# DIPOLE FITTING IN UNIT-LEVEL SPACECRAFT EQUIPMENT WITH DEEP NEURAL NETWORKS

**Anastasios E. Giannopoulos [(1)], Sotirios T. Spantideas [(1)], Christos D. Nikolopoulos [(1),(2)], Anargyros T. Baklezos [(1),(2)] and Christos N. Capsalis [(1)]**

*[(1)] School of Electrical and Computer Engineering, National Technical University of Athens, 9 Iroon Polytechniou St., 15780, Greece. Email of corresponding author: sospanti@gmail.com*
*[(2)] Department of Electronic Engineering, Hellenic Mediterranean University, Romanou 3 Chalepa, 73133, Crete, Greece.*

## ABSTRACT

**Magnetic Dipole Modeling (MDM) is a well-established method for fitting the spatial magnetic signature generated by magnetic sources into a model comprised of magnetic dipoles. In this paper, we study the dipole fitting problem using Machine Learning (ML) techniques. Simulated data were used to estimate the dipole parameters that can accurately reconstruct the measured magnetic field. We applied our methodology in two widely used measurement facilities, namely the Magnetic Coil Facility (MCF) and the Multi Magnetometer Facility (MMF). To solve this regression problem, Artificial Neural Networks (ANNs) were considered. Simulation results showed that ANNs significantly outperform other benchmark ML methods in terms of the achieved accuracy of the estimated dipole parameters. Finally, we observed that MMF model shows enhanced field reconstruction accuracy with reduced position estimation loss (~70% lower position error), magnetic moment estimation loss (~55% lower moment error) and model complexity (33.3% lower complexity) compared to the MCF model.**

## 1. INTRODUCTION

Well-established measurement methodologies and characterization procedures for static magnetic cleanliness programmes in space missions primarily involve unit-level measurements during the design of the spacecraft. Post-processing of the measured data is also required to derive an accurate magnetic model that can be interchangeably used to represent the original piece of equipment [1]. In this framework, several spacecraft units that may be significant magnetic field generators are subjected to test measurements with detailed specifications, depending on the individual magnetic cleanliness requirements of each space mission.

These tests are traditionally performed in the Magnetic Coil Facility (MCF) or the Multi-Magnetometer Facility (MMF); in the former the unit rotates and a fixed tri-axial magnetic field sensor is acquiring the spatial signature of the unit every 10 degrees (Fig. 1), while in the latter, multiple sensors simultaneously measure the signature of the unit from various orientations, obtaining a snapshot of the magnetic field from various spatial orientations (Fig. 2) [2].

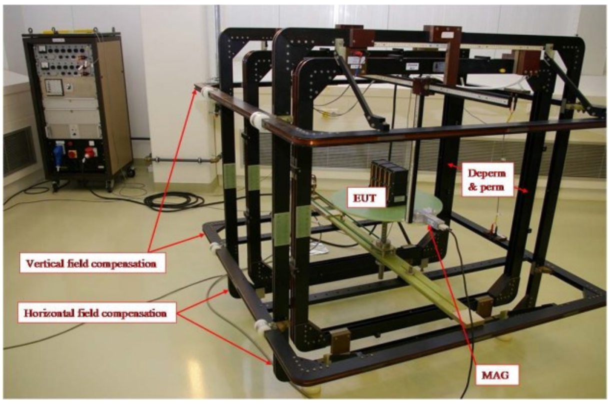


*Figure 1. Magnetic Coil Facility (MCF). The magnetic field measurements are obtained while the unit completes a full rotation.*

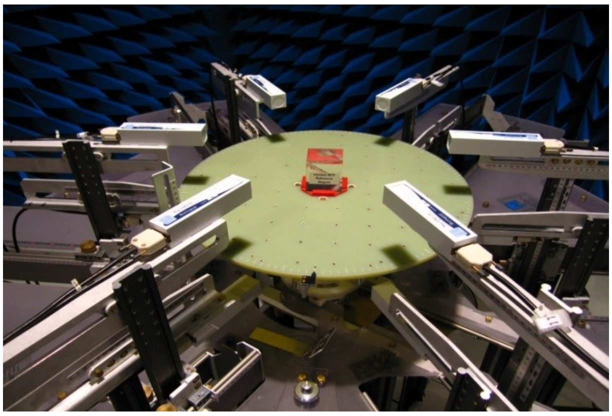

*Figure 2. Multi Magnetometer Facility (MMF). Multiple sensors simultaneously capture the spatial magnetic signature of the unit from different orientations.*

