AAS 21-393

American Astronautical Society

American Institute of
Aeronautics and Astronautics

APPLICATIONS OF PHYSICS-INFORMED NEURAL NETWORKS FOR GRAVITY FIELD MODELING

J. R. Martin , and H. Schaub

AAS/AIAA Space Flight Mechanics Meeting

Charlotte, North Carolina January 31–February 3, 2020

AAS Publications Office, P.O. Box 28130, San Diego, CA 92198

APPLICATIONS OF PHYSICS-INFORMED NEURAL NETWORKS FOR GRAVITY FIELD MODELING

J. R. Martin [*] and H. Schaub[†]

Accurate and computationally efficient dynamics are paramount for high-accuracy astrodynamics simulation and spacecraft control. To yield such dynamics, researchers need high-fidelity representations of the gravitational potential from which trajectories are propagated. Traditionally these models are constructed analytically using spherical harmonics, mascons, or polyhedrons, and accelerations are computed by taking the gradient of the potential function. While these representations are convenient for theory, they each come with unique disadvantages in application. Broadly speaking, analytic representations are often not compact, requiring thousands or millions of parameters to adequately model high-order features in the environment. In some cases, analytic models can also be operationally limiting by diverging near the surface. Moreover, these representations can expensive to regress, requiring large volumes of carefully distributed data which may not be readily available in new environments. To combat these challenges, this paper aims to shift the discussion of potential representations away from analytic models and towards computational models. Within the past decade alone there have been dramatic advances in the field of deep learning which may help to bypass some of the limitations inherit to the analytics of existing gravity models. Specifically, this paper investigates the use of a recent type of neural network, Physics-Informed Neural Network (PINN), to represent the gravitational potential of a celestial body and predict consequent dynamics. The findings presented suggest that these neural network representations can offer advantages over their analytic counterparts in model compactness, regressive ability, and computation speed.

INTRODUCTION

The search for an efficient representation of the gravitational potential continues to be a dominant discussion within the astrodynamics community. Starting in 1933, Brillouin proposed using the spherical harmonic basis to represent the gravitational field of Earth [1]. This representation is especially efficient at capturing Earth's largest gravitational perturbation – its oblateness or J_2 – as well as other large-scale perturbations making it the defacto standard within the field. The success of the spherical harmonic representation has prompted multiple generations of experiments and missions like GRACE, GRAIL, and GRACE-FO to resolve exponentially higher-order spherical harmonic representations of the potential[2, 3]. Today the highest-fidelity spherical harmonic representation of Earth is EGM2008 which combines satellite ranging data as well as ground base measurements to reach a maximum spherical harmonic degree of 2,190 – a model with over 4 million parameters [4].

[*] NSF Graduate Research Fellow, Ann and H. J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, 431 UCB, Colorado Center for Astrodynamics Research, Boulder, CO, 80309.

[†] Glenn L. Murphy Chair of Engineering, Ann and H. J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, 431 UCB, Colorado Center for Astrodynamics Research, Boulder, CO, 80309, AAS Fellow, AIAA Fellow.

While spherical harmonics are ubiquitous for most spacecraft operations, they become less reliable in landing or touch-and-go maneuvers. Specifically, the dynamics imparted from a spherical harmonic representation begin to diverge once inside the Brillouin sphere – the sphere that circumscribes the celestial body of interest. With the recent push for small body exploration, this becomes a problem as many asteroids are irregularly shaped and their surface often exists within that sphere. Ellipsoidal harmonics are a more general extension of the spherical harmonics representation which allow for a tighter circumscribing ellipsoid about the small body, but they remain prone to the same diverging dynamics once inside of the circumscribing ellipsoid [5].

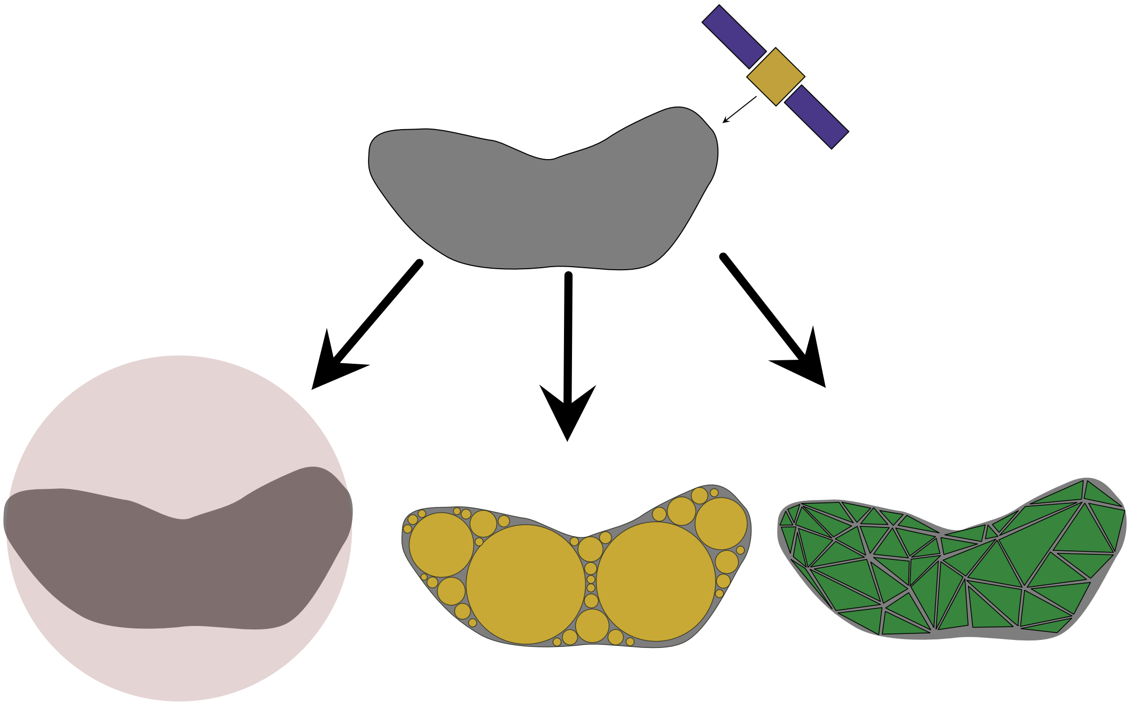


Figure 1: Popular gravity modeling options include spherical harmonics (left), mascons (center), or polyhedral (right) representations of the potential.

Multiple alternative potential representations exist to combat this divergence. One alternative is to use mascons to represent the potential. Assuming a shape model of the body exists, the shape can be algorithmically packed with discrete mass elements, whose individual contributions summed together can form a representation of the potential [6]. While mascons yield stable dynamics inside the circumscribing sphere, the representation begins to generate inconsistent dynamics as the spacecraft approaches the surface where the discrete nature of the mascons become increasingly apparent [7]. In 1996, Werner and Scheeres introduced the polyhedral gravity model to resolve this problem. The polyhedral model generates a potential directly from the shape model, without additional discretization. This allows for a stable solution all the way down to the surface of the body [6]. This stability comes at the cost of assuming an underlying density profile for the body, which is often difficult to uniquely estimate [8]. Each of these potential representations are powerful under certain conditions, but they all come with unique limitations or assumptions from the analytics. Ideally there can exist a representation that is flexible enough to capture the most dominant features, robust enough to work across all operational conditions, and require no assumptions about the underlying body – all-the-while remaining relatively inexpensive to compute.

This paper attempts to address some of these challenges using machine learning representations of the gravitational environment. Machine learning is a field markedly successful at constructing accurate models from data observed in complex environments, typically through the use of deep artificial neural networks [9]. These neural network models learn an optimal mapping between some desired input variable and output variable, while making no inherent assumptions about the problem formulation. This is immediately advantageous to the gravity modeling problem, as the classical analytics are precisely what cause divergence within the circumscribing sphere, or require assumptions about the shape and density of the body. By having an artificial neural network learn the mapping between position and acceleration from the data alone, there is not an a priori expectation of how the model must be represented. Instead the network's only goal is to resolve a basis that is maximally efficient in representing the data rather than the mathematics.

Using machine learning methods to model small-body gravity fields has already been demonstrated with some level of success in the literature [10, 11, 12]. These works predict small bodies dynamics with machine learning models, but they fail to thoroughly compare the advantages and disadvantages of these representations with other analytic models. For instance, it is unclear what conditions are required to regress a sufficiently accurate neural network representation of the gravity field, or if the neural network representations are inherently richer/more representationally compact than their analytic counterparts. At what point do these networks begin to overfit or diverge? Can they regress gravitational perturbations faster than other approaches?

This paper attempts to broaden the discussion about robustness of the neural network representation of the gravity field and explore new constraints that can improve performance. While artificial neural networks are strong candidates to model high-dimensional problems, their flexibility often comes at the cost of interpretability and predictability. Because these machine learning models are data driven, there are no obvious physical insights that can be drawn about the learned representation – that is until recently. In 2019, Raissi et. al. introduced the physics-informed neural network (PINN) which injects the underlying physics and dynamics into the training process of a traditional network [13]. Using automatic differentiation, the PINNs are trained not only to prioritize an accurate mapping from an input space to an output space, but also to enforce that the solution resolved by the network satisfies some underlying differential equation and boundary conditions. PINNs are therefore able to unify the flexibility of a machine learning model with centuries of analytic insight. This makes the PINN a natural candidate to apply to the gravity modeling problem. Rather than forcing a solution to be of a preconstructed analytic form, a PINN can generate a flexible solution and simply ensure that neural network representation satisfies the equations of Newton and Poisson.

