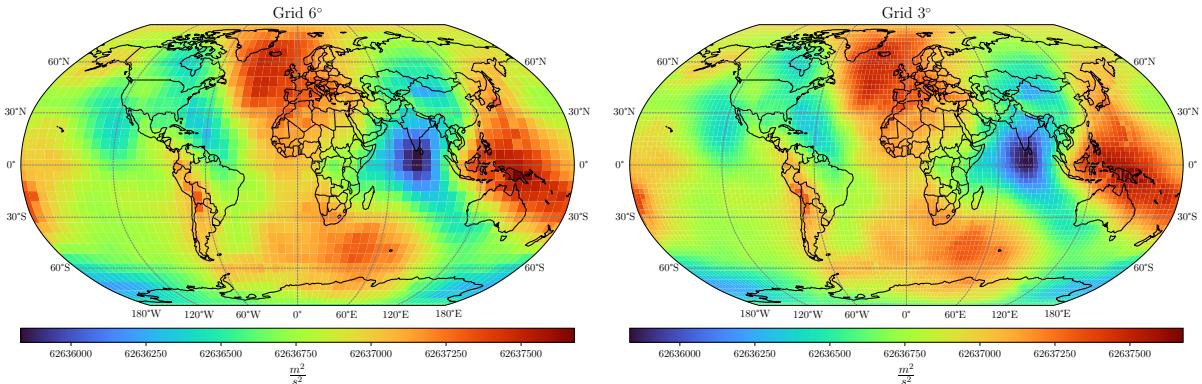


Figure 1: Gravity potential data used for interpolation and validation.

One of the main challenges of this project is determining the optimum value for parameter h . The method used here is a simple Cross-Validation (CV), where 5% of input data was left aside from calculations and the norm of difference between interpolated value and true value for these points were taken into account for determining the best h , as shown in equation 10. In this equation, S is the gravity potential values and \hat{S} is the interpolated value for the validation points.

$$h_{opt} = \arg \min_h \|S - \hat{S}\|_2 \quad (10)$$

Different values for h were tested in a manner of three steps. Firstly, values from 0.1 to 0.9 by step size of 0.1 were tested. Then values around the result of previous step

¹International Center for Global Earth Models: <https://icgem.gfz-potsdam.de>

with the step size of 0.01 were tested. And Lastly the values by the step size of 0.001 around the result of the previous step were used.

The design matrix A for solving coefficients is ill-conditioned, with the condition number in order of 10^{40} (Abel-Poisson kernel with $h = 0.3$). The Picard plot for this matrix is shown in figure 2.

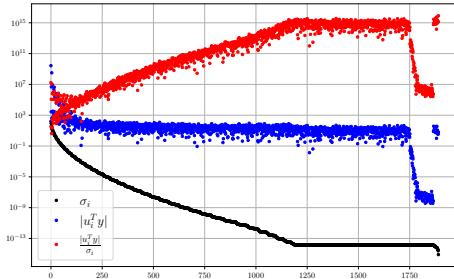


Figure 2: Picard plot of the design matrix (Abel-Poisson kernel with $h = 0.3$).

Therefore methods of regularization is required for calculating the coefficients, and the solution by simply inverting the design matrix is not a reasonable solution, as shown in figure 3.

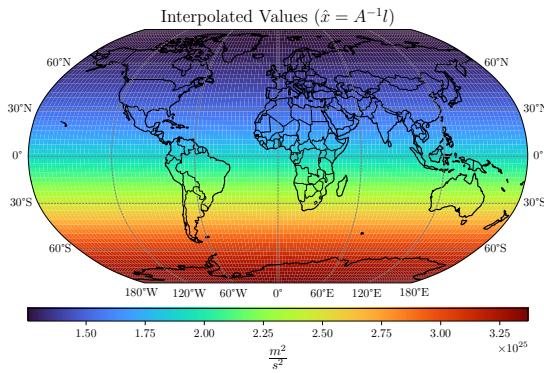


Figure 3: Results without using regularization methods.

Regularization methods used are TSVD and First order Tikhonov (FOT) With regularization parameter calculated from GCV, L-Curve and VCE methods. Cholesky decomposition is also used, which is a method for cases with numerical problems.

Error of interpolation is defined as shown in equation 11, Where l is the true value acquired from XGM2019 model, and \hat{l} is the interpolated value.

$$e = l - \hat{l} \quad (11)$$

3 Results

3.1 Abel-Poisson

Interpolated and error values for Abel-Poisson kernel is shown in the figures 4 to 8. The results for this kernel, including parameter h , regularization parameter, Mean and norm (L^2) of error are shown in table 1. As it can be understood from the table, TSVD method was able to achieve the best result.

Table 1: Results of interpolation using Abel-Poisson kernel (units of norm and mean values are $\frac{m^2}{s^2}$).

Method	Cholesky	TSVD	GCV (FOT)	L-Curve (FOT)	VCE (FOT)
Regularization Parameter	—	1010	$4.12 * 10^{-7}$	$8.57 * 10^{-5}$	$2.16 * 10^{-8}$
h	0.273	0.481	0.298	0.249	0.248
Mean of Error	0.2083	0.1711	-28.8998	2.5335	0.5352
Norm of Errors	2076.6487	2010.3925	4487.5036	7731.3095	5979.1538