The measured spatial variations of the magnetic field are then used for dipole fitting purposes, usually following the well-established Multiple Dipole Modeling (MDM) method [1], [3]. The target is to derive a model composing of one or more magnetic dipoles to represent the field created from the original unit, enabling system level simulations and/or identifying possible design defects due to non-uniform distribution of the magnetic material. The dipole(s) parameters determined by the MDM algorithm include the 3D position vectors of the sources inside the unit and their respective 3D magnetic moment vectors, leading to 6 variables per dipole source. The MDM algorithm can be applied by using either deterministic or stochastic (genetic algorithm or particle swarm optimization based stochastic solver) optimization techniques [1], [3].

These solvers, currently implemented both on the MCF and MMF facilities, follow an iterative way to fine-tune the dipole(s) parameters in order to minimize the deviation between the measured field and the field generated by the model. However, they show some key limitations: (i) although the algorithms are optimized to provide fast estimations (for instance in MCF and MMF the algorithm runtime is upper-bounded to 1s to determine the model parameters), they fail to provide reliable models in terms of estimated position and magnetic moment [4], especially when more complicated spatial magnetic field variations are encountered (i.e. the unit consists of multiple dipoles). To this end, these methods need significant runtime to provide accurate models and, beyond that, they can be stuck in suboptimal solutions [5]; (ii) the facility operator has to manually register the values of the position of the magnetometers; the algorithms use these values as input in order to solve the inverse problem and provide the model parameters, whose accuracy naturally depends on the accuracy of the position of the magnetic field sensors [2].

Machine Learning (ML) algorithms can be used to solve regression problems given a complex mapping between input and output vectors [6]. ML targets to find a mapping between inputs (or features) and outputs (or dependent variables) in order to minimize the difference between the predicted and groundtruth values. When the desired groundtruth values are known in advance, we refer to a specific branch of ML, namely the Supervised Learning (SL). SL relies on historically collected data (both input and desired outputs) for training an ML model to accurately predict the output values of input samples not encountered during the training (validation samples). Deep Network Networks (DNNs) comprise a powerful toolset to estimate extremely complex and non-linear functions by stacking multiple layers of fundamental units (perceptrons or neurons) for purposes of resolving the hidden correlation patterns between the input vectors [7].

Based on the above considerations, this paper uses DNNs to solve the MDM inverse problem. Following the principles of Deep Learning (DL), we train a DNN to "learn" the magnetic field equation, i.e. to "learn" the inverse mapping between dipole parameters (output variables) and the generated magnetic field at an observation distance (input variables). Extending the previous work on DL-based MDM approach, here we test the performance of DNNs in estimating the dipole parameters derived by two different facilities (MCF and MMF). In further extending the previous work [4], we introduce varying observation distance in the model input to obtain a generalized distance-independent solution. Additionally, we aim to investigate whether (i) DNNs generally outperform other baseline ML schemes and (ii) compare the model performance against two widely used facilities.

In summary, the benefits of using DNNs to solve the MDM problem include:

- Training and testing of the two neural networks can be performed offline with simulated data from virtual magnetic sources.
- Once the neural networks are trained, the inference (output estimation, i.e. model prediction) can be performed swiftly, while possible model updates can be effortlessly conducted through soft DNN retraining.
- The trained model exhibits the property of generalizability, in the sense that the training is performed for various positions of the magnetic field sensors, targeting to make the neural network independent of the specific measurement setup.
- This approach can be further extended for increasing number of virtual dipole sources, i.e. the training data consists of magnetic field generated by multiple dipoles in order to overcome the limited accuracy of the stochastic solutions.