The paper aims to investigate the compactness and rigor of these machine learning gravity solutions for two representative gravitational bodies: Earth and 433-Eros. Earth is an interesting body as it has a dominant near-ellipsoidal shape which is well modeled with spherical harmonics, but the higher-order perturbations form an almost random surface topology that is challenging to capture using a periodic basis. Alternatively, 433-Eros is non-spherical asteroid whose dynamics are commonly represented using a polyhedral model. Together these bodies allow for a comprehensive comparison of the neural network model in contrast to other popular representations.

The paper is divided into the following sections: First, the paper discusses the current state of spherical harmonic models used for Earth – investigating the performance of these models at varying levels of fidelity. These results are the baseline metrics which the traditional and physics-informed neural network models are compared against. Following this characterization is a discussion of the neural network representations alongside information about the training data and hyperparameters used to generate them. The paper then compares the performance of the neural network representations to that of the spherical harmonic representation focusing specifically on arguments of representational compactness, generalization ability, regressive efficiency, and finally evaluation speed. The paper concludes with an brief exploration of how the network gravity representations can be applied the asteroid 433-Eros along with a discussion of future work.

SPHERICAL HARMONIC REPRESENTATION

In 1933 Brillouin introduced the spherical harmonic representation of the gravitational potential:

$$U(\bar{\mathbf{r}}) = \frac{\mu}{r}\sum_{l=0}^{l}\sum_{m=0}^{l}\left(\frac{R}{r}\right)^{l} P_{l,m}[\sin(\phi)]\left[C'_{l,m}\cos(m\lambda) + S'_{l,m}\sin(m\lambda)\right] \tag{1}$$

where r is the position, μ is the gravitational parameter of the body, R is the circumscribing radius of the celestial body, l is the degree of the spherical harmonic model, m is the order of the spherical harmonic model, $C'_{l,m}$ and $S'_{l,m}$ are the regressed spherical harmonic coefficients, λ is the geodetic longitude, ϕ is the geodetic longitude, and $P_{l,m}$ are the associated Legendre polynomials.

This representation is efficient at modeling large, predominately spherical celestial bodies as it captures one of the most dominant perturbing features (the oblateness) with the first non-zero term after the point mass. While this is a significant advantage of the spherical harmonic representation, its modeling efficiency does not persist at higher degrees. To demonstrate, consider the next most dominant gravitational perturbations beyond the oblateness of the body by removing the point mass and degree/order 2 contributions from the potential and taking the gradient:

$$\delta a_{l,i} = |\nabla U_{1000}(\mathbf{r}_{i,DH}) - \nabla U_{l}(\mathbf{r}_{i,DH})| \tag{2}$$

where U is the scalar gravitational potential, the l denotes the maximum degree of the potential expansion, $\delta a_{i,l}$ is the norm of the high-order contributions of the acceleration generated at $\mathbf{r}_{i,DH}$, the i-th position vector in an Driscoll and Healy (DH) grid which is uniformly sampled every 0.5 degrees in latitude and longitude [14]. U_{1000} is constructed from the EGM2008 gravity model and is taken to be ground truth. To avoid singularities at the poles, Pines' algorithm is used to take the gradient of the potential [15].

Figure 2 shows the values of $\delta a_{2,i}$ across Earth and reveals that the next most dominant gravitational perturbations beyond J_2 are discontinuous features in the crust like tectonic shelves and mountain ranges like the Himalayas and Andes. These perturbations, or gravitational features, are what will be used to evaluate the efficacy of the spherical harmonic gravity model and, later, the neural network gravity models.

Specifically, a mean root-squared error (MRSE) metric is introduced to evaluate how well the spherical harmonics model capture these high-order features:

$$\text{MRSE}(\mathcal{A}) = \frac{1}{N}\sum_{i=1}^{N}\delta a_{l,i} \quad \delta a_{l,i} \in \mathcal{A} \tag{3}$$

where N is the total number of position vectors in the DH grid, and $\delta a_{l,i}$ are drawn from one of three sets: The first set includes all accelerations within the DH grid, $\mathcal{A} : \{\delta a_{i,l} \quad \forall i < N\}$. The second

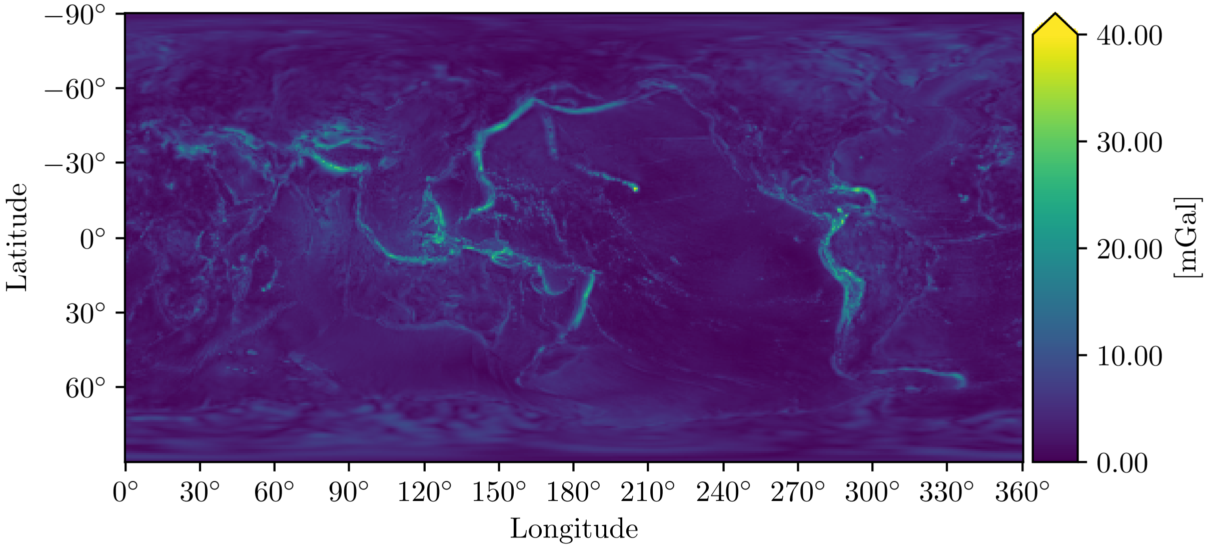


Figure 2: Map of $\delta a_{2,i}$

is a feature set, generated by masking data within the DH grid that exceed 2 standard deviation of the mean perturbing acceleration ($\delta\bar{a}$) or $\mathcal{F} : \{|\delta a_i - \delta\bar{a}| > 2\sigma(\mathcal{A})\}$. The final set is a mask of the compliment of those features, $\mathcal{C} : \{\mathcal{A}/\mathcal{F}\}$. The mean root-squared error of δa_i within these sets are presented in Figure 3 as a function of spherical harmonic fidelity.

The significant relative error of MRSE($\mathcal{F}$) as compared to MRSE($\mathcal{A}$) demonstrates the stark inefficiency of the spherical harmonic representation at capturing the most dominant gravitational features. Note that $|\mathcal{F}| << |\mathcal{C}|$ so MRSE($\mathcal{A}$) more closely resembles MRSE($\mathcal{C}$), but the dynamical significance of $\mathcal{C}$ is much lower than that of $\mathcal{F}$. The clear separation between MRSE($\mathcal{A}$) and MRSE($\mathcal{F}$) showcases how spherical harmonics fail to capture information in order of dynamical significance. Instead, spherical harmonics prioritize fitting to prescribed geometries that are not obviously present in the system. As a consequence, the spherical harmonic model must superimpose many high-order frequencies before capturing these features – the three dimensional analog to Gibbs' phenomenon [16]. In some circumstances this may not be seen as a problem. When a sufficiently high-fidelity model exists and the researcher is not computationally limited, spherical harmonics will eventually converge even over the discontinuous features. However, on-board, computational resources may be limited and a high-fidelity spherical harmonic model may not exist for the body in question. In these conditions, operations over short time scales near large perturbations, like in a touch-and-go or landing maneuvers, could be negatively affected by spherical harmonics inability to efficiently represent the most dominant features.

