Computers & Geosciences 208 (2026) 106092

Contents lists available at ScienceDirect

Computers and Geosciences

journal homepage: www.elsevier.com/locate/cageo

Research paper

Optimising the computational performance of high degree lithospheric field models

Michael Bareford [a,*], William Brown [b], Ciarán Beggan [b], Callum Watson [b], Mark Bull [a]

[a] EPCC, Bayes Centre, University of Edinburgh, UK
[b] British Geological Survey, Lyell Centre, Edinburgh, UK

ARTICLE INFO

Keywords:
Lithospheric magnetic field
Spherical harmonic model
Computational efficiency
GPU offloading

ABSTRACT

The British Geological Survey (BGS) World Magnetic Anomaly Model (WMAM) code calculates spherical harmonic models of the natural magnetisation of the rocks of Earth's crust. These models allow us to estimate the value of the full magnetic field vector at any location, based on scattered pointwise marine or aero-magnetic measurements of only the scalar magnetic field. Modelling the magnetic field in this way serves many important purposes, such as geological research, navigation and safe resource extraction.

Global spherical harmonic models of degree and order 1440 (~28 km spatial resolution) have been successfully computed on the HPC facilities local to BGS, but such runs require nearly the full compute capacity for multiple days. Further, the available resolution of the scalar field measurements is too high to be fully exploited by the WMAM code, limiting models of the crustal magnetic field to a resolution of 28 km.

To overcome these issues, we refactored the WMAM code such that models of spherical harmonic degree 1440 and 2000 (~20 km resolution) can be produced in hours rather than days. For example, a degree 2000 model was calculated using 64 HPE Cray EX nodes (8192 cores) in 3 h and 44 mins. The resulting model power spectra and magnetic field maps showed excellent agreement with the existing degree 1440 model and the original input data. The performance of the WMAM code was further improved via offloading to GPU. We show the improvements due to GPU acceleration in terms of energy consumption as well as runtime.

This fruitful collaboration between experts in the fields of Geoscience (BGS) and HPC (EPCC) has created the opportunity for the WMAM code to be used to gain new knowledge about crustal magnetic fields.

1. Introduction

The Earth's magnetic field consists of three primary sources: the core (or main) field, the lithospheric (or crustal) field, and the ionospheric and magnetospheric (or external) fields. Any measurement made on or above the surface consists of contributions from these sources as well as a myriad of other weaker signals such as tidal or induced fields (Olsen and Stolle, 2012). The magnetic field varies in time and space, making it a complex system to accurately model as it requires extensive and repeated global measurements to capture the long and short term temporal variations as well as the spatial form.

The main field, representing around 95% of the total magnetic field strength, is driven by the flow of molten iron in the outer core. It changes on timescales ranging from years to billions of years, and varies over distances of thousands of km. The external field is driven by interaction between the Earth's atmosphere and magnetic field on the one hand, and solar irradiance and the solar wind on the other. It varies on timescales ranging from a fraction of a second to decades, and can reach 5%–10% of the total magnetic field strength during extreme solar storms. The lithospheric field remains effectively static in time, varying over millions of years, though it can locally vary more quickly in volcanic regions. The lithospheric field is relatively weak on average, it represents 0.1%–1% of the total field strength at the Earth's surface, but can rise to 10% depending on location.

Spherical harmonic magnetic field inverse modelling has long focussed primarily on the large-scale core field (Gauss, 1839; Glassmeier and Tsurutani, 2014). The International Geomagnetic Reference Field (IGRF) is a long-running representation of the core field up to degree and order 13 (Alken et al., 2021). The IGRF series of models have been used for navigation, mineral exploration and scientific research standards since the 1960s, as they require a relatively small number of data and computing power to create. In theory, as there are $L(L+2)$ parameters for a given point in time, where L is the spherical harmonic (SH) degree, only 195 data points are needed for $L = 13$. In practice, hundreds of thousands of data points are used to capture and separate

* Corresponding author.
E-mail address: m.bareford@epcc.ed.ac.uk (M. Bareford).

https://doi.org/10.1016/j.cageo.2025.106092
Received 26 June 2025; Received in revised form 4 November 2025; Accepted 9 December 2025
Available online 10 December 2025

out the core from the other sources in a joint inversion for all three primary sources and to capture time variations (e.g. Finlay et al., 2020).

Spherical harmonic inverse modelling of the crustal field presents a greater challenge: an $\mathcal{O}(L^2)$ increase in computing resources is needed to accommodate the larger volume of model parameters to represent smaller spatial scales. However, despite its relatively low strength, the static nature of the lithospheric field provides an advantage in that repeated measurements over the same area can provide an improved signal-to-noise ratio by averaging out the time-varying external field component (e.g. Thébault et al., 2021).