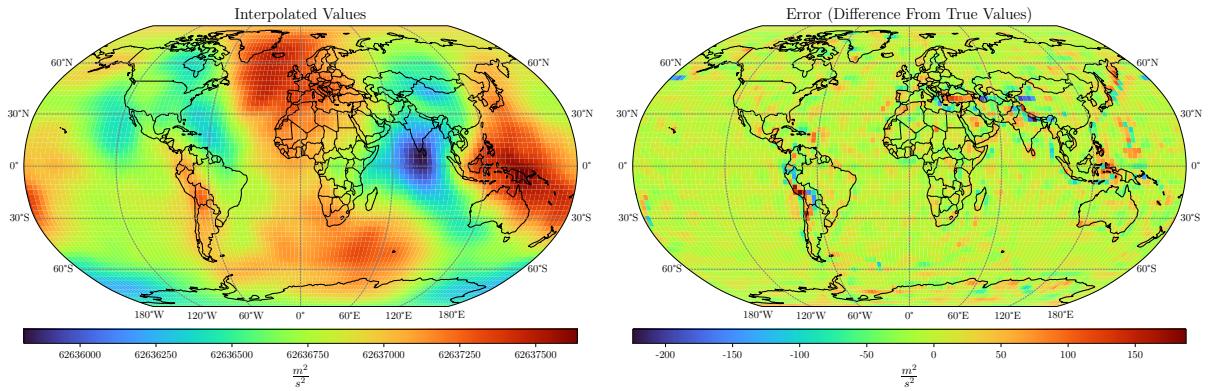


Figure 4: Results of Abel-Poisson kernel with Cholesky decomposition.

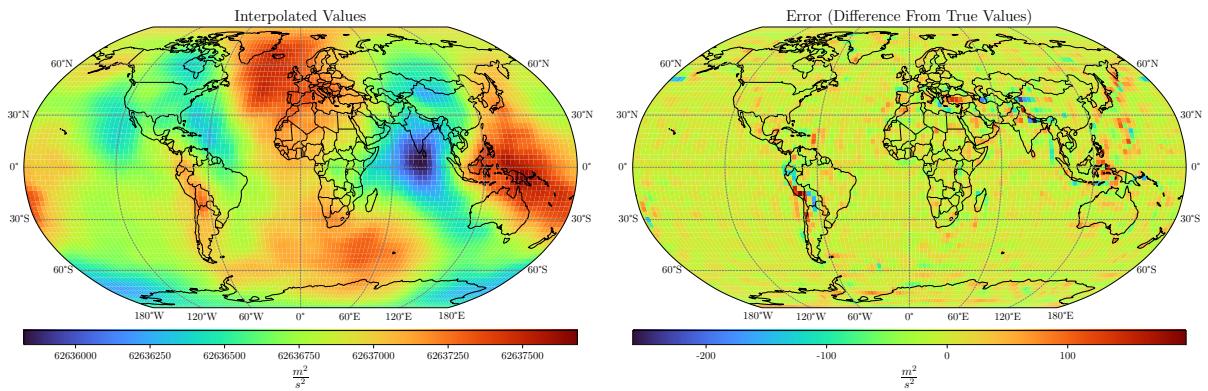


Figure 5: Results of Abel-Poisson kernel with TSVD method.

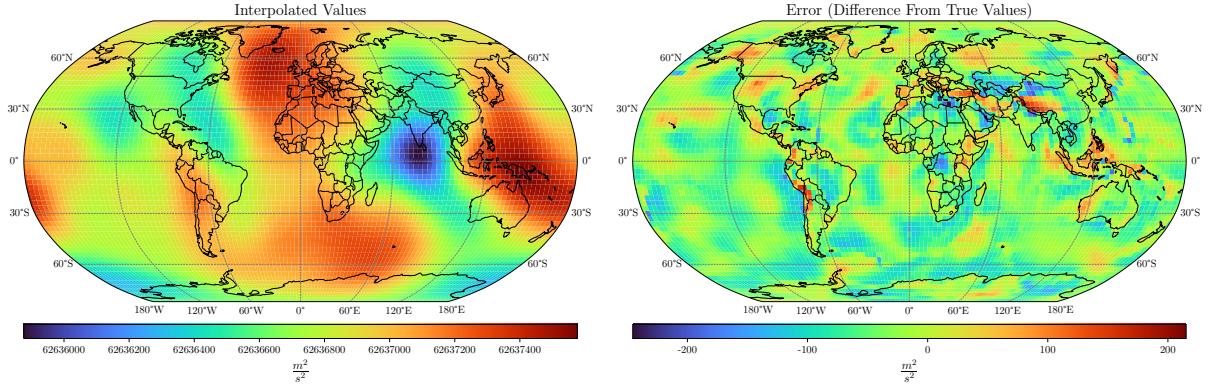


Figure 6: Results of Abel-Poisson kernel with GCV (FOT) method.

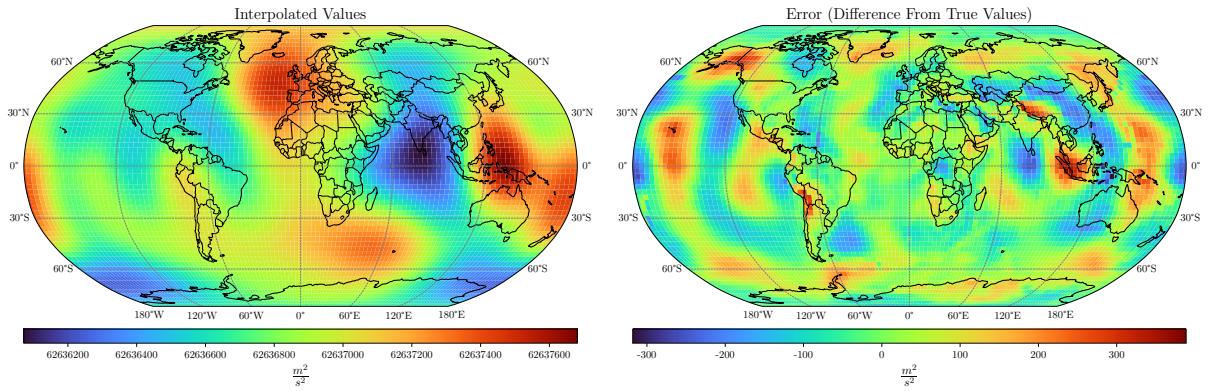


Figure 7: Results of Abel-Poisson kernel with L-Curve (FOT) method.