MACHINE LEARNING REPRESENTATIONS

Traditional Neural Network

Artificial neural networks are a series of learned, non-linear transformations that map data from an input space to a desired output space by minimizing a prescribed loss function such as mean

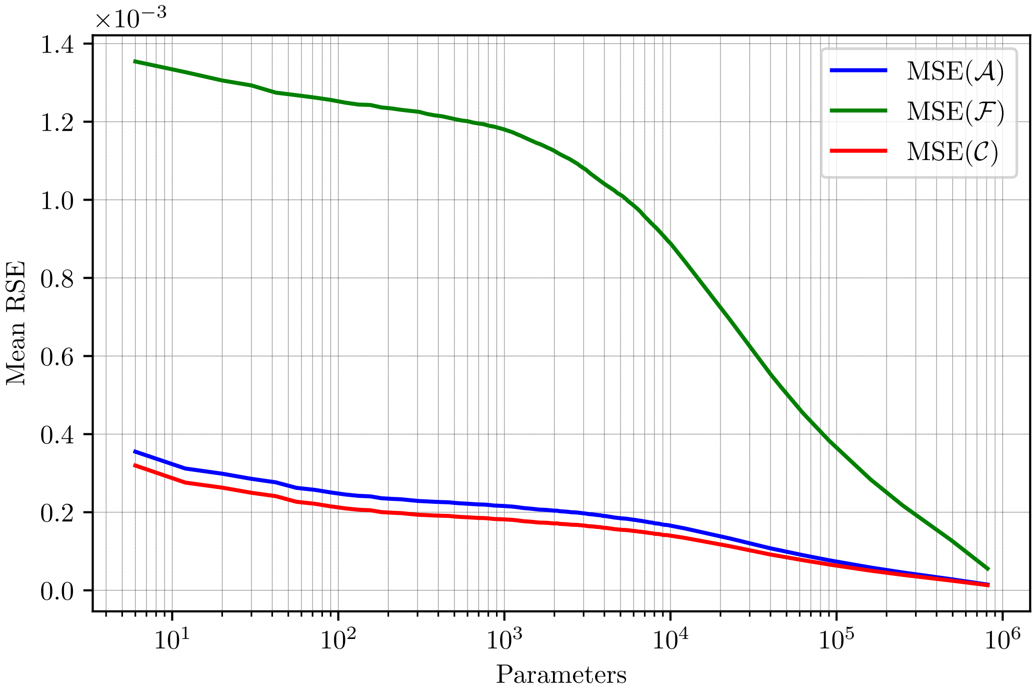


Figure 3: Plot of MRSE$_i$ as a function of total parameters, p, in used the spherical harmonic gravity model where $p = l(l+1)$.

squared error:

$$\mathcal{J}(\mathbf{\Theta}) = \frac{1}{N_f} \sum_{i=1}^{N_f} \left|(y_i - \hat{y}(x_i|\mathbf{\Theta})_i)^2\right| \tag{4}$$

where y_i is the true output, $\hat{y}_i$ is predicted output by the artificial neural network, $\mathbf{\Theta}$ is the vector of trainable weights, $\mathbf{w}$, and biases, $\mathbf{b}$, of the network, and N_f is the total number of points used to train the network.

The networks for this paper are constructed as a series of densely connected hidden layers with N nodes per layer:

$$h_i^{(k)} = \sigma\left(w_{ij}^{(k-1)} h_j^{(k-1)} + b_i\right) \tag{5}$$

where $h^{(k)}$ is the k-th hidden layer, i is the node in the layer, w_{ij} are the weights connecting the hidden layers, b_i are the biases attached to the nodes in the layer, and σ is the non-linear transformation (typically sigmoid, hyperbolic tangent, or rectified linear unit). Note that $h^{(0)} = x$, and $h^{(k_{\text{max}})} = \hat{y}$.

The neural network is trained by tuning the weights and biases to minimize Equation 4 such that:

$$\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathbf{\Theta}}(J(\mathbf{w})); \quad \mathbf{b}^* = \arg\min_{\mathbf{b}\in\mathbf{\Theta}}(J(\mathbf{b})) \tag{6}$$

which can be solved using a gradient descent algorithm like Adam or SGD [17, 18]:

$$\mathbf{\Theta}^{m+1} = \mathbf{\Theta}^m - \eta \nabla_{\mathbf{\Theta}^m} \mathcal{J}^m(\mathbf{\Theta}) \tag{7}$$

where η is the learning rate and m is the training iteration. This allows the network to iteratively update its weights such that it learns an optimal mapping from the input data to the output data. When applied to the gravity modeling problem, the input data x for the traditional neural network are the position vectors and the output data y are the acceleration vectors with the point mass and degree 2 contributions removed.

Physics-Informed Neural Networks

One of the disadvantages of traditional neural networks is that the networks do not account for the underlying dynamics of the solution they are attempting to represent. In the case of gravity modeling, the traditional neural networks do not know the acceleration they are attempting to model has a relationship with an underlying potential function U as expressed through

$$\mathbf{a} = -\nabla U \tag{8}$$

Consequently, the network will be trained agnostic to the fact that the force it represents is a conservative force and the potential needs be sufficiently smooth and continuous for sensible dynamics. Instead the network prioritizes minimizing error, with no respect for the physics.

In 2019, Raissi et. al. recognized this problem and suggested that the numerical, machine learning models do not need to be agnostic of the dynamics they are attempting to model [13]. Instead, they can make use the underlying differential equation guiding those dynamics to help identify a solution more efficiently while respecting the analytics. To this end, Raissi et. al. introduced the Physics-Informed Neural Network (PINN). PINNs inject analytic forms of a differential equation into the cost function of a neural network, and use machine learning methods like automatic differentiation to enforce the dynamics. For the gravity modeling problem, consider the following cost function:

$$J(\mathbf{\Theta}) = \frac{1}{N_f} \sum_{i=1}^{N_f} \left| \mathbf{a_i} - \nabla \hat{U}_i(x_i|\mathbf{\Theta}) \right|^2 \tag{9}$$

where $\mathbf{a_i}$ is the measured acceleration at position $\mathbf{x_i}$, and $\hat{U}_i(\mathbf{x_i})$ is the neural network predicted potential solution at position $\mathbf{x_i}$.

This setup is extremely similar to that of a traditional network with the small difference: the neural network output is not the acceleration vector. The output is instead a model of the potential ($\hat{U}$), from which the gradient is taken using automatic differentiation and *then* compared against the measured acceleration. The cost function is therefore enforcing that the predicted accelerations must be a byproduct of a more fundamental, and physically realistic, solution – thereby using analytics to more efficiently train the network.

This is the only physics-informed constraint applied to the networks in this paper. It is worth noting that additional physics constraints could be applied. For example, the cost function in Equation 9 only accounts for errors in the acceleration. The cost function could also include a penalty for violating the boundary condition $U(\mathbf{x}) = 0$ as $|x| \to \infty$, or for mis-modeling the potential function itself (assuming an accurate representation of the potential already exists) i.e.

$$J(\mathbf{\Theta}) = \frac{1}{N_f} \sum_{i=1}^{N_f} \left| \mathbf{a_i} - \nabla \hat{U}_i(\mathbf{x_i}|\mathbf{\Theta}) \right|^2 + \left| U(\mathbf{x_i}) - \hat{U}(\mathbf{x_i}|\mathbf{\Theta}) \right|^2 \tag{10}$$

Network Type	N	**Network Params**	l	**Spherical Harmonics Params**
NN	20	3,083	55	3,080
NN	40	11,763	110	12,210
NN	80	45,923	215	46,440
PINN	20	3,040	55	3,080
PINN	40	11,680	110	12,210
PINN	80	45,760	215	46,440

Table 1: Total number of free parameters in the neural network representations matched with the spherical harmonic degree which shares a similar number of parameters

Raissi et. al. shows a variety of these additional constraints applied to different partial differential equations which further improve accuracy. For this paper, only Equation 8 is applied to cost to generate a conservative estimate of how much these constraints affect performance. Future work will explore the impact of including preexisting knowledge of the potential, as well as other dynamics and boundary considerations.

Architectures

Three network architectures are tested in this study. Each network consists of 8 densely connected hidden layers with N nodes per layer, using a hyperbolic tangent activation function, and a weight initialization defined in Glorot and Bengio [19]. The number of nodes per layer were chosen specifically so that the total number of trainable parameters in the network would share approximately the same number of parameters as a particular degree spherical harmonic model as shown in Table 1. This way the modeling accuracy can be compared using the same representational compactness (i.e. number of parameters required to describe the model).

All data fed to the network is preprocessed by scaling all input and output data using a min-max transformation to fit between the interval $[-1, 1]$, and weights in the network are updated using the Adam optimizer. Assume each network was trained for 100,000 epochs with a batch size of 40,000 unless otherwise specified. All networks were trained using mixed precision in Tensorflow 2.1* on a Nvidia RTX 2060 graphics card.

REPRESENTATIONAL COMPACTNESS

One of the primary considerations when approximating a physical system is how many parameters are necessary to capture a certain level of accuracy. As shown in Figure 3, a spherical harmonic representation requires on the order of 10^4 coefficients/parameters before it begins converging on the dominant gravitational features. This section aims to demonstrate how many parameters a neural network representation requires to capture comparable performance.

The representational efficiency of the network models is tested by training each network on 950,000 position/acceleration vector pairs which are drawn from a uniform distributions in altitude ($\mathcal{U}(0, 420)$km), latitude ($\mathcal{U}(0, 180)$ degrees) and longitude ($\mathcal{U}(0, 360)$ degrees). Once trained, the networks are evaluated using the same testing data as is used to evaluate spherical harmonic efficiency – $\delta a_{2,i} \in \mathcal{A}, \mathcal{F}, \mathcal{C}$. The mean root-squared error for the traditional neural networks with

*https://www.tensorflow.org/

$N = \{10, 20, 40, 60, 80\}$ and the physics-informed networks of $N = \{10, 20, 40\}$ are presented in Figure 4.