The first magnetic anomaly grids of the crustal field were created as regional maps in the 1950s and 1960s using aeromagnetic surveys (Beamish and White, 2011, and references therein). The external noise was controlled using ground-based stations for reference and removing the main field with IGRF or similar models. The first global lithospheric field maps to SH degree and order 30 were developed from the satellite data collected by the NASA MAGSAT mission, which operated in low-Earth orbit during 1979–80 (Langel and Hinze, 1998). As new polar orbiting satellite missions launched from the early 2000s onwards, improvements to degree and order 133 became possible (Maus et al., 2008), and later degree 180 (Olsen et al., 2017). In parallel, there was a global effort to amalgamate decades worth of measurements of the lithospheric field from marine, aero-magnetic, and regional ground surveys (Maus et al., 2009).

The release of version 2 of the World Digital Magnetic Anomaly Map (WDMAM) in 2016 provided a regular approximately 5 km spatial grid of values (or 0.05 ° in latitude and longitude) (Lesur et al., 2016). The most recent version, 2.2, was updated in early 2025. However, the WDMAM grid is composed of magnetic anomaly scalar values only at a fixed altitude. In order to derive the vector direction from these maps, they must be converted to a spherical harmonic representation. This allows the field to be evaluated at any location and altitude.

The inverse modelling problem to fit a spherical harmonic model to scalar magnetic field observations stems from the following basic equations. The internal magnetic potential V^{int} can be represented as the sum of two scalar potential fields; the time varying main field V^{core} and static crustal field V^{crust}, expanded in spherical harmonics in spherical polar coordinates as

$$\begin{aligned} V^{\text{int}}(\theta,\phi,r,t) &= V^{\text{core}}(\theta,\phi,r,t) + V^{\text{crust}}(\theta,\phi,r) \\ &= a\sum_{l=1}^{15}\sum_{m=-l}^{+l} g_l^m(t)\left(\frac{a}{r}\right)^{l+1} Y_l^m(\theta,\phi) \\ &\quad + a\sum_{l=16}^{L}\sum_{m=-l}^{+l} g_l^m\left(\frac{a}{r}\right)^{l+1} Y_l^m(\theta,\phi), \end{aligned} \tag{1}$$

with position in colatitude θ, longitude ϕ, radius r and time t. The Gauss coefficients of spherical harmonic degree l and order m (spatial basis wavelengths, where larger integer values of l or m indicates smaller spatial features, to a maximum degree L), g_l^m, are our desired model parameters at a reference radius a which weight the spherical harmonic basis functions of a given degree and order Y_l^m. The magnetic field vector is the gradient of the scalar potential, $\vec{B} = -\nabla V$, with the vector components in the spherical coordinate directions given by $B_r = -\frac{\partial V}{\partial r}$, $B_\theta = -\frac{1}{r}\frac{\partial V}{\partial \theta}$ and $B_\phi = -\frac{1}{r\sin\theta}\frac{\partial V}{\partial \phi}$. This gives the sought Gauss coefficients of the scalar potential a linear relationship to observations of the field vector. However, for high resolution anomaly mapping we have only scalar observations $F = \sqrt{B_r^2 + B_\theta^2 + B_\phi^2}$, and thus must solve a non-linear problem relating Gauss coefficients representing the scalar potential of the crustal anomaly field, with a known main field model, to these observations. The relationship between magnetic scalar and vector of the main (B^{core}) and crustal anomaly (δB^{crust}) fields can be linearised (see e.g. Langel, 1987) as

$$\delta F^{\text{crust}} \approx \frac{B_r^{\text{core}}}{F^{\text{core}}}\delta B_r^{\text{crust}} + \frac{B_\theta^{\text{core}}}{F^{\text{core}}}\delta B_\theta^{\text{crust}} + \frac{B_\phi^{\text{core}}}{F^{\text{core}}}\delta B_\phi^{\text{crust}}. \tag{2}$$

The normal equations of our inverse problem are constructed from the partial derivatives of the above equations with respect to the model parameters.

With such a high global resolution grid as the WDMAM, we could compute a degree and order 8000 model, but in practice this would be unwieldy and time-consuming for computing spot values. For this reason, lower-degree models are preferred. Hence, a degree-2000 model offers an attractive trade-off between computational resources and usability.

The existing BGS global (or world) lithospheric magnetic field anomaly inverse modelling code, hereafter referred to as WMAM, is used to produce high spatial resolution spherical harmonic models of the natural magnetisation of the Earth's lithosphere. In essence, such models allow for a physically meaningful manner to interpolate the various scattered measurements of the scalar magnetic field. By assuming the magnetisation typically lies along the direction of the main field, the full magnetic field vector of the Earth's crust can also be computed. The resulting derived maps of the magnetic field have a wide array of academic and industrial uses; for geological research, navigating steerable drills, and for certain engineering problems associated with mineral deposit prospecting.