## 2. BACKGROUND AND METHODOLOGY

In principle, given the parameters of $M$ equivalent magnetic dipoles representing the magnetic signature produced by a unit, the dipole-specific field can be calculated via:

$$\mathbf{B}_{ij} = \frac{\mu_0}{4\pi}\left[\frac{3(\mathbf{r}_i - \mathbf{r}'_j)[(\mathbf{r}_i - \mathbf{r}'_j)\cdot \mathbf{m}_j]}{|\mathbf{r}_i - \mathbf{r}'_j|^5} - \frac{\mathbf{m}_j}{|\mathbf{r}_i - \mathbf{r}'_j|^3}\right], \quad (1)$$

where $\mathbf{B}_{ij}$ denotes the magnetic field vector generated by the $j^{th}$ dipole ($j = 1, 2, \dots M$) and captured at the $i^{th}$ measurement point ($i = 1, 2, \dots N$). In addition, $\mathbf{r}_i - \mathbf{r}'_j$ denotes the relative 3D position between the $i^{th}$ measurement point and the $j^{th}$ dipole source, $\mathbf{m}_j$ stands for the magnetic moment of the $j^{th}$ dipole and $\mu_0$ is the permeability of free space. The total magnetic field at a measurement point can be calculated as the accumulated contribution of $M$ dipoles.

Eq. (1) defines a forward problem in the sense that the magnetic field captured at the measurement points is calculated based on the dipole parameters (position and magnetic moment). On the contrary, the inverse problem refers to the exploitation of the magnetic field measurements to quantify the dipole parameters. This problem has been proven ill-posed and, theoretically, requires massive measurement points to uniquely describe the parameters of the dipole sources.

In this paper, we address the inverse problem by forming the MDM as a regression problem, considering the dipole parameters as predictors of the produced magnetic field (dependent variable). To obtain the training dataset, we use Eq. (1) to generate 10000 input/output pairs relating the magnetic field at all observation points (input) to the dipole parameters (output). Two separate neural networks (MCFnet and MMFnet) are trained on the generated data, according to the specifications of the measurement procedures of these two facilities. Thus, two different training datasets were generated separately for each facility and measurement procedure. In each case, 9000 data samples are used for the DNNs training phase consisting of the dipole parameters (as features) and the associated magnetic field at an observation distance (as labels). DNN training is in charge of efficiently solving this non-linear regression problem, ensuring high performance not only in the training samples, but also in unseen validation data (i.e. avoidance of overfitting). The remaining 1000 samples were used to evaluate the performance of the DNNs on data not encountered during the training.

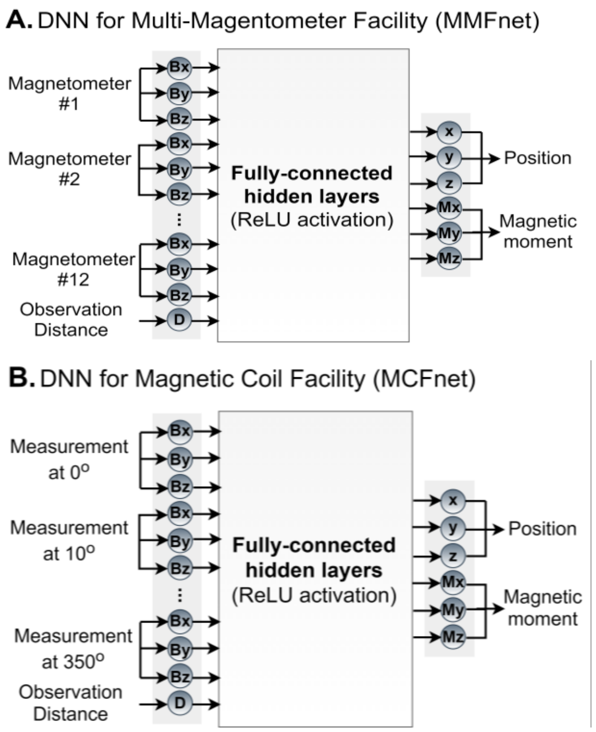


*Figure 3. Dimensionality of the MMFnet (panel A) and MCFnet (panel B). DNN inputs involve the magnetic field measurements and the observation distance defined by each facility specifications, whereas DNN outputs are the dipole parameters.*