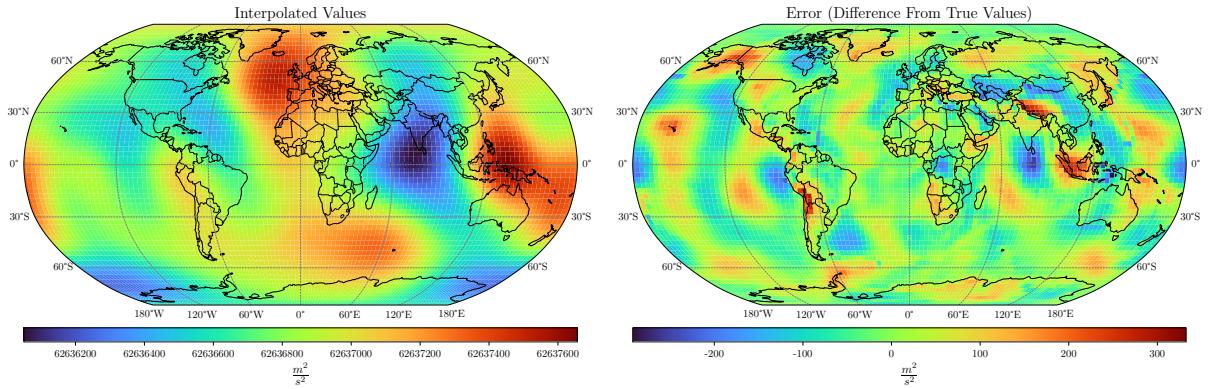


Figure 8: Results of Abel-Poisson kernel with VCE (FOT) method.

3.2 Singularity

Interpolated and error values for Singularity kernel is shown in the figures 9 to 13. The results for this kernel, including parameter h , regularization parameter, Mean and norm (L^2) of error are shown in table 2. As it can be understood from the table, TSVD method was able to achieve the best result.

Table 2: Results of interpolation using Singularity kernel (units of norm and mean values are $\frac{m^2}{s^2}$).

Method	Cholesky	TSVD	GCV (FOT)	L-Curve (FOT)	VCE (FOT)
Regularization Parameter	—	1005	$4.26 * 10^{-6}$	$1.43 * 10^{-4}$	$3.27 * 10^{-8}$
h	0.264	0.429	0.364	0.427	0.440
Mean of Error	0.2437	0.1703	9.1348	1.8637	5.5732
Norm of Errors	2167.4400	2013.2368	4211.3836	5557.0321	3745.0423

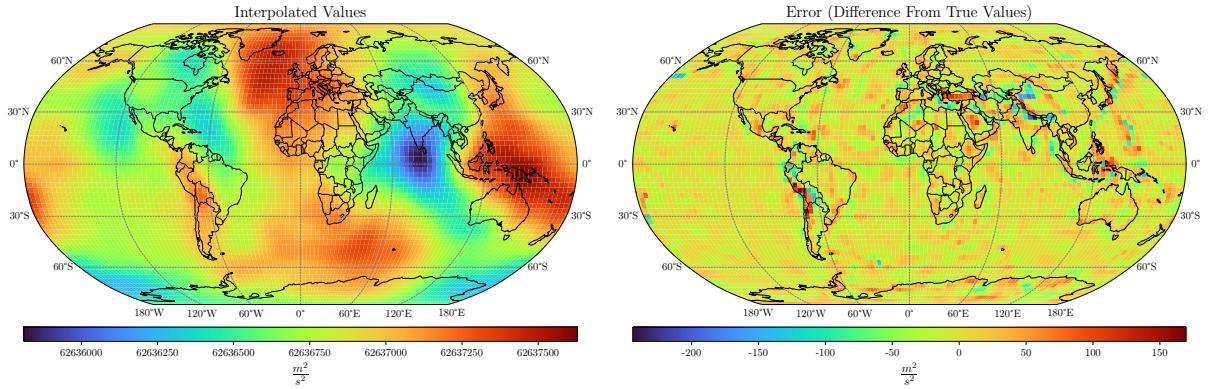


Figure 9: Results of Singularity kernel with Cholesky decomposition.

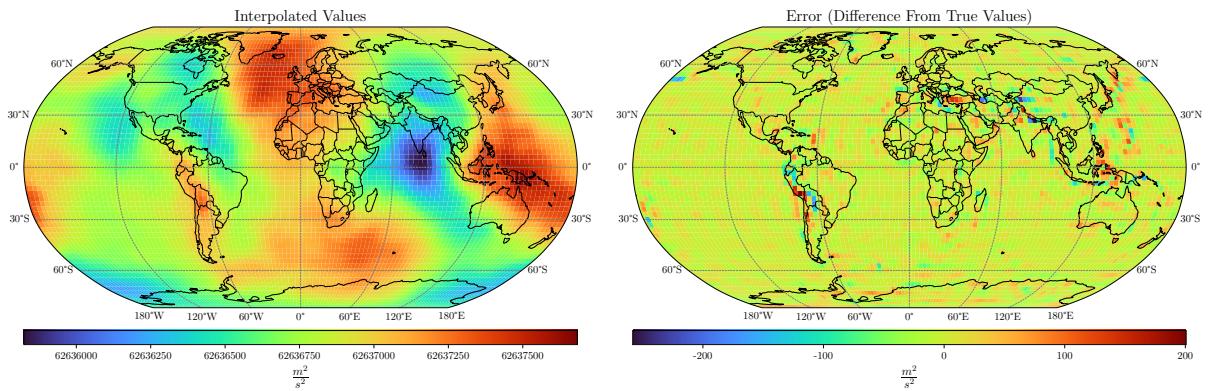


Figure 10: Results of Singularity kernel with TSVD method.