In this study, we focus on how the original spherical harmonic inverse modelling code, developed in the 2010s to convert from scalar to vector representation, was updated and improved. This paper is structured as follows. We first describe the WMAM code and the original goals of the refactoring work. In Section 3, we outline how the code development proceeded, from initial profiling to performance benchmarking on the HPE Cray EX CPU nodes (Sections 4 to 6), through to a second round of code development involving three generations of NVIDIA GPUs (Sections 7 to 8). The science enabled by this work (specifically the results of the degree 2000 model run) are given in Section 6.2. Lastly, the conclusions are presented in Section 9.

2. WMAM code

The WMAM code uses a pre-conditioned non-linear conjugate gradient algorithm, with a quasi-Newton estimate of a diagonal Hessian, to iteratively solve a large non-linear inverse problem. WMAM fits a spherical harmonic representation of a scalar potential field, featuring millions of parameters, to input data consisting of magnetic scalar field anomaly magnitudes, i.e. tens of millions of observations made at or near to the Earth's surface. The problem is thus over-determined in theory, though in practice the magnetic measurements are not uniformly distributed (e.g. fewer data samples over the oceans compared to land regions) and hence the solution requires a mathematical damping parameter to reduce the effects of incomplete sampling and general noise in the data. The problem is non-linear as the spherical harmonic coefficients (the model parameters) are not linearly related to the observations of scalar field (the data), as would be the case for vector field observations.

The spatial resolution of a spherical harmonic model is approximated by $2\pi a/\sqrt{L(L+1)}$, where L is the maximum degree of the model, and a is the Earth's average radius (6 371.2 km); the higher the spherical degree, the greater the spatial resolution of the magnetic field model. The fact that the problem is naively of $\mathcal{O}(L^2)$ is the reason why WMAM performance becomes intractable when computing higher degree models. In addition, the memory requirements of the WMAM code are likely to exceed the available resources if run on HPC platforms that offer greater parallelism but lower memory per CPU. Thirdly, as noted previously, the resolution of the available scalar field measurements (of order 5 km) is too high to be fully exploited by the WMAM code.

The aims of this work were to improve the WMAM code such that it can:

1. Produce models of spherical harmonic degree higher than 1440;

2. Exploit the parallelism offered by a Tier 1 HPC resource such as ARCHER2 (EPCC, 2025a);
3. Take advantage of GPU accelerators so as to minimise computational energy consumption.

The WMAM Fortran source code and supporting libraries are held in a public GitHub repository (BGS, 2025a). Various versions of WMAM are *tagged* in this repo, WMAM v1.2 references the original unrefactored source code and WMAM v4.2 is the version for CPU that incorporates the software redesign described in this paper.

WMAM v1.2 and v4.2 are statically linked to two libraries, the BGS in-house Iterative Global Model Fitting Library (BGS, 2025b) hereafter referred to as GlobLibI, and the SLATEC Common Mathematical Library v4.1 (BGS, 2025f). Both WMAM and GlobLibI call SLATEC functions. A slight modification[1] was made to the SLATEC code which was otherwise unchanged during the redesign of WMAM and GlobLibI. The SLATEC library is freely distributable under a public domain licence (Fong et al., 1993). Hence, our modified SLATEC library is accessible via the aforementioned public repository (BGS, 2025a).

WMAM v4.2 was then used as a starting point for porting WMAM to GPU. As that work required further substantial changes to the source code (e.g. the GlobLibI library was merged with the WMAM code), it was decided to store the GPU version (WMAM v5.0) within a separate branch of the repository, namely, `gpu-openmp`. WMAM v5.0 uses preprocessor constants to control code compilation such that the iterations of two frequently called loops are either (a) distributed across OpenMP threads running on CPU or (b) offloaded to GPU.

3. Methodology

We first profiled WMAM v1.2 to understand how best to improve code performance. After a discussion of the profiling results, we describe the changes made for WMAM v4.2, focusing on those improvements that had the most impact. We then present results showing the speedups achieved for increasing spherical harmonic degree (L). The performance gains realised by WMAM v4.2 make it possible to further resolve the Earth's crustal magnetic field, increasing the resolution from 28 km ($L = 1440$) to 20 km ($L = 2000$). We show for the first time the higher-resolution maps alongside a comparison of the power spectra for different model degrees.

Returning to the WMAM code, we profile v4.2, so as to confirm which parts should be offloaded to GPU for further performance gains. We offload to GPU two loops within the WMAM code that iterate over the data points assigned to each MPI rank. The offloading is achieved using OpenMP directives. This technique guarantees some level of portability between different GPU devices and also minimises changes to the code. The GPU-enabled WMAM v5.0 performance is given in terms of runtime and energy consumption.