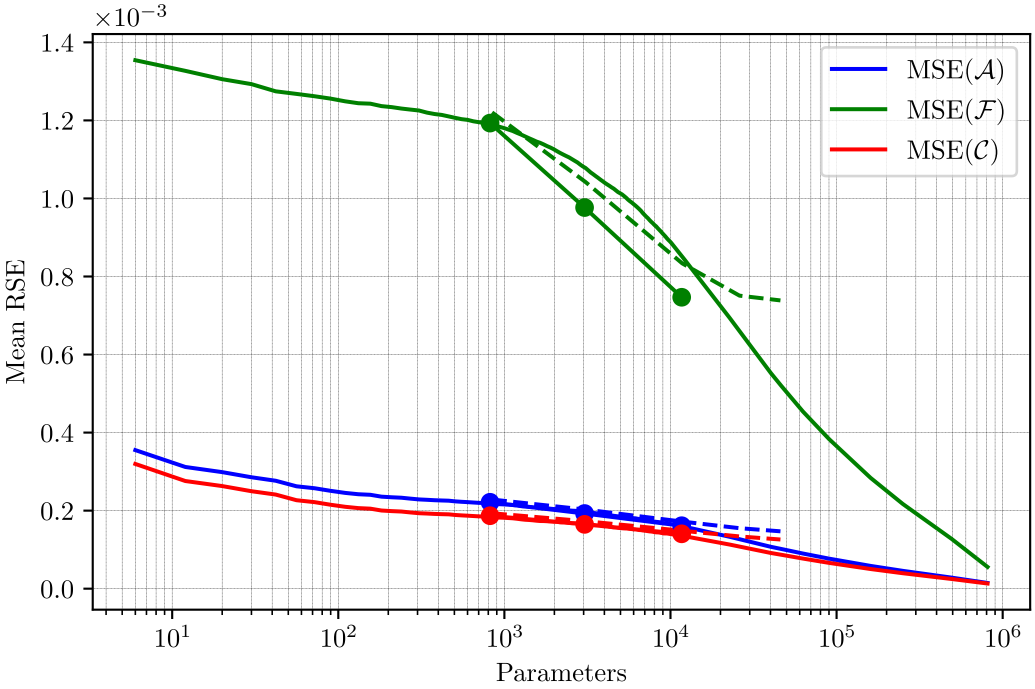


Figure 4: Plot of MRSE$_i$ as a function of total model parameters. Dashed lines represent traditional neural networks. The lines with circle markers represent the physics-informed neural networks.

Figure 4 demonstrates that there exists a relatively wide regime ($10^3 - 10^4$ parameters or between $l = 30$ to $l = 100$) in which neural networks outperform their spherical harmonic counterparts at capturing features. While the performance across the sets $\mathcal{A}$ and $\mathcal{C}$ remained nearly identical in both the traditional and physics-informed neural network cases to that of the spherical harmonic representation, the performance in $\mathcal{F}$ offered up to a maximum of 50% reduction in model size for the PINN case of $N = 20$.

Figure 4 also provides helpful insight to interpreting the value of using physics-informed constrains when training the networks. While the traditional neural networks offered comparable performance to their spherical harmonic counterparts on average, there exist cases where the neural networks do worse. Particularly in the $N = \{10, 60, 80\}$ cases, the traditional neural network has higher error even in $\mathcal{F}$. The physics-informed cases offer a different story, consistently outperforming their analytic counterparts in $\mathcal{F}$. This aligns with expectation as the physics-informed networks are attempting to find solutions to a much more fundamental (and continuous) function – the potential. The dynamics included in the cost of Equation 9 gives a richness to the physics-informed network representation, that is otherwise unobserved by the traditional networks – allowing the PINN to make more efficient use of the same amount of training data.

It is worth noting that for both the traditional and physics-informed cases, the representational success is a function of network capacity and amount/quality of training data. Deep neural networks are capable of modeling any function so long as the network is sufficiently wide and/or deep (i.e. has high capacity) [20]. This is why in Figure 4 the error decreases with the number of parameters – a behavior generally true of any regressed model. If the networks are not given adequate amounts of sufficiently diverse data, however, the network capacity will not be the factor which limits performance. This is seen in the $N = \{60, 80\}$ cases of the traditional network representation in Figure 4. Despite having many more parameters in the model, there is simply not enough data to properly resolve the optimal weights and biases in that network. The physics-informed network does not suffer as much as the traditional network, because its additional dynamics constraint in the cost function allow the data to be "triple-counted" thanks to the partial derivatives of the potential. Recognizing these architecture and data requirements, it is reasonable to assume that given even more data, each of these networks would perform better than observed in Figure 4 and the representational advantage of the neural network representations would grow even more apparent.

GENERALIZATION

The analysis thus far has only focused on accuracy at the Brillouin sphere, but it is also important to consider how well the network models generalize to different orbit regimes. Generalization to higher altitudes is tested using the same isotropic 0.5 degree latitude/longitude DH grid as before, but varying the altitude of that grid between 0 to 500 km to generate $\mathcal{A}_m$ where m is the altitude of the grid. $\mathcal{F}_m$ is also generated by keeping the same points in the original set $\mathcal{F}$ and observing modeling error as those data position themselves at higher altitudes. Measuring the MRSE of $\mathcal{A}_m$ and $\mathcal{F}_m$ grants insight into at what altitudes the traditional network representation begins to weaken. Figure 5 shows the errors associated with $\mathcal{A}_m$ and Figure 6 shows the errors associated with $\mathcal{F}_m$. Both figures include a histogram showing the radial distribution of the data which are used to train the traditional networks.

In general, the traditional neural networks outperform spherical harmonics in the lower altitudes feature set, $\mathcal{F}_m$, where the perturbations are more discontinuous. Spherical harmonics surpass the neural networks in higher altitudes where discontinuities are attenuated. The success of the neural networks at low altitudes is attributed to network training process. Given the MSE loss function, the network will prioritize fitting the model to areas where the perturbations are highest (at the surface). Until such perturbations are adequately captured, the neural network will not prioritize regions in the data where model errors are less variable.

The generalization analysis assumes that the training data distribution is uniformly distributed in altitude as shown by the histograms in Figures 5 and 6. The uniform training distribution, however, is a luxury that is unlikely to be found in application. More realistic training datasets might include data that is predominantly taken at the surface of a body with infrequent observations at higher altitudes (surface constrained) or data taken predominately from orbit with infrequent observations near the surface (orbit constrained). To study generalization behavior under more realistic data conditions, two new training sets are generated using the following exponential probability distribution defined by:

$$\mathbb{E}(x, x_0, \beta) = \exp\left\{-\frac{x - x_0}{\beta}\right\} \tag{11}$$

where x_0 is the reference altitude (0 km for surface constrained and 420 km for orbit constrained), and β is the scaling parameter responsible for determining how quickly the exponential decays. For both datasets $\beta = 42$ and 950,000 data are sampled. Note that to prevent sampling from inside the circumscribing sphere, the distribution of the orbit constrained dataset is restricted to $x \in [0, 420]$ km, whereas the surface constrained dataset can span $x \in [0, \infty)$. The results of the

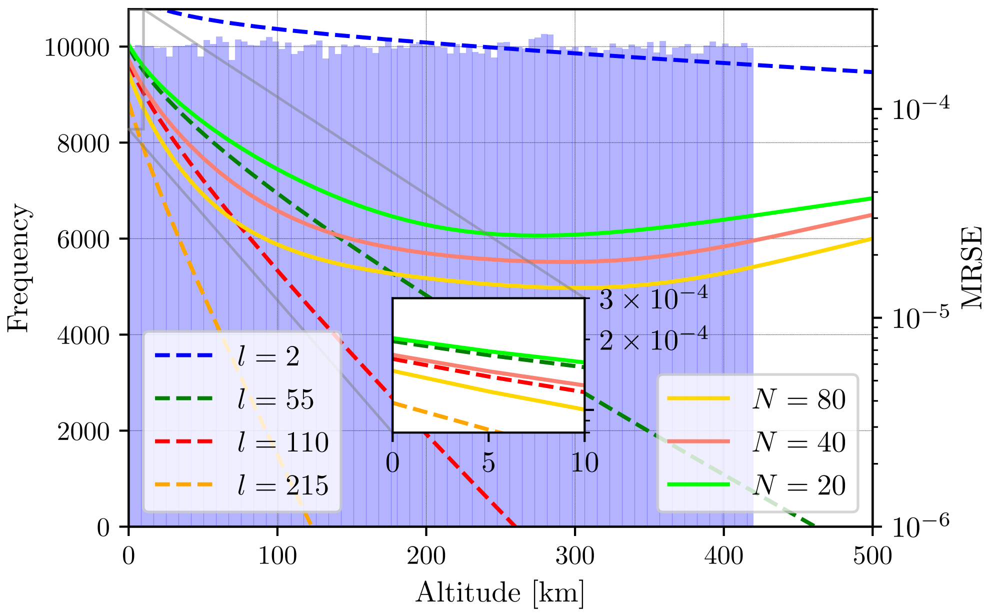


Figure 5: MRSE($\mathcal{A}_m$) as a function of altitude. Solid lines represent the neural network generalization error. Dashed lines represent the spherical harmonics generalization error. The blue histogram represents the training data distribution in altitude that is used to train the traditional neural networks.