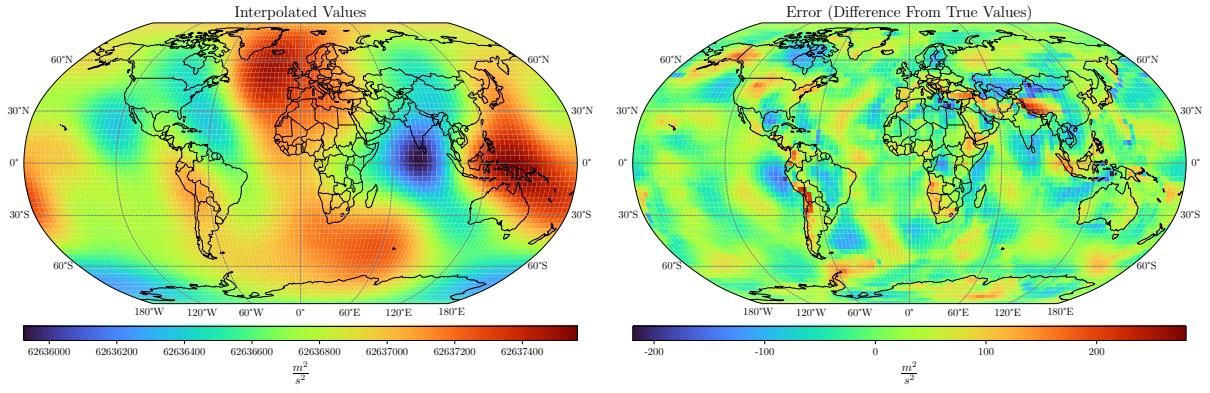


Figure 11: Results of Singularity kernel with GCV (FOT) method.

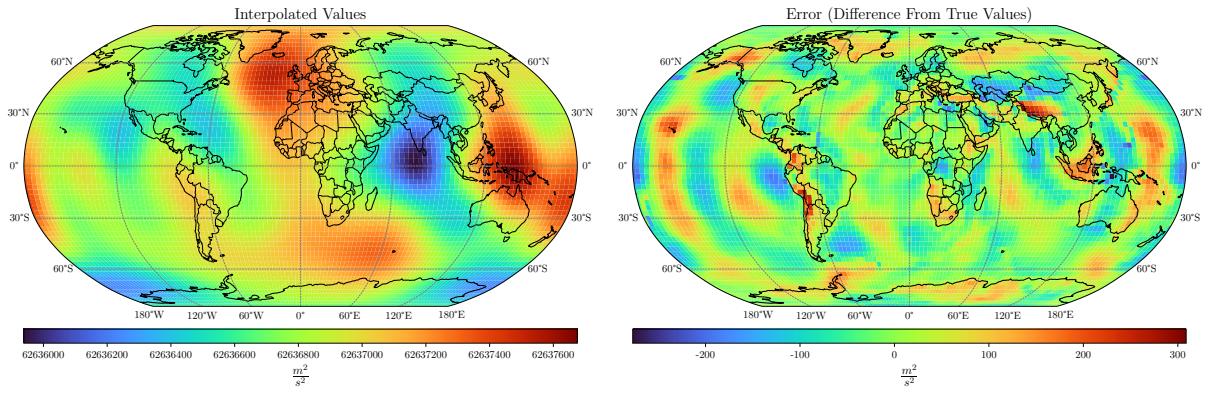


Figure 12: Results of Singularity kernel with L-Curve (FOT) method.

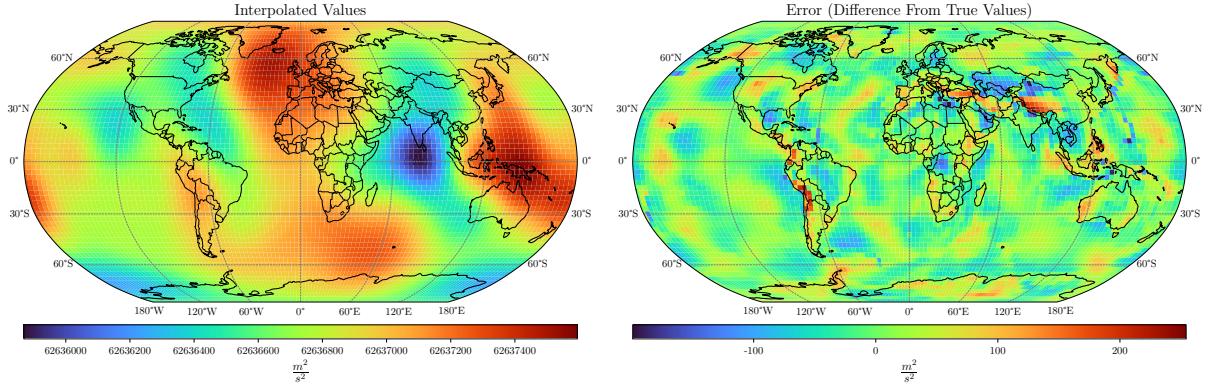


Figure 13: Results of Singularity kernel with VCE (FOT) method.

3.3 Logarithmic

Interpolated and error values for Logarithmic kernel is shown in the figures 14 to 18. The results for this kernel, including parameter h , regularization parameter, Mean and norm (L^2) of error are shown in table 3. As it can be understood from the table, TSVD method was able to achieve the best result.

Table 3: Results of interpolation using Logarithmic kernel (units of norm and mean values are $\frac{m^2}{s^2}$).

Method	Cholesky	TSVD	GCV (FOT)	L-Curve (FOT)	VCE (FOT)
Regularization Parameter	—	1010	$7.96 * 10^{-7}$	$1.20 * 10^{-4}$	$1.47 * 10^{-7}$
h	0.286	0.559	0.411	0.535	0.367
Mean of Error	0.2559	0.1712	57.395	1.8417	1.3222
Norm of Errors	2220.5339	2007.4311	6614.5388	5723.9421	7804.3402