The development work involving WMAM v4.2 was done on the ARCHER2 (EPCC, 2025a) machine, an HPE Cray EX system, providing the UK National Supercomputing Service. WMAM v5.0 was developed on Cirrus (EPCC, 2025f), a UK National Tier-2 HPC service that in a previous incarnation featured NVIDIA Volta 100 GPUs (EPCC, 2025e). Following this, WMAM was benchmarked on NVIDIA Ampere 100 GPUs and on a single GraceHopper GH200 Superchip.

Throughout this paper, monospaced font is reserved for (Fortran) code-specific terms such as source files, subroutines and variables. All source filenames are enclosed in double quotes with a "`.f`" suffix. Subroutine names have a "`()`" suffix and variable names are denoted by single quotes. For brevity, the term *degree* will denote the maximum spherical harmonic degree of the resulting field model; *parameter* refers to the parameter(s) of said model, and *resolution* is the spatial resolution of the input data. The spatial resolution of the input data is reduced by resampling the original WDMAM data set of Lesur et al. (2016) using the `blockmedian` function of the Generic Mapping Tools library (Wessel et al., 2019), to apply a grid block averaging function by median estimation. The term *point* refers to a geographic location identified by co-latitude, longitude, radius, and date and time. Lastly, in the following sections, references will be made to real and integer data types; these are 8-byte reals and 4-byte integers.

[1] The array bounds of the arguments to the SLATEC `DS2Y()` subroutine were resized for compatibility with the WMAM code.

4. Initial profiling

We profiled WMAM v1.2 running over 32 fully-populated ARCHER2 compute nodes (4096 MPI ranks). The spherical harmonic degree was set to 1440 and a value of 0.1 ° was used for the spatial resolution of the input data on a grid of equal latitude and longitude.

The code instrumentation used the VI-HPS Score-P v7.1 profiler (VI-HPS, Forschungszentrum Jülich GmbH, Germany, 2025). The instrumented executable ran for approximately 27 700 s (7 h and 41 mins) and produced a "`profile.cubex`" file, which can be viewed using the CubeGUI v4.7.0 client (VI-HPS, 2025).

Fig. 4.1 (produced by CubeGUI) shows percentage splits of an aggregate runtime, that is to say a runtime summed over all parallel processes — 4096 in this case. The left panel of Fig. 4.1 reveals where within the source code the processes spend their time. Over 80% of which is accounted for by the `sub_sph_wmam_l()` subroutine. This part of the WMAM code performs linearised forward modelling of the scalar crustal magnetic field anomaly combined with the main field, for a given set of model parameters at a given datum.

The aggregate runtime due to the `sub_sph_wmam_l()` subroutine *alone* is presented in the right panel (Fig. 4.1), as a percentage split across the 32 compute nodes. There is evident load imbalance: on the last 14 nodes assigned to the job (`nid005402-13` and `nid005436-42`), the percentage split is 3.39%, higher than the 2.92% attributed to the first 17 nodes.

The heart of the WMAM code is an inversion algorithm, which converges towards an optimal parameter set for the spherical harmonic model, one that minimises the differences between the field derived from the model and the measured magnetic field magnitudes. Each parameter set needs to be evaluated. This is done by having the MPI ranks iterate through their assigned coordinate points for which there exist measurements, calculating the magnetic field produced by the current model parameter set and comparing it with the observed values. This looping over the coordinate points is done within the `cpt_dat_vals()` subroutine, which in turn makes heavy use of `sub_sph_wmam_l()`. The `cpt_dat_vals()` subroutine repeatedly calculates individual vector magnetic field component values, or the scalar field (which depends on the three vector field components) for a given set of model parameters at a given datum. Hence, a call to this routine to compute a modelled scalar field value (as our input data are), will result in three calls to compute each vector field component value.

There is a second critical subroutine called `ssqgh_d()` that is used to determine the next set of model parameters based on how well the previous set matched observations. The routine calculates the gradient of the weighted sum of squared differences between predicted data for the current model iteration and the observed data values. Thus the code can score how close it is to a good model. In addition, the routine calculates the diagonal of the Hessian matrix of the inverse problem, to inform us of a suitable step size towards the local optimum. This routine also iterates over the coordinate points, calling `sub_sph_wmam_l()` each time.

These two subroutines, `cpt_dat_vals()` and `ssqgh_d()`, account for 80% of the runtime (split 70% to 10%), and both feature independent loops over geographic coordinates. `cpt_dat_vals()` performs the forward modelling calculation and, to compute the scalar magnetic field, must first compute the three vector field components, hence the significant computation time required. We therefore concluded that optimising those two parts of the WMAM code should have most impact on performance.