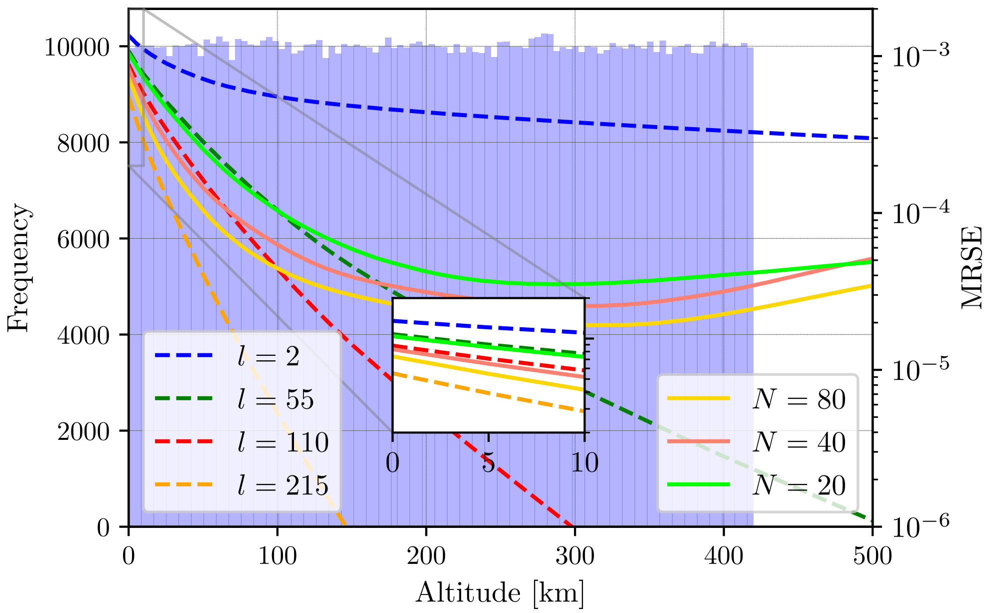
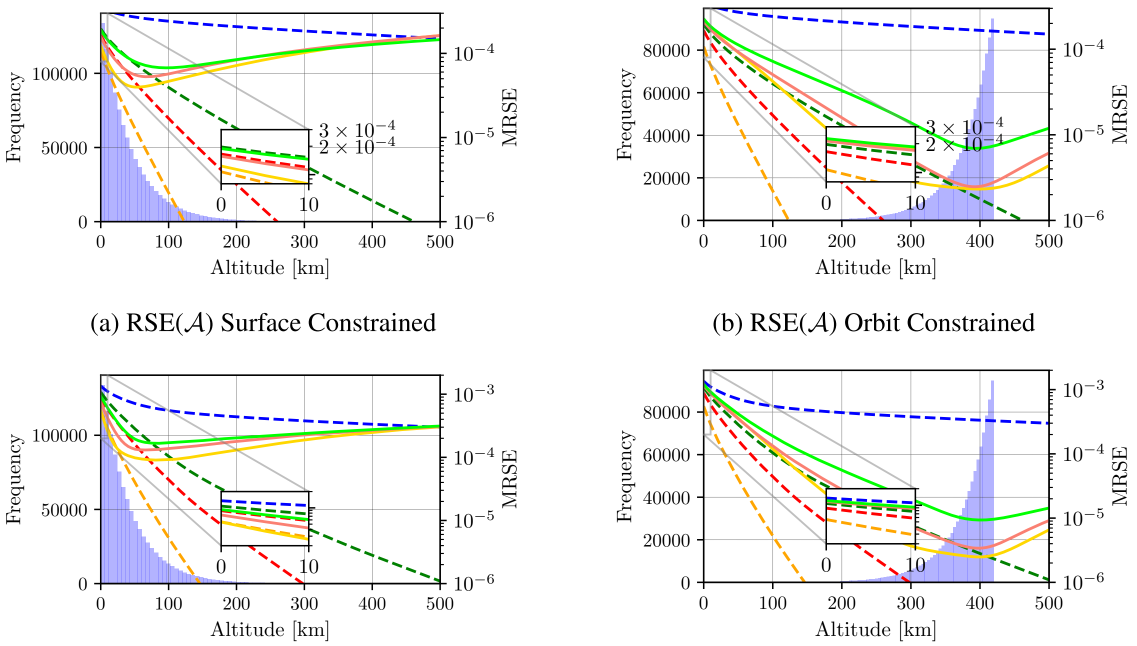


Figure 6: MRSE($\mathcal{F}_m$) as a function of altitude. Solid lines represent the neural network generalization error. Dashed lines represent the spherical harmonics generalization error. The blue histogram represents the training data distribution in altitude that is used to train the traditional neural networks.



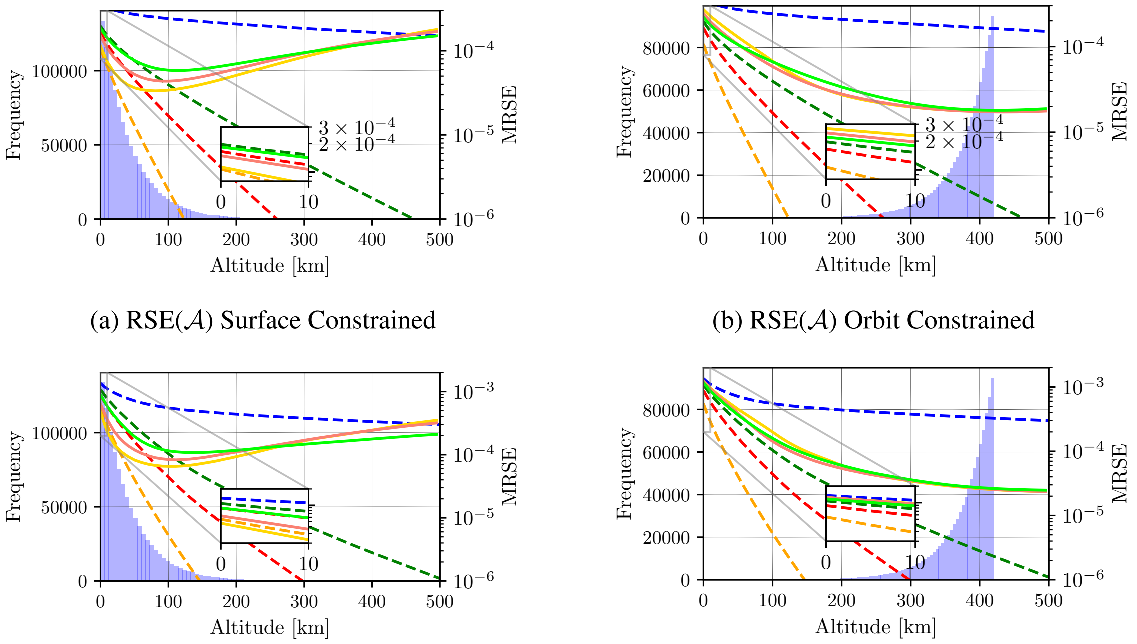
(a) RSE($\mathcal{A}$) Surface Constrained

(b) RSE($\mathcal{A}$) Orbit Constrained

(c) RSE($\mathcal{F}$) Surface Constrained

(d) RSE($\mathcal{F}$) Orbit Constrained

Figure 7: Training data distribution and error curves for traditional neural networks



(a) RSE($\mathcal{A}$) Surface Constrained

(b) RSE($\mathcal{A}$) Orbit Constrained

(c) RSE($\mathcal{F}$) Surface Constrained

(d) RSE($\mathcal{F}$) Orbit Constrained

Figure 8: Training data distribution and error curves for physics-informed neural networks

$N = \{20, 40, 80\}$ traditional networks are shown in Figure 7 and the $N = \{20, 40, 80\}$ physics-informed networks are shown in 8.

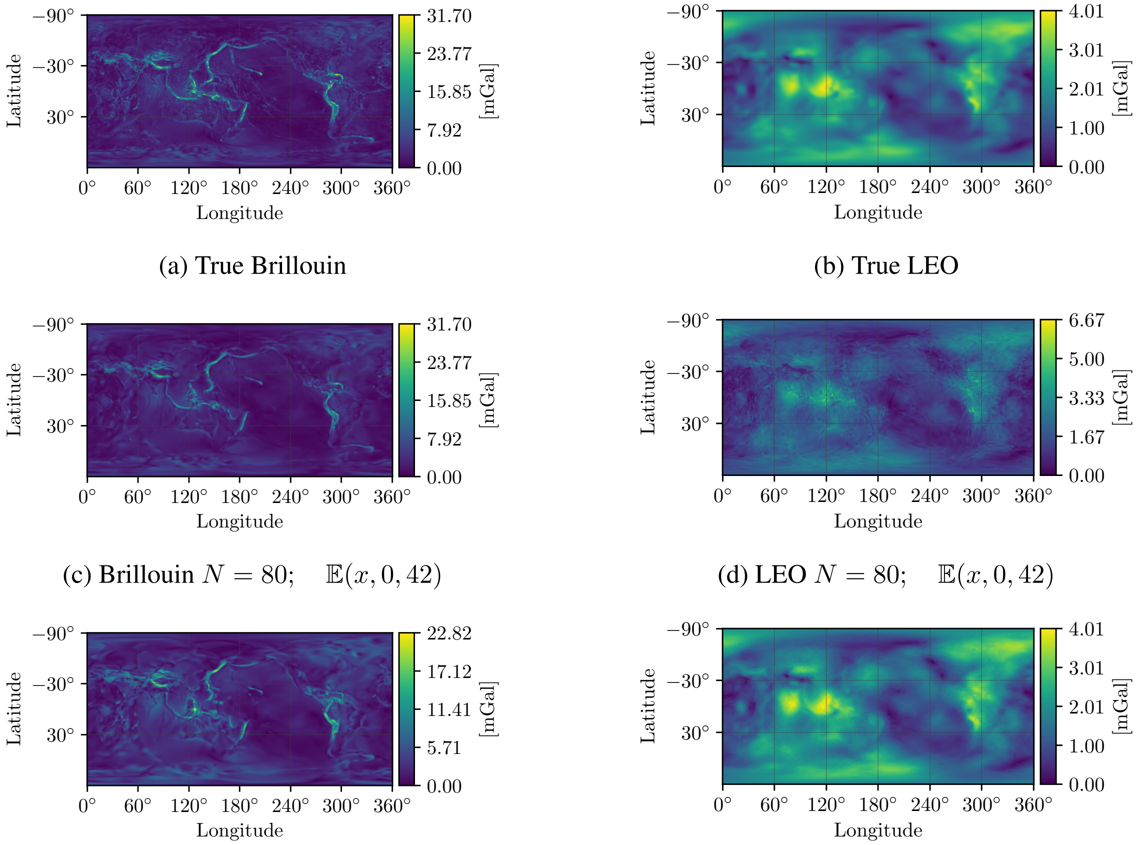
These figures yield a number of interesting findings. Figure 7c shows that the traditional neural network representations trained on the surface constrained dataset are better at capturing features up to approximately 30 km in the $N = 80$ case, 60 km in the $N = 40$ case, and 75 km in the $N = 20$ case. The performance of the physics-informed neural networks exceed this, reaching reaching 30, 70, and 95 km for its $N = \{80, 40, 20\}$ cases respectively as shown in Figure 8c. Despite the low-altitude accuracy of these networks, they begin to diverge from truth at higher altitudes, albeit their error remains below that of a simple $l = 2$ spherical harmonic model. There also appears to be a relationship between the value of N and how strongly the network begins to diverge. This relationship again highlights the challenges of training a high-capacity network without enough data to prevent overfitting.