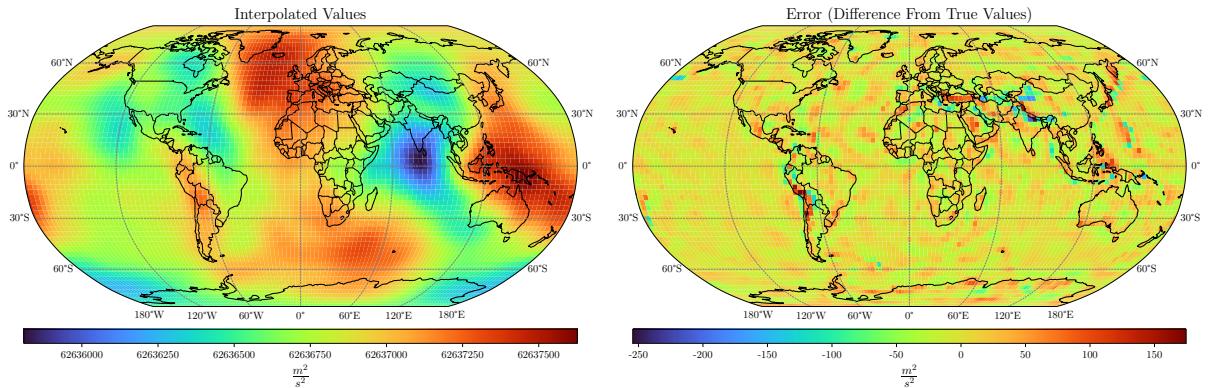


Figure 14: Results of Logarithmic kernel with Cholesky decomposition.

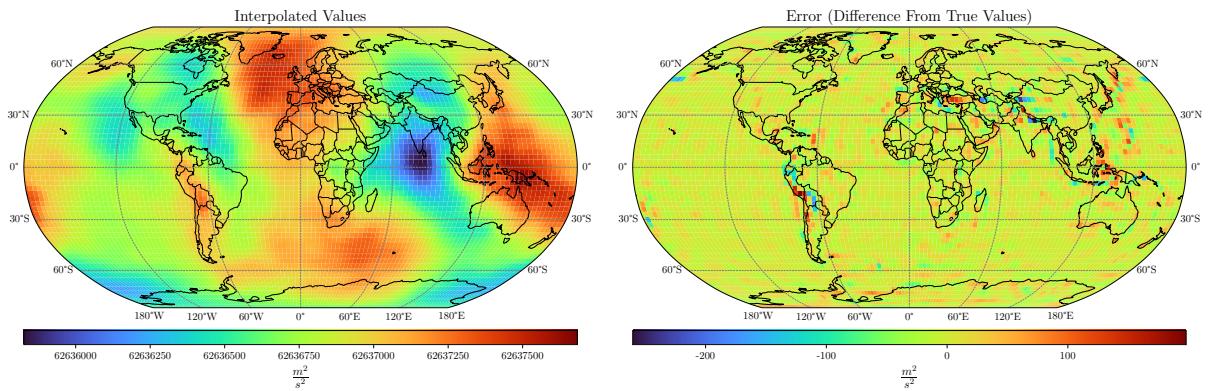


Figure 15: Results of Logarithmic kernel with TSVD method.

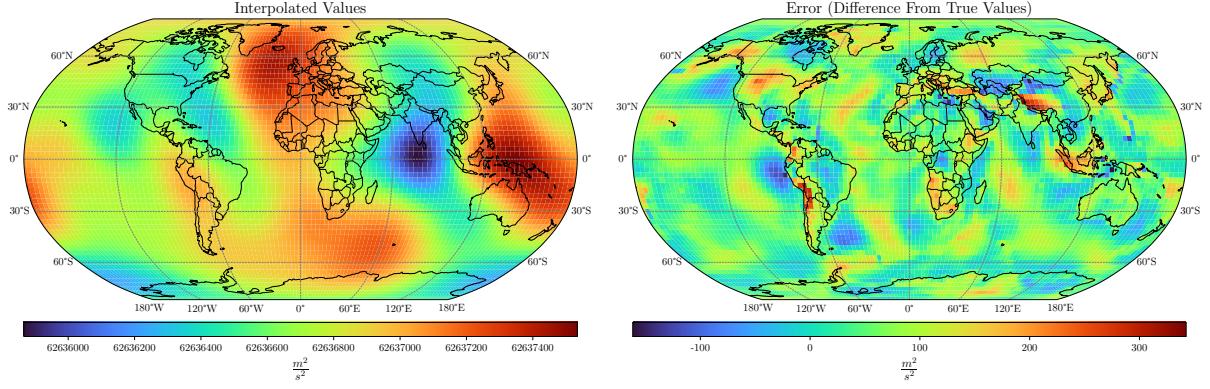


Figure 16: Results of Logarithmic kernel with GCV (FOT) method.

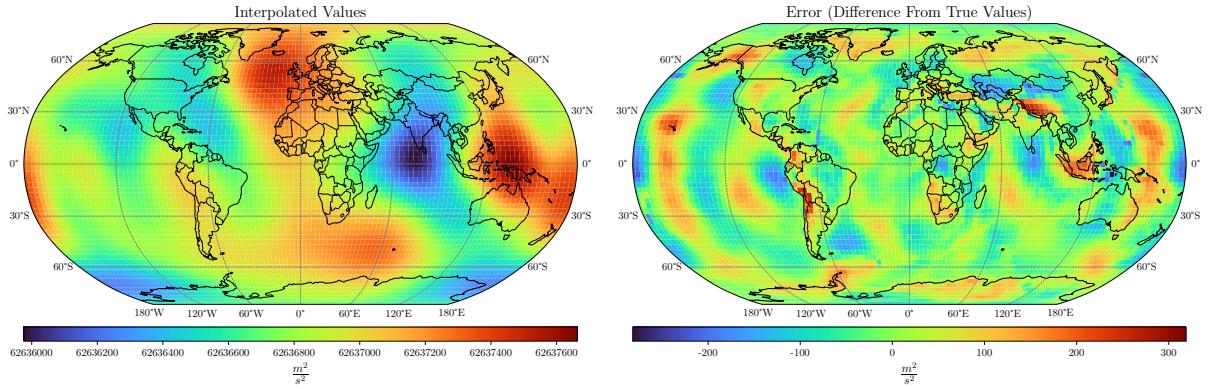


Figure 17: Results of Logarithmic kernel with L-Curve (FOT) method.