The dimensionality of the neural networks for both facilities are shown in Fig. 3, taking into account that the MCF and MMF obtain 36 (0-360° in steps of 10°) and 12 (6 equally-spaced sensors below and above the turntable) tri-axial magnetic field measurements, respectively. Without loss of generality, the single dipole fitting approach was used to train both DNNs. The virtual dipoles in each of the 9000 training samples are assumed to be located on the turntable of each facility with multiple values of 3D position and varying magnetic moment vectors, randomly drawn from a uniform distribution. Moreover, the radial observation distance $D$ of the magnetic field sensors is included in the input (feature) set, allowing to train the models with varying values of distance, thus obtaining distance-independent model predictions.

It should be noted that, during the training process, DNNs target to properly adjust their weights (interconnections between layers), so as to minimize a predefined loss function. In this study, we use the Mean Squared Error (MSE) as the loss function for both MMFnet and MCFnet, given by:

$$Loss_{MSE} = \frac{1}{N_s}\sum_{i=1}^{N_s}\left(y_{actual}(i) - y_{predicted}(i)\right)^2, \qquad (2)$$

where $N_s$ is the number of samples, $y_{actual}(i)$ and $y_{predicted}(i)$ are the groundtruth and the DNN-derived output values of the $i^{th}$ sample.

Following the architecture of the conventional multi-layer perceptrons, the number of neurons between two consecutive layers is reduced by a factor of 2, whereas the final hidden layer has at least 2x6 neurons (6 is the number of neurons in the output layer). Finally, the Rectified Linear (ReLu) activation function was selected as a typical setting in the regression problems.

## 3. SIMULATION RESULTS

### 3.1. MMFnet and MCFnet training

The DNN training process mainly involves the fine-tuning (or stabilization) of the major learning parameters, including the number of hidden layers, the number of neurons and the learning rate ($a$) in order to guarantee stable-and-high performance. This means that before using a DNN for making predictions, we have to determine the optimal hyper-parameter configuration that ensures high predictive accuracy in the validation data. To this end, we conducted extensive simulations with different combinations of learning rate values ($a = 0.1, 0.01, 0.001, 0.0001, 0.00001$) and number of hidden layers (1-7). Fig. 4 depicts the validation MSE loss (i.e. the deviation metric between the actual dipole parameters and DNN-estimated parameters across the 1000 validation samples) for the MMFnet and MCFnet.

To effectively monitor the training process and convergence of both DNNs, we additionally applied the following stopping criterion: the training process terminates if at least 0.001 decrement in the loss function was not observed for 20 consecutive epochs. This stopping criterion prevents time-consuming training process, allowing to identify the period of convergence.

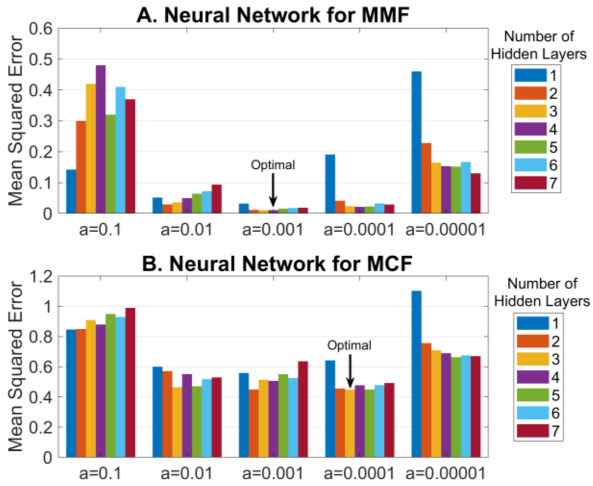


*Figure 4. Validation loss (MSE) relative to different values of learning rate, number of hidden layers and number of neurons for the MMFnet (panel A) and the MCFnet (panel B).*

As observed from Fig. 4, the optimal hyper-parameters (corresponding to the minimum MSE validation loss) for the MMFnet are obtained for $a = 0.001$ and 4 hidden layers, whereas the optimal hyper-parameters for MCFnet are $a = 0.0001$ and 3 hidden layers. Notably, the MMFnet converged to lower MSE value (by approximately one order of magnitude) compared with the MCFnet.