The neural networks trained on the orbit constrained dataset offer a different narrative. These networks consistently perform worse than their spherical harmonic counterparts; however, they are surprisingly effective at generalizing to lower altitudes. The well-behaved extrapolation is likely a result of the smoothness of the perturbations as seen in a high-altitude orbit. The discontinuities from the Brillouin sphere are largely attenuated at high-altitude and only the most prominent perturbing features are preserved. The trained network therefore only extrapolates the dominant perturbations down to the Brillouin sphere, irrespective of the finer details of those features. The surface constrained models have the opposite, more difficult, problem, as they are tasked with extrapolating fine surface features to high-altitude. These extrapolation effects can be directly observed in Figure 9.

The last observation comes from the orbit constrained dataset where the physics-informed networks did worse at modeling the high altitude $\mathcal{A}_m$ and $\mathcal{F}_m$ sets than their traditional network counterparts. This runs counter to expectation, as the physics-informed networks are typically better at extracting more information from the data than traditional networks. This surprising result is likely because physics-informed constraints act as a means of regularization. Traditional neural networks will fit to the data regardless if the fit is physically realistic, granting greater modeling flexibility. Alternatively, physics-informed networks must constantly amend their representation to match the dynamics. Using the orbit constrained dataset, the physics-informed network likely found a simple and physically-realistic representation of the attenuated features; however, each time that representation is given a data point closer to the highly discontinuous Brillouin sphere (data that is significantly less well-behaved) that same model must scramble to reconcile a dynamical explanation of the unusual surface signature. The physics-informed neural networks therefore sacrifices model accuracy at high altitudes to provide a physical relevant explanation of the occasional Brillouin data.

REGRESSIVE ABILITY

The neural network models in each of the prior analyses are compared against the performance of a truncated model of the EGM2008 potential spherical harmonic representation. This is not an inherently fair comparison. The EGM model was constructed using large swaths of information taken from formerly regressed potential models, data collected from multiple satellite missions, and ground measurements. Conversely, the neural networks are trained with only 950,000 training data and nothing more. This is unfair, as the neural network representation will always be at a disadvantage because it is only constructed using a small fraction of the data made available to



(a) True Brillouin

(b) True LEO

(c) Brillouin $N = 80$; $\mathbb{E}(x, 0, 42)$

(d) LEO $N = 80$; $\mathbb{E}(x, 0, 42)$

(e) Brillouin $N = 80$; $\mathbb{E}(x, 420, 42)$

(f) LEO $N = 80$; $\mathbb{E}(x, 420, 42)$

Figure 9: $N = 80$ traditional neural network predictions of the surface and LEO features as a function of training data distribution. Note how the surface constrained set fails to extrapolate to higher altitudes, whereas the orbit constrained set extrapolates to the surface with greater success.

the high-fidelity EGM2008 model. A more interesting comparison is how well the neural network representations compare to a spherical harmonic model that is fit with the same data.

A new experiment is conducted which fits both a neural network representation using back-propagation and a spherical harmonic representation using iterative least squares with only 9,500 randomly distributed position and acceleration measurements from the same uniform distribution in altitude between 0 and 420 km as before. Immediately the spherical harmonic representation is at a disadvantage. The spherical harmonic model is formed in a periodic basis, and its coefficients can only be regressed to high-degree successfully if there are sufficient data that are distributed properly such that the regression can observe the higher frequencies and avoid aliasing. The maximum angular separation between two points in the dataset is approximately 5.5 degrees, therefore the spherical harmonic fit can only observe a maximum degree of $l = 33$ [21].

This effect is shown by fitting increasingly high degree spherical harmonic models beginning at $l = 3$ and ending at $l = 33$ to the randomly distributed training data. These models are then compared against the $N = \{5, 10, 15, 20, 40, 80\}$ traditional and physics-informed neural networks by observing the MRSE on the $\mathcal{A}$, $\mathcal{F}$, and $\mathcal{C}$ sets. Their performance is shown in Figure 10.

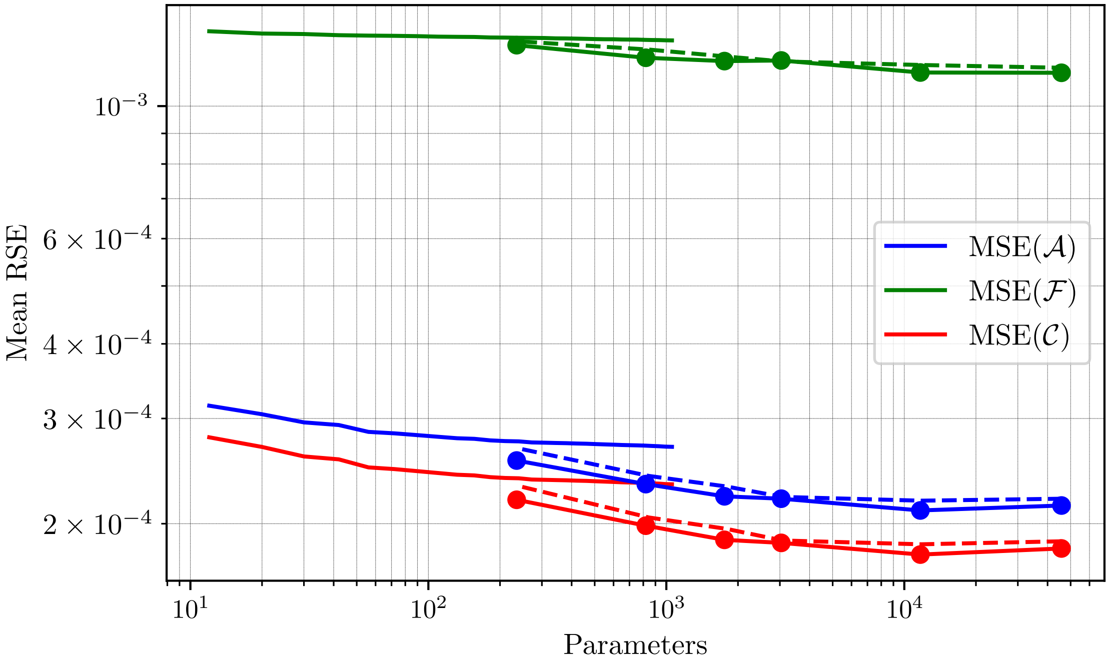


Figure 10: Plot of MRSE$_i$ as a function of total model parameters. Solid lines represent the error associated with the regressed spherical harmonic model. Dashed lines represent the performance of traditional neural networks. Lines with circles represent the physics-informed networks. All models were regressed on a datasets of 9,500 randomly distributed position and acceleration pairs.

The neural network representations outperform their analytic, regressed counterparts – offering a 20% more accurate solution in $\mathcal{A}$ for the physics informed $N = 40$ case. The spherical harmonic representation struggles to capture the same accuracy as the original EGM model for multiple reasons. One reason is that the signals from higher-degree harmonics still exist within the measurements. If those signatures are not explicitly sought out in the least-squares fit, their contributions will wrongfully bias the low-degree coefficients in the solution. The other challenge is identifying the contributions of the high-degree harmonics in high-altitude samples. The perturbing signature of high-degree harmonics decay as $\left(\frac{R}{r}\right)^l$, so identifying their contributions from a high-altitude sample is challenging. Altogether, this irregular sampling distribution and faint signatures cause the model to overfit to the training data, and generate more erroneous results than the EGM2008 solution.

The neural network representations maintain a consistently lower error on the test set in a manner that is generally independent with the number of parameters used in the representation. While there exists a slight advantage of using a physics-informed network over the traditional networks, both network representations are more accommodating to non-uniform distributions of data which could have potential ramifications for resolving the gravity field of bodies for which data is generally sparse – such as with flybys.

COMPUTATIONAL SPEED

The last analysis investigates the total evaluation speed of the neural network representations as compared to the spherical harmonics and polyhedral gravity models. Each representation is tested at varying levels of fidelity. The spherical harmonics representations are each generated from the EGM2008 dataset with different truncation degrees. The polyhedral models are evaluated on increasingly degraded shape models of 433-Eros generated using Blender [†]. The two analytic representations are written in Python, just-in-time compiled with multithreading using Numba [‡], and executed on a Ryzen 3400G CPU. The neural network representations are written in Tensorflow 2.1 and executed on an NVIDIA RTX 2060 GPU in the GPU case and on the Ryzen 3400G for the CPU case. Figure 11 shows the execution time required to evaluate the accelerations of 10,000 randomly distributed position data for models of varying levels of fidelity/parameters.