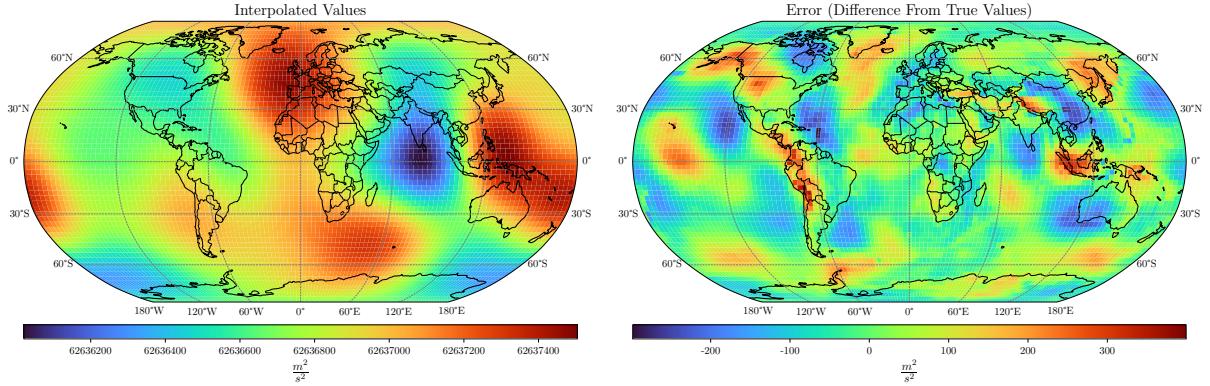


Figure 18: Results of Logarithmic kernel with VCE (FOT) method.

3.4 Abel-Poisson With Noise

In this case, white-noise with standard deviation of $200 \frac{m^2}{s^2}$ was added to the input values, the noisy input data is shown in figure 19. Interpolated and error values are shown in the figures 20 to 24. Other results, including parameter h , regularization parameter, Mean and norm (L^2) of error are shown in table 4. As it can be understood from the table,

TSVD and Cholesky methods were able to achieve the best results. Furthermore, results from other methods are too smooth and not reliable. Between FOT methods, VCE could result in better answer than other two methods.

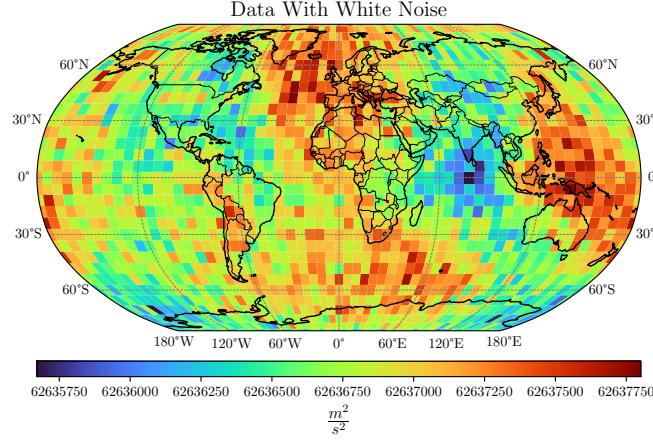


Figure 19: Input data after adding white noise with standard deviation of $200 \frac{m^2}{s^2}$.

Table 4: Results of interpolation using Abel-Poisson kernel and added white-noise (units of norm and mean values are $\frac{m^2}{s^2}$).

Method	Cholesky	TSVD	GCV (FOT)	L-Curve (FOT)	VCE (FOT)
Regularization Parameter	—	145	$3.50 * 10^{-6}$	$5.33 * 10^{-4}$	$2.03 * 10^{-7}$
h	0.065	0.381	0.121	0.201	0.173
Mean of Error	0.1131	1.1173	4.1071	13.5542	-3.1657
Norm of Errors	5840.2691	5735.7724	8289.586	21767.4379	6836.7964

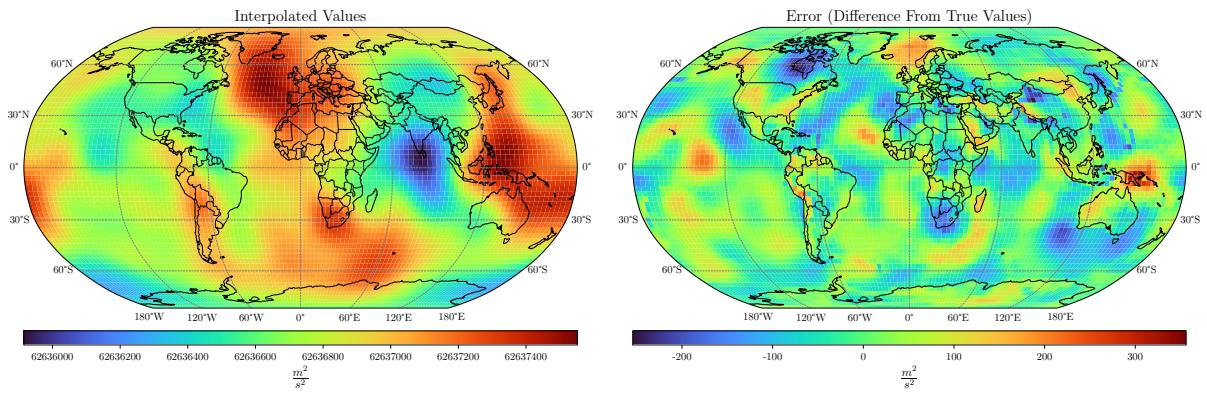


Figure 20: Results of Logarithmic kernel with Cholesky decomposition and added white-noise.