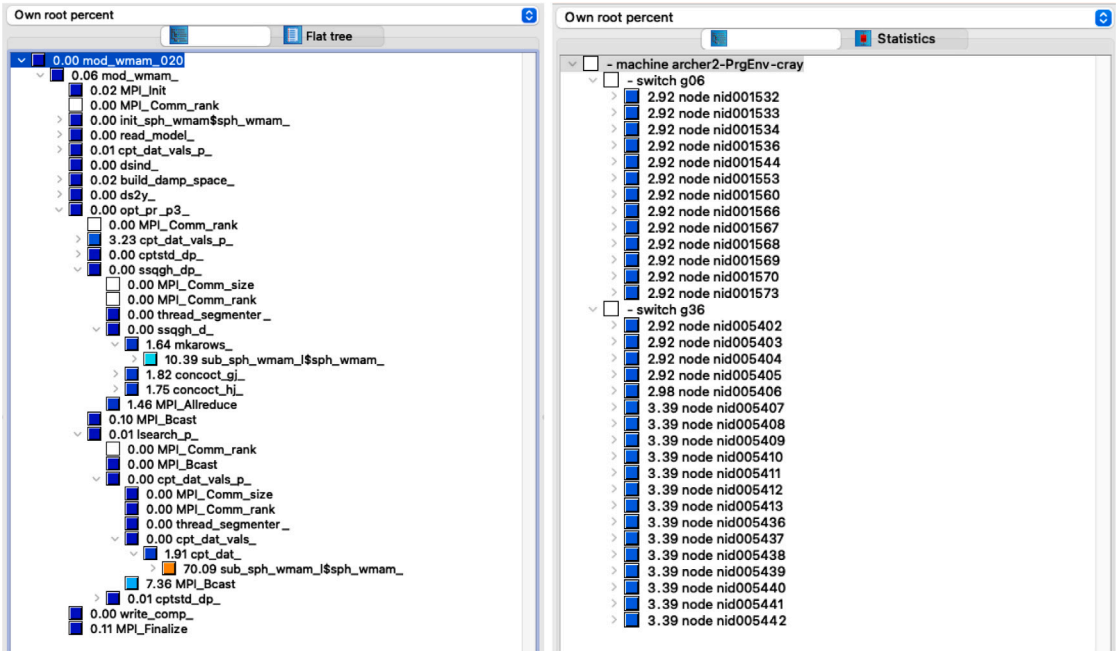

Fig. 4.1. Left, the aggregate runtime attributable to certain WMAM/GlobLibI subroutines when running WMAM v1.2 with degree 1440 and resolution 0.1 °. WMAM was run over 32 fully-populated ARCHER2 nodes (4096 MPI tasks). The `sub_sph_wmam_l()` subroutine accounts for more than 80% of the runtime. **Right**, the aggregate runtime percentage due to the `sub_sph_wmam_l()` subroutine split across the 32 compute nodes.

Table 5.1
The key top-level arrays used within WMAM v1.2 – M denotes one million. The array sizes are unnecessarily large for small degree values (e.g., 200) and insufficient for degree values higher than 1440.

Name	Type	Shape	Purpose
`ppos`	real	[8, 6M]	input data
`bc`	real	[4M]	model parameters
`cov`	real	[18M]	covariance matrix
`ijcov`	integer	[18M, 2]	covariance index matrix
`dw`	real	[6M]	difference between the measured magnetic field magnitudes and those predicted from the model

5. Code development for CPU

This section describes the changes made to WMAM and GlobLibI that did the most to achieve the first two objectives (Section 2).

5.1. Sizing arrays dynamically

The top-level arrays used throughout WMAM and GlobLibI (Table 5.1) were originally of fixed size.

The input data array, '`ppos`', holds eight attributes for each data point, which are, in index order, co-latitude, longitude, radius, year, the X (southward), Y (eastward) and Z (downward) magnetic components of a reference magnetic field model, and finally the magnetic field anomaly magnitude. The reference model comprises the main and lithospheric components of the Comprehensive Model v4 (CM4) (Sabaka et al., 2004), truncated at degree and order 15. The main field values for the year 1990 are used, as this is approximately the midpoint of the era of the input aeromagnetic and marine magnetic dataset.