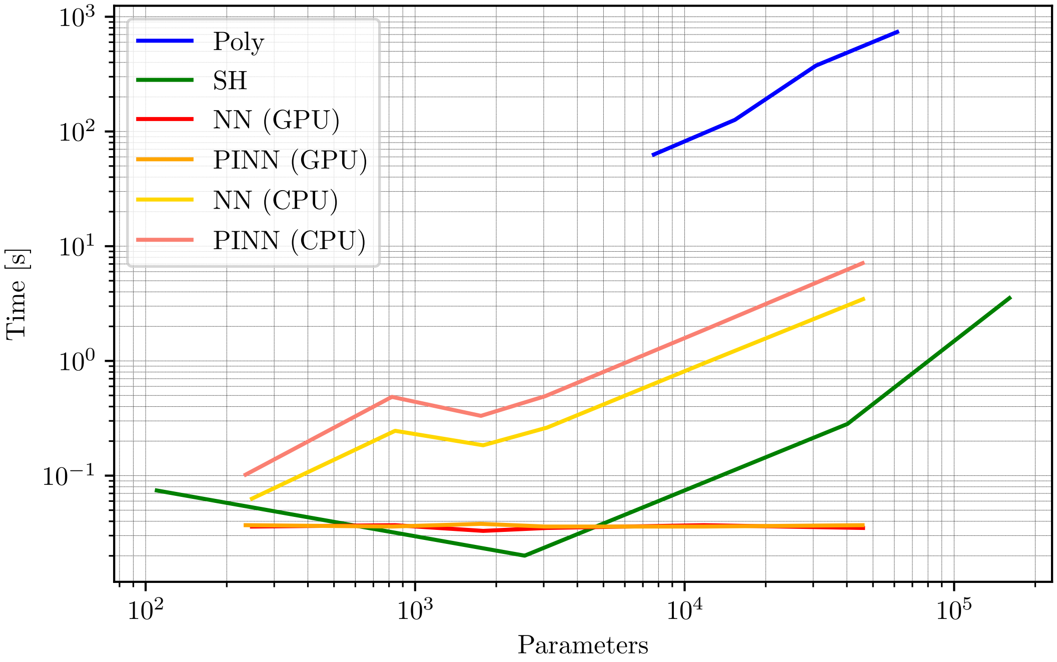


Figure 11: Total evaluation time to evaluate 10,000 random data using the various gravity models.

The polyhedral representation is by far the most time consuming to evaluate. The spherical harmonics representation is quite efficient to evaluate at low degree, but it also has the steepest gradient as the degree increases. The neural network GPU representation holds a relatively steady evaluation speed, independent to the number of parameters in the model. This is attributed to the parallelization capabilities of the GPU which allow for all nodes per layer in the network to be evaluated simultaneously. So long as the number of nodes per layer does not exceed a given hardware-specific core and memory threshold, the evaluation time should remain relatively stagnant for the architectures chosen in this paper. Furthermore this baseline time will likely decrease as GPU memory access grows faster. The neural network CPU representation grows exponentially with respect to the number of parameters in the model, similar to that of polyhedral models.

[†] https://www.blender.org/

[‡] https://numba.pydata.org/

APPLICATION: EROS

An interesting application for the traditional and physics-informed neural network gravity model is in the case of small-body exploration. Unlike with larger celestial bodies, it often undesirable to use a spherical harmonic representation of the potential for dynamics as the solutions begin to diverge within the circumscribing sphere. A polyhedral gravity model is typically used to combat this divergence. A polyhedral gravity model takes a shape model of a body comprised faces and vertices and uses it to predict dynamics down to the surface of the body using the following equation:

$$\nabla U = -G\sigma \sum_{e\in \text{ edges}} \mathbf{E}_e \cdot \mathbf{r}_e \cdot L_e + G\sigma \sum_{f\in \text{ faces}} \mathbf{F}_f \cdot \mathbf{r}_f \cdot \omega_f \tag{12}$$

where G is the gravitational constant, σ is the density of the body, $\mathbf{E}_e$ is an edge dyad, $\mathbf{r}_e$ is the position vector between the center of the edge and the field point, L_e is an analog to the potential contribution by the edge, $\mathbf{F}_f$ is the face normal dyad, $\mathbf{r}_f$ is the distance between the face normal and the field point, and ω_f is analog to the potential contribution by the face. More details can be found in Werner and Scheeres [6].

While this gravity model remains stable within the circumscribing sphere, it suffers two disadvantages. First, there is an assumption of uniform density within the body as indicated by the lack of subscript for σ in Equation 12. A unique density profile can be forced onto the model, but those profiles are challenging to resolve uniquely [22, 8]. The second disadvantage is the computational overhead of such a model as shown in Figure 11. The computation time of Equation 12 directly relates to the fidelity of the shape model and can include looping over 100,000s of edges and faces to evaluate a single field point. This become computationally expensive and poses a challenge for both offline simulation and onboard control unless properly parallelized (similar challenges plague the mascon model [23]).

A neural network gravity model may circumvent these primary disadvantages. The neural network representation does not rely on the fidelity of a shape model, nor does it make any assumptions about the underlying density distribution. To this end it may be able to acquire a more representative picture of the accelerations experienced by a spacecraft in orbit. There are also no analytic reasons that the neural network model cannot generalize within the circumscribing sphere – already yielding an advantage over the alternative spherical harmonic representation.

These theories are tested using similar analyses from the Earth specific case. The representational compactness of the polyhedral model is tested by generating four increasingly high-fidelity representations of the astroid 433-Eros using Blender. The number of unique parameters needed to construct the polyhedral model are defined by counting the number of unique vertices and multiplying by 3 to include the components of their positions. The number of faces on the model are then added to this total to account for the unique combinations of these vertices that make up the individual faces. The truth gravity field for Eros is generated using a polyhedral model based on the highest fidelity shape in the set which comprised of 245,760 total parameters. The same mean root-squared error metric is taken at the circumscribing sphere to compute the error of the lower fidelity shape models, though the density of the DH grid is reduced from 1 datum per 0.5 degree to 1 datum per degree in latitude and longitude to form sets $\mathcal{A}'$, $\mathcal{F}'$, and $\mathcal{C}'$.

The neural networks are trained using only 100,000 position and acceleration data that are randomly distributed in latitude and longitude and between the surface of the body and 10 km above the

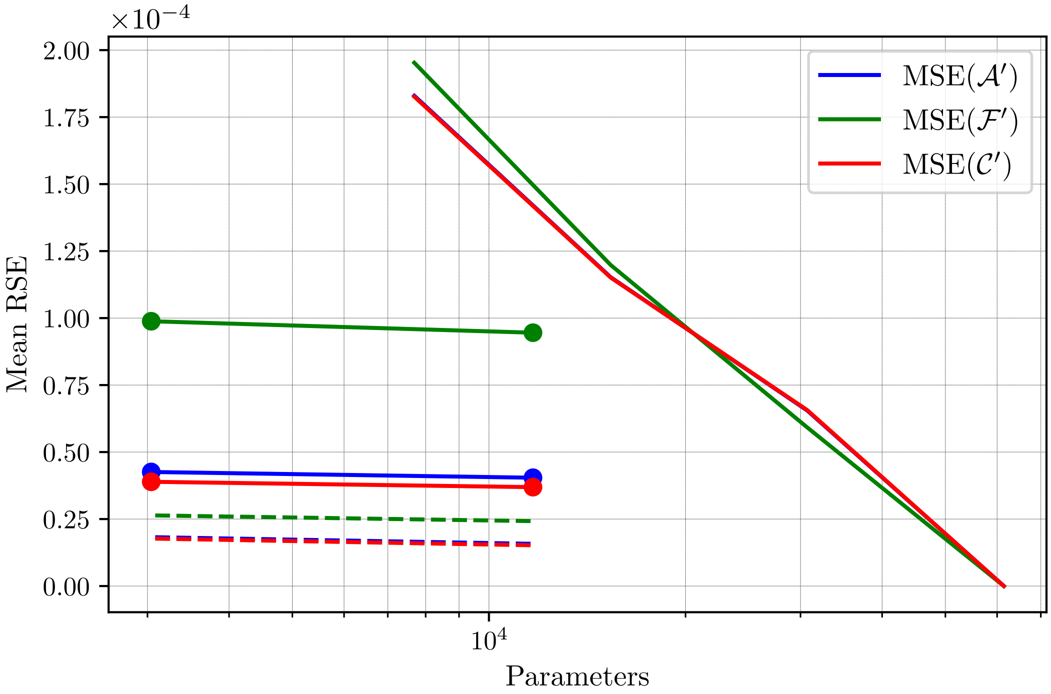
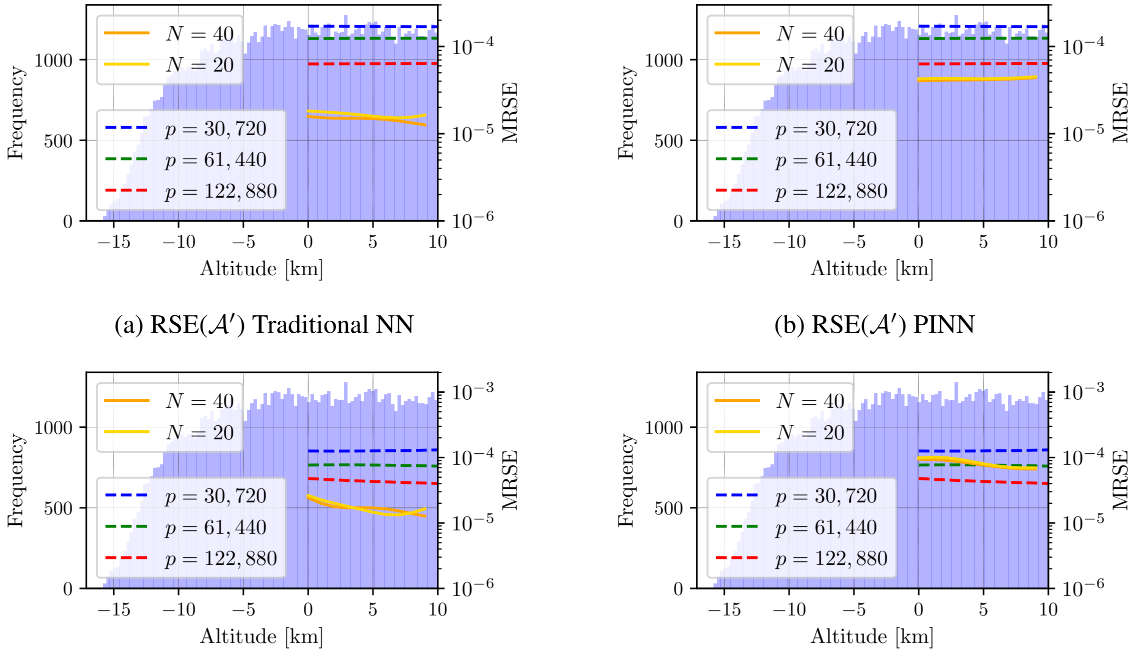