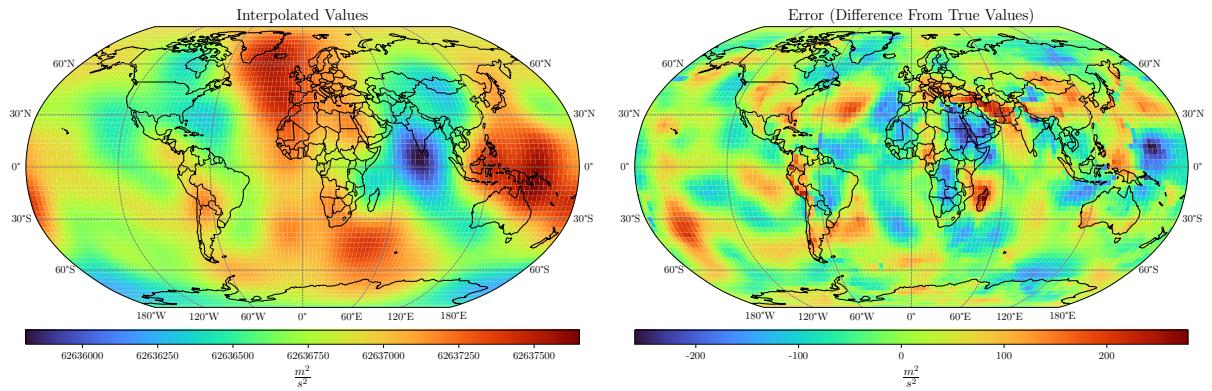


Figure 21: Results of Logarithmic kernel with TSVD method and added white-noise.

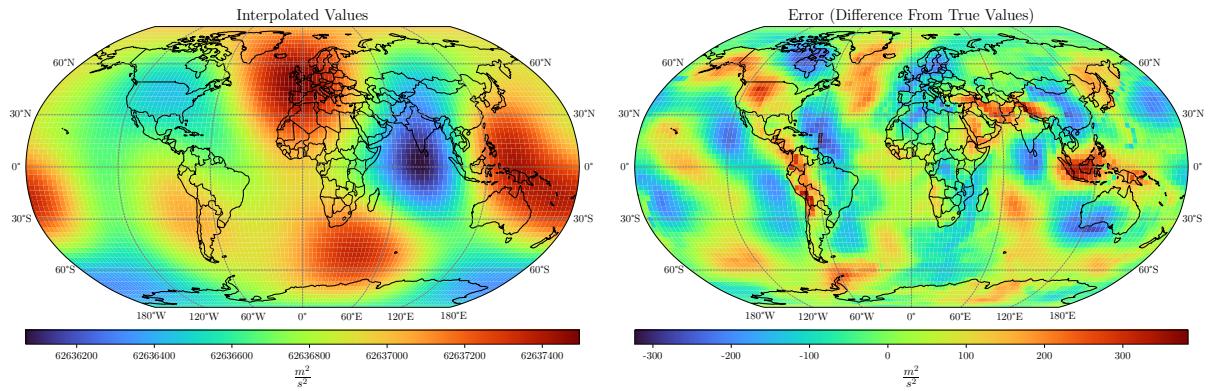


Figure 22: Results of Logarithmic kernel with GCV (FOT) method and added white-noise.

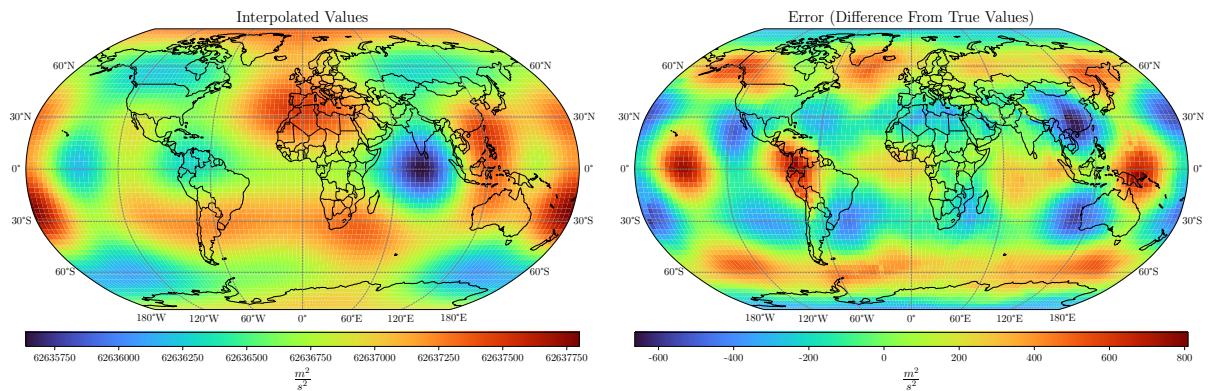


Figure 23: Results of Logarithmic kernel with L-Curve (FOT) method and added white-noise.

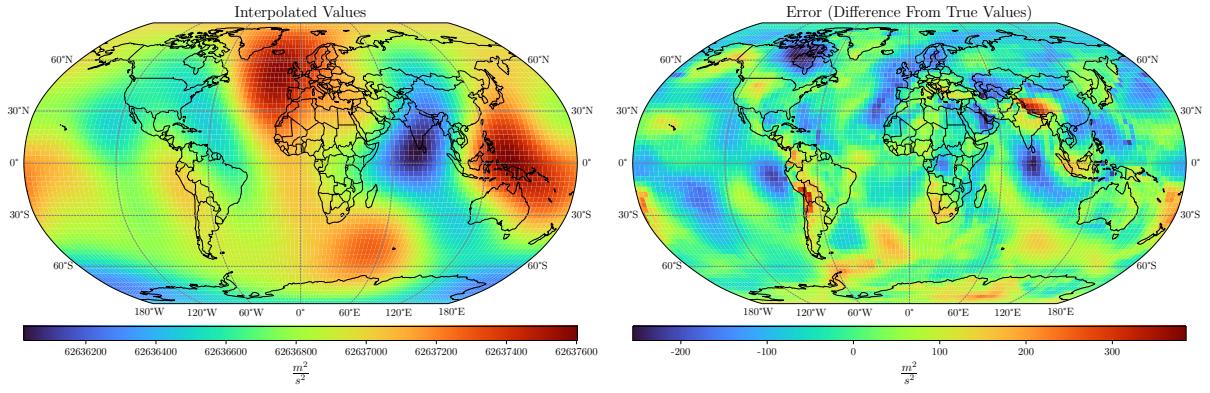


Figure 24: Results of Logarithmic kernel with VCE (FOT) method and added white-noise.