### 3.2. Comparison with ML baseline regressors

In this subsection, we compare the optimally configured DNNs (see section 3.1.) with 6 typical ML regressors. Specifically, the following benchmark algorithms were implemented: (i) Linear Regression (LR); (ii) Lasso Regression; (iii) ElasticNet Regression; (iv) Ridge Regression; (vi) Decision Tree (DT) Regression and (vii) Random Forest (RF) Regression. All regressors were trained on the same input and output samples and had the same objective as the DNNs.

Fig. 5 demonstrates the validation performance of all regressors in a descending order in terms of MSE. In both MCF and MMF scenarios, it is evident that DNNs outperform the other ML baselines. Moreover, the linear regressors (Lasso, Elastic, LR and Ridge) fail to effectively estimate the dipole model parameters, converging in higher MSE values, whereas the non-linear regressors (DT, RF and ANN) show reduced validation loss values. It is also worth mentioning that RF outperforms the DT regressor, since it inherently uses multiple decision trees (ensemble learning) for prediction-making.

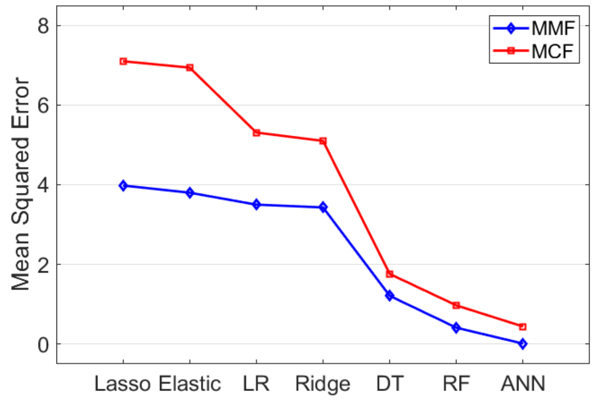


*Figure 5. Comparison between the seven ML regressors in terms of validation loss (MSE).*

### 3.3. MCFnet versus MMFnet predictions

Having fine-tuned the DNNs hyper-parameters for the minimal MSE validation loss, the performance of the MCFnet and the MMFnet is also assessed in terms of resulting position and magnetic moment deviation in order to quantify the predictions of the trained models. Based on 1000 instances of DNN-predicted model parameters, we test the model performance (i.e. goodness of dipole fitting) against the actual parameters provided by the validation samples. Fig. 6 demonstrates the average position deviation (root mean squared – RMS – deviation between the predicted and the actual dipole position in cm) and the average magnetic moment deviation (RMS deviation between the predicted and the actual magnetic moment in $mAm^2$).

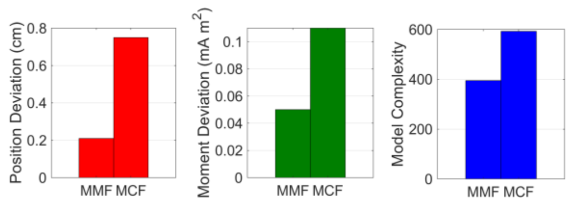


*Figure 6. RMS position deviation (left panel), moment deviation (central panel) and model complexity (right panel) for MCFnet and MMFnet.*

Evidently, the predictions of the MMFnet exhibit a position deviation of ~0.2 cm and a moment deviation of ~0.05 $mAm^2$, whereas the MCFnet estimation exhibit

~0.75 cm and ~0.12 $mAm^2$ deviations respectively. The DNN model complexity was quantified taking into account: (i) the neural network density ($D$, total number of neurons divided by the total number of hidden layers) and (ii) the number of training epochs required for convergence ($T$). Therefore, the model complexity is defined by the sum $D + T$. This parameter is shown in the third panel of Fig. 6, where the MCFnet displays 33.3% more complexity than the MMFnet. This might be attributed to the more complete geometrical coverage of the magnetic signature offered by MMF (as compared to the MCF), capturing the generated field from two planes and several orientations.