Figure 12: Plot of MRSE$_i$ as a function of total model parameters. Solid lines represent the error associated with the polyhedral model of various shape fidelities. Dashed lines represent the performance of traditional neural networks. Dotted lines represent the performance of PINNs.



(a) RSE($\mathcal{A}'$) Traditional NN

(b) RSE($\mathcal{A}'$) PINN

(c) RSE($\mathcal{F}'$) Traditional NN

(d) RSE($\mathcal{F}'$) PINN

Figure 13: Training data distribution and error curves for neural networks

circumscribing sphere. There is also no feature engineering of this data prior to training in contrast to the Earth data which removed ∇U_2 from the acceleration measurements.

Figure 12 shows the average error across $\mathcal{A}', \mathcal{F}'$, and $\mathcal{C}'$ for the lower fidelity polyhedral models and the traditional and physics-informed neural networks of $N = \{20, 40\}$. Both network types form more compact representations of the gravitational environment, however the traditional neural network achieves greater accuracy than the physics-informed network. It is assumed that this behavior is similar to that of the orbit-constrained training distribution for Earth. The physics-informed potential must accommodate infrequent and non-conformant samples near the surface which challenge the representation generated by the more regular high-altitude samples. Figure 13 shows how the training samples grow less frequent within the circumscribing sphere (negative altitude), but above the asteroid surface. Future work remains to investigate if this behavior can be combatted, and how the networks perform within the Brillouin sphere.

FUTURE WORK AND CONCLUSIONS

Combining new machine learning techniques and artificial neural networks offer an interesting alternative to analytic gravity models. In many cases, the machine learning representation of the gravitational environment offers strong advantages to modeling highly perturbing features within the environment without the need for excessive parameters. The machine learning representation is also more efficient at regressing the gravitational field of a given body as compared to the spherical harmonic representation which has more stringent requirements about the distribution of data used to constrain the fit. Finally, the neural network representation also has the advantage of fast execution times when run on a GPU which could make it a viable alternative particularly to the computationally cumbersome polyhedral model.

Altogether the neural network gravity model is a powerful way to represent the gravitational environment and offers a number of encouraging prospects for future research. In particular, all networks in this paper involved minimal user-led optimization. No hyperparameter searches are conducted, no alternative network architectures are investigated, and no efforts to impose additional physics-informed constraints are explored. Despite this, the performance of these networks is already comparable with state-of-the-art spherical harmonic models. With further feature engineering and tuning, it is likely that these neural network representations may yield even more promising results in the future.

Other work remains such as analyzing the performance of these networks within the circumscribing sphere. It is possible that such a representation may yield stable dynamics in a regime that is otherwise dynamically challenging to model. Other avenues of research include applying these techniques to real data and investigating the effect of uncertainty in measurements on the performance of the networks. It is possible that the efficient regression capabilities of these networks could be used to generate higher-fidelity estimates of the gravity field of minimally explored celestial bodies. In any case, these networks demonstrate promise for many future gravity modeling efforts.

ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. 2040434.

REFERENCES

[1] M. Brillouin, "Équations aux dérivées partielles du 2e ordre. Domaines à connexion multiple. Fonctions sphériques non antipodes," Vol. 4, 1933, pp. 173–206.

[2] B. D. Tapley, "Gravity model determination from the GRACE mission," *Journal of the Astronautical Sciences*, Vol. 56, No. 3, 2008, pp. 273–285, 10.1007/bf03256553.

[3] F. G. Lemoine, S. Goossens, T. J. Sabaka, J. B. Nicholas, E. Mazarico, D. D. Rowlands, B. D. Loomis, D. S. Chinn, G. A. Neumann, D. E. Smith, and M. T. Zuber, "GRGM900C: A degree 900 lunar gravity model from GRAIL primary and extended mission data," *Geophysical Research Letters*, Vol. 41, No. 10, 2014, pp. 3382–3389, 10.1002/2014GL060027.

[4] N. K. Pavlis, S. A. Holmes, S. C. Kenyon, D. Schmidt, and R. Trimmer, "A preliminary gravitational model to degree 2160," *International Association of Geodesy Symposia*, Vol. 129, 2005, pp. 18–23, 10.1007/3-540-26932-0-4.

[5] G. Romain and B. Jean-Pierre, "Ellipsoidal harmonic expansions of the gravitational potential: Theory and application," *Celestial Mechanics and Dynamical Astronomy*, Vol. 79, No. 4, 2001, pp. 235–275, 10.1023/A:1017555515763.

[6] R. A. Werner and D. J. Scheeres, "Exterior gravitation of a polyhedron derived and compared with harmonic and mascon gravitation representations of asteroid 4769 Castalia," *Celestial Mechanics and Dynamical Astronomy*, Vol. 65, No. 3, 1996, pp. 313–344, 10.1007/bf00053511.

[7] S. Tardivel, "The limits of the mascons approximation of the homogeneous polyhedron," *AIAA/AAS Astrodynamics Specialist Conference*, 2016, p. 5261.

[8] Y. Takahashi and D. J. Scheeres, "Morphology driven density distribution estimation for small bodies," *Icarus*, Vol. 233, 2014, pp. 179–193, 10.1016/j.icarus.2014.02.004.

[9] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, Vol. 521, No. 7553, 2015, pp. 436–444, 10.1038/nature14539.

[10] L. Cheng, Z. Wang, Y. Song, and F. Jiang, "Real-time optimal control for irregular asteroid landings using deep neural networks," *Acta Astronautica*, Vol. 170, No. January 2019, 2020, pp. 66–79, 10.1016/j.actaastro.2019.11.039.

[11] A. Gao and W. Liao, "Efficient gravity field modeling method for small bodies based on Gaussian process regression," *Acta Astronautica*, Vol. 157, No. December 2018, 2019, pp. 73–91, 10.1016/j.actaastro.2018.12.020.

[12] R. Furfaro, R. Barocco, R. Linares, F. Topputo, V. Reddy, J. Simo, and L. Le Corre, "Modeling irregular small bodies gravity field via extreme learning machines and Bayesian optimization," *Advances in Space Research*, Vol. 67, No. 1, 2021, pp. 617–638, 10.1016/j.asr.2020.06.021.

[13] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, Vol. 378, 2019, pp. 686–707, 10.1016/j.jcp.2018.10.045.

[14] J. R. Driscoll and D. M. Healy, "Computing fourier transforms and convolutions on the 2-sphere," 1994, 10.1006/aama.1994.1008.

[15] S. Pines, "Uniform Representation of the Gravitational Potential and its Derivatives," *AIAA Journal*, Vol. 11, No. 11, 1973, pp. 1508–1511.

[16] E. Hewitt and R. E. Hewitt, "The Gibbs-Wilbraham phenomenon: An episode in fourier analysis," *Archive for History of Exact Sciences*, Vol. 21, No. 2, 1979, pp. 129–160, 10.1007/BF00330404.

[17] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015, pp. 1–15.

[18] L. Bottou, "Stochastic Gradient Descent Tricks," Vol. 1, No. 1, 2012, pp. 421–436, 10.1007/978-3-642-35289-8-25.

[19] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *Journal of Machine Learning Research*, Vol. 9, 2010, pp. 249–256.

[20] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, Vol. 2, No. 5, 1989, pp. 359–366, 10.1016/0893-6080(89)90020-8.

[21] C. Jekeli, "Potential Theory and the Static Gravity Field of the Earth," *Treatise on Geophysics: Second Edition*, Vol. 3, 2015, pp. 9–35, 10.1016/B978-0-444-53802-4.00056-7.

[22] R. S. Park, R. A. Werner, and S. Bhaskaran, "Estimating small-body gravity field from shape model and navigation data," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 1, 2010, pp. 212–221, 10.2514/1.41585.

[23] R. P. Russell and N. Arora, "Global point Mascon models for simple, accurate, and parallel geopotential computation," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 5, 2012, pp. 1568–1581, 10.2514/1.54533.