The arrays in Table 5.1 can in fact be sized according to the degree (L) and resolution (R). The size of the '`bc`' array is based on the number of model parameters, $N_{mp} = L(L+2)$. The other arrays are all sized according to the number of data points, N_{dp}, the expression for which is a little more complicated:

$$N_{dp} = \left(\lfloor\frac{1}{R}\rceil 180 - 1\right)\left(\lfloor\frac{1}{R}\rceil 360\right) + (L+1)(2L+1). \quad (3)$$

The brackets surrounding the $\lfloor 1/R \rceil$ terms indicate rounding to the nearest integer. Note that the expression for N_{dp} has two parts, one involving the resolution and the other the degree. The resolution part gives the number of *input* points used after the `blockmedian` resampling. To reduce the Backus effect (Backus, 1970) of propagation of noise in the direction perpendicular to the main magnetic field when modelling only scalar observations of a vector field, a set of $(L+1)(2L+1)$ *sampled* points are also used. These points are distributed across $2L+1$ evenly spaced longitudes, at latitudes corresponding to the inverse sine of the roots of the Legendre polynomial of degree $L+1$. The *sampled* points have their magnetic field anomaly set to zero, but have the reference field set up in the same way as the *input* points, and have a weighting applied, known as the *damping*, or *regularisation*, parameter. This damping acts to minimise the integral over the sphere of the magnetic crustal anomaly perpendicular to the main field vector direction (Lesur et al., 2016). The percentage of points that are *sampled* points is between 40% and 55% for the degree-200 to degree-1440 model runs, but is below 25% for the degree-2000 model. Fig. 5.1 shows how the total number of data points (input and sampled) and model parameters grow with the degree given the choices of resolution used when resampling the input data. These two counts represent the workload. It can be seen that the number of points outstrips the number of parameters as the degree is increased.

As the WMAM conjugate gradient algorithm proceeds, the magnetic field needs to be determined at each (coordinate) point for the current parameter set. The computational expense of this task is greater for the

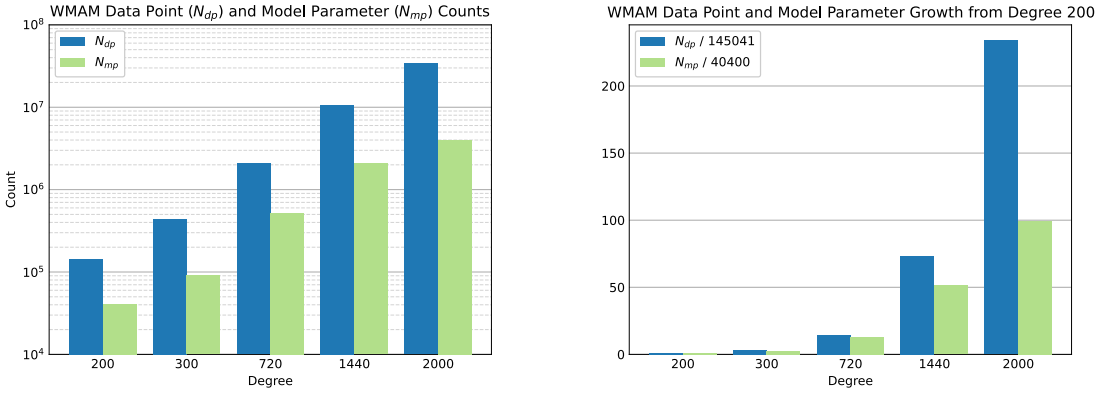

Fig. 5.1. The WMAM point and parameter counts for degrees 200, 300, 720, 1440 and 2000. The resolutions used for resampling the input data points were 1.0 °, 0.5 °, 0.25 °, 0.1 °, and 0.05 ° respectively. The plot on the right scales the counts according to the number of points ($N_{dp} = 145\,041$) and parameters ($N_{mp} = 40\,400$) for a degree 200 model.

Table 5.2
The key top-level arrays used within WMAM v4.2 are sized according to the values of N_{mp} and N_{ppr} ($\approx N_{dp}/N_r$) is the number of points per rank.

Name	Type	Shape	Purpose
`ppos`	real	$[8, N_{ppr}]$	input data
`bc`	real	$[N_{mp}]$	mode parameters
`cov`	real	$[N_{ppr}]$	covariance matrix
`ijcov`	integer	$[N_{ppr} + 2, 2]$	covariance index matrix
`dw`	real	$[N_{ppr}]$	difference between the measured magnetic field magnitudes and those predicted from the model

sampled points than it is for the input points[2] – this has implications for how the workload should be partitioned across MPI tasks.

Table 5.2 gives the new top-level array sizes used in WMAM v4.2, ensuring that the memory required is appropriate to the model degree and resolution.

5.2. Workload partitioning

The points represent the workload that is distributed across the MPI ranks (or tasks). For WMAM v1.2, each MPI rank held an array containing *all* the points with the sampled points following the input points. This array would then be partitioned many times during the inversion, even though the point data did not change. Another issue was that the sampled points existed as a contiguous sequence at the end of the point array, which meant that some MPI ranks would be assigned input points only, while other ranks would get just sampled points (with one rank being assigned a mixture of the two). As mentioned in Section 5.1, determining the magnetic field for the sampled points is more expensive than for the input points, creating a workload imbalance (Fig. 4.1, right).