4 Appendix

As mentioned before, for determining the value of h in each kernel, a simple 3-step cross-validation method was implemented. As an example the norm mentioned in equation 10 for different values of h using TSVD method and three kernels are shown in figures 25 to 27.

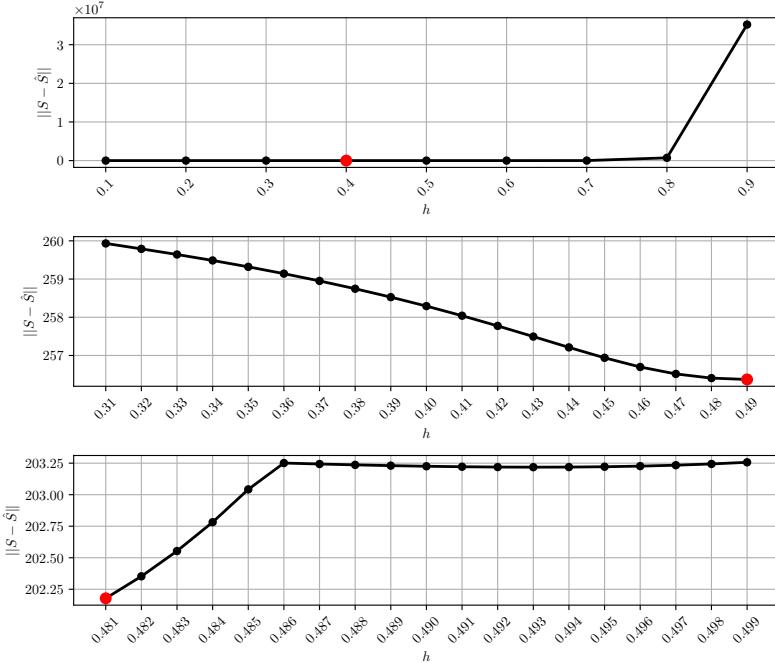


Figure 25: Results of cross-validation method used for determining h using Abel-Poisson kernel with TSVD method.

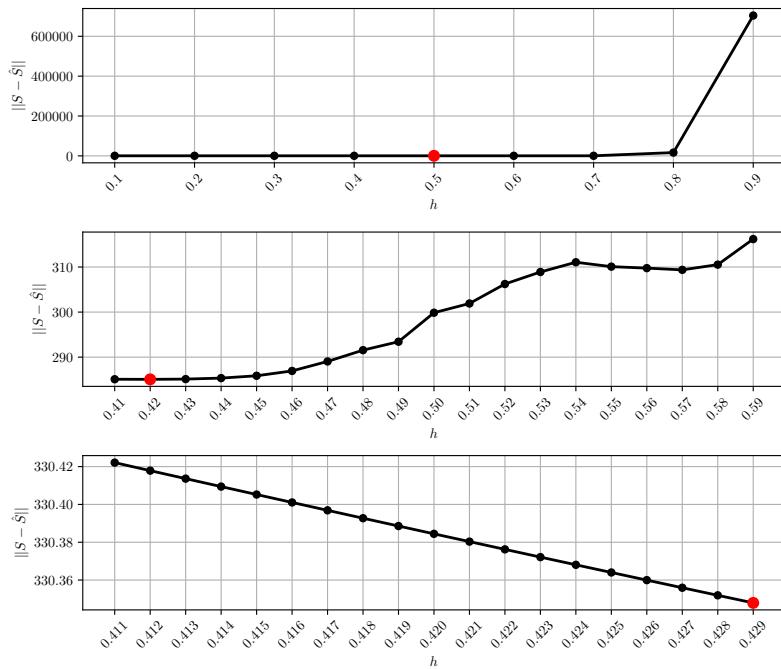


Figure 26: Results of cross-validation method used for determining h using Singularity kernel with TSVD method.

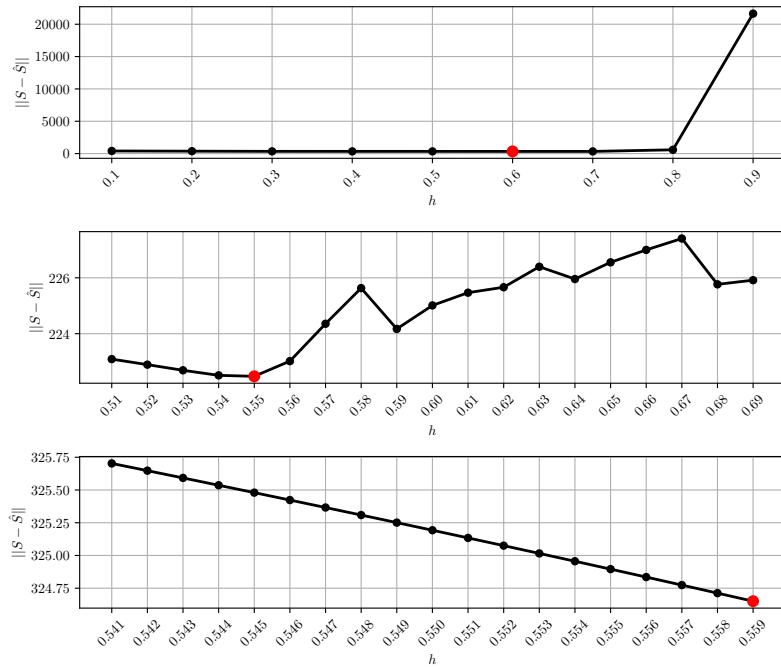


Figure 27: Results of cross-validation method used for determining h using Logarithmic kernel with TSVD method.