Finally, we extracted the magnetic field that is reconstructed by using the DNN-predicted dipole parameters. Fig. 7 shows the actual and the reconstructed magnitude of the magnetic field at each measurement point both for the MMFnet and MCFnet models.

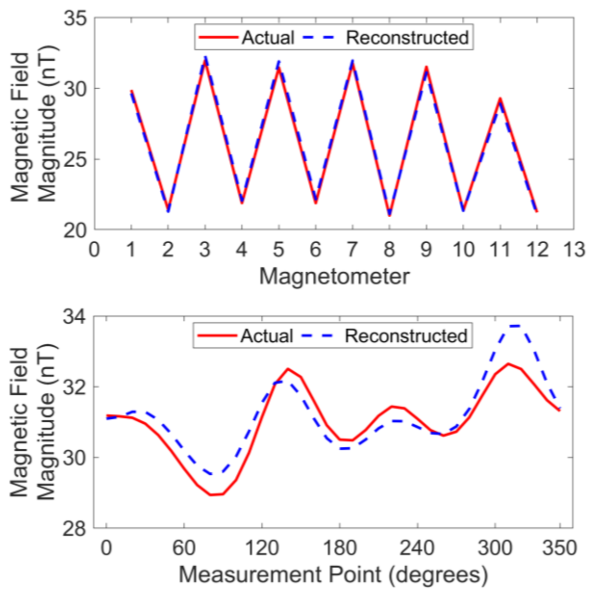


*Figure 7. Magnetic field (magnitude) reconstruction accuracy for the MMFnet (upper panel) and MCFnet (lower panel) at the measurement points.*

The resulting field reconstruction values deviate from the actual field magnitude by at most ~0.1% for the MMFnet, and at most ~3% for the MCFnet.

## 4. CONCLUSION

This paper applies ML methods for purposes of dipole fitting, based on magnetic field measurements obtained by two facilities. To this end, two DNNs are trained and tested on two separate datasets, properly adapted to the MCF and MMF setups. The trained DNNs are also compared to other baseline ML schemes, showing beneficial performance in terms of their MSE convergence. We also provide evidence regarding the performance of both MMF and MCF neural networks, quantifying their accuracy in the position and magnetic moment estimation, as well as their model complexity. We conclude that MMFnet can effectively estimate the dipole parameters with lower complexity compared with the MCFnet.

## 5. REFERENCES

1. ESA. (2012). Space Engineering: Electromagnetic Compatibility Handbook. *ECSS-E-HB-20-07A, Eur. Space Agency, ESA-ESTEC*, Noordwijk, The Netherlands. http://everyspec.com/ESA/ECSS-E-HB-20-07A_47799/

2. Tsatalas, S., Vergos, D., Spantideas, S. T., Kapsalis, N. C., Kakarakis, S. D., Livanos, N. A., ... & Junge, A. (2019). A novel multi-magnetometer facility for on-ground characterization of spacecraft equipment. *Measurement*, *146*, 948-960.

3. Junge, A., & Marliani, F. (2011, August). Prediction of DC magnetic fields for magnetic cleanliness on spacecraft. In *2011 IEEE International Symposium on Electromagnetic Compatibility* (pp. 834-839). IEEE.

4. Spantideas, S. T., Giannopoulos, A. E., Kapsalis, N. C., & Capsalis, C. N. (2021). A Deep Learning Method for Modeling the Magnetic Signature of Spacecraft Equipment Using Multiple Magnetic Dipoles. *IEEE Magnetics Letters*, *12*, 1-5.

5. Spantideas, S. T., Kapsalis, N. C., Kakarakis, S. D., & Capsalis, C. N. (2016, May). A novel technique for accurate extrapolation of complex magnetic sources. In *2016 ESA Workshop on Aerospace EMC (Aerospace EMC)* (pp. 1-5). IEEE.

6. Kaloxylos, A., Gavras, A., Camps Mur, D., Ghoraishi, M., & Hrasnica, H. (2020). AI and ML—Enablers for beyond 5G networks. *Zenodo, Honolulu, HI, USA, Tech. Rep*.

7. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.