In WMAM v4.2, the point workload is partitioned just once during the initialisation stage defined in the main WMAM source file, "`mod_wmam_020.f`". Furthermore, the point arrays for each MPI task are constructed such that they contain roughly equal numbers of input and sampled points, balancing the workload more effectively.

5.3. Eliminating large arrays

As mentioned previously, most of the runtime is spent within `sub_sph_wmam_l()` and the most expensive part of that subroutine is the call to `XYZsph_bi0()`,[3] which populates three double precision model parameter arrays. Each array is of size $L(L+2)$, where L is the maximum degree. Three such arrays overflow the ARCHER2 L3 cache for $L \geq 1440$, thereby hampering performance. It turns out, however, that for two of the three scenarios under which `XYZsph_bi0()` is called, the arrays are merely reduced to dot products. Listing (BGS, 2025e) shows this for the scenario that involves sampled points. We can therefore improve performance by implementing variants of `XYZsph_bi0()` for each scenario and simply accumulate dot products within the appropriate variant subroutines, circumventing the need to handle three large arrays, see Listing (BGS, 2025g).

6. CPU results

We now review the performance gains achieved by WMAM v4.2 before confirming the successful calculation of a degree-2000 model. The WMAM code and libraries (GlobLibI and Slatec) were compiled using the HPE Cray compiler CCE 15.0.0 that is provided by the 22.12 Cray Programming Environment (CPE) available on ARCHER2. All code was compiled at Optimisation level 3.

6.1. Performance

We profiled WMAM v4.2, visualising the results (Fig. 6.1) according to the scheme described in Section 4. This time the run completed within 1 h and 40 min, a speed-up of 4.7. Load imbalance also improved, dropping to around 6% of what it was previously, a consequence of the code changes discussed in Section 5.2.

Table 6.1 shows the performance of the WMAM code on ARCHER2, which features AMD EPYC 7742 64-core 2.0 GHz[4] processors, arranged as 5860 nodes of two processors (and so each node has 128 cores) – the memory per node is 256 GB. The WMAM v1.2 runtimes rise significantly, from 3 min to 30 h, as the maximum model degree is increased.

[2] The sampled points are dependent on the model parameters such that they need to be recalculated for each new parameter set.

[3] `XYZsph_bi0()` accounts for around 62% of the time spent in `sub_sph_wmam_l()`.

[4] ARCHER2 CPU frequency can be set as high as 2.25 GHz, but its default setting as of 12 Dec 2022 is 2.0 GHz.

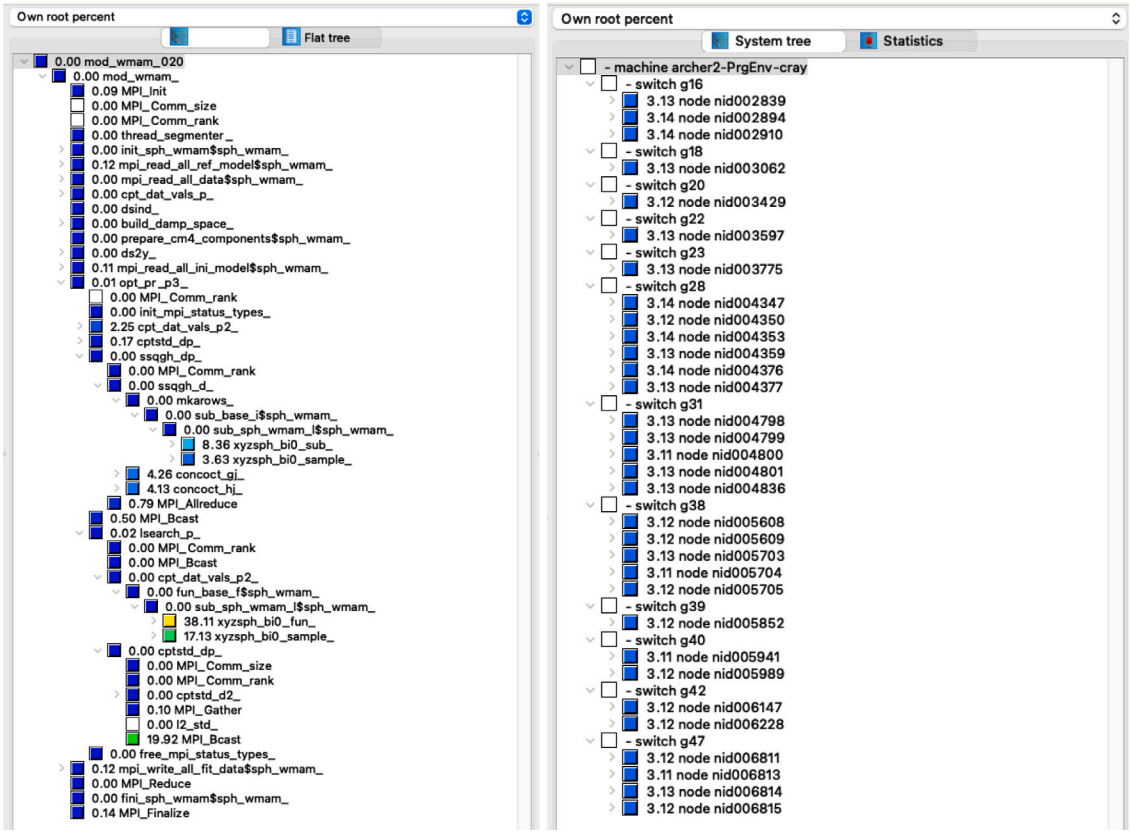

Fig. 6.1. Left, the aggregate runtime attributable to certain WMAM/GlobLibI subroutines when running WMAM v4.2 with degree 1440 and resolution 0.1 °. WMAM was run over 32 fully-populated ARCHER2 nodes (4096 MPI tasks). The `sub_sph_wmam_l()` subroutine accounts for nearly 70% of the runtime. **Right**, the aggregate runtime percentage due to the `sub_sph_wmam_l()` subroutine split across the 32 compute nodes.

Table 6.1
WMAM performance on ARCHER2 for versions v1.2 and v4.2. The degree 200 and 300 runs had the 64 MPI tasks assigned to the first 64 cores on the node, whereas the degree 720 and 1440 runs had fully-populated nodes. Runtimes are an average of three runs (all standard deviations were within 2% of the mean). Speedup is simply the runtime listed for WMAM v1.2 divided by the v4.2 result.

Degree	Resolution [°]	Nodes	Tasks	Runtime [s]		Speedup
				v1.2	*v4.2*	
200	1.00	1	64	184	70	2.63
300	0.50	1	64	1 504	426	3.53
720	0.25	2	256	22 772	3 989	5.71
1440	0.10	8	1024	107 850	19 332	5.58

On the other hand, the runtimes for WMAM v4.2, the version developed by EPCC, rise much less dramatically. Compared to WMAM v1.2, a significant speedup is maintained for all degree values.

6.2. Scientific output

There are several types of scientific studies that become feasible or are improved with global spherical harmonic models, particularly the ability to vertically continue the field, as compared to the input gridded map of the scalar field (e.g. Hemant and Maus, 2005; Williams et al., 2025). One of the science outputs of the research is the ability to robustly extrapolate the field to other altitudes and to extract the directional components locally from scalar observations. The extensive code alterations described above make it feasible to run at $L = 2000$, thereby achieving the first objective.

We can analyse the output spherical harmonic parameters in spectral terms as power per degree. The mean square value of the vector geomagnetic field over a sphere of radius r due to all harmonics of SH degree l for a given model is defined as

$$R_l = (l+1)\left(\frac{a}{r}\right)^{2l+4} \sum_{m=0}^{l} \left[(g_l^m)^2 + (h_l^m)^2\right]. \tag{4}$$

This is known as the Lowes-Mauersberger power spectrum (Lowes, 1974). Typically $r = a$, where a is the Earth's average geophysical radius (6 371.2 km). Gauss coefficients g_l^m and h_l^m are the weightings given to each spatial basis function of a given spherical harmonic degree and order, and are therefore our model parameters that we solve for with the WMAM code. From the spectra, we can determine the overall energy of the model along with the effect of changing the damping parameter. The power spectra of the degree 1440 (28 km) and prospective degree 2000 (20 km) model are shown in Fig. 6.2, along with the spectral difference between the models and the correlation per SH degree.

Fig. 6.2 shows similar spectral content and a strong spectral correlation between the outputs of the WMAM code and the BGS benchmark, above 0.9 at all SH degrees for the equivalent 28 km models. When extended from maximum SH degree of $L = 1440$ to $L = 2000$, it was found that the damping parameter needed to be increased from 5 to 10, to get an approximate match in the peak of spectral power around SH degree 150 and a similar fall off of power with increasing SH degree. This spectral shape is expected, both from existing models (such as the BGS 28 km benchmark), and the statistical lithospheric magnetisation model of Thébault and Vervelidou (2015). We see a closer spectral match between the WMAM $L = 2000$ model and the BGS $L = 1440$ benchmark than between the two $L = 1440$ models, for SH degrees 100 to 700. At smaller spatial wavelengths we see the spectral correlation decrease (though the spectral difference is of a similar magnitude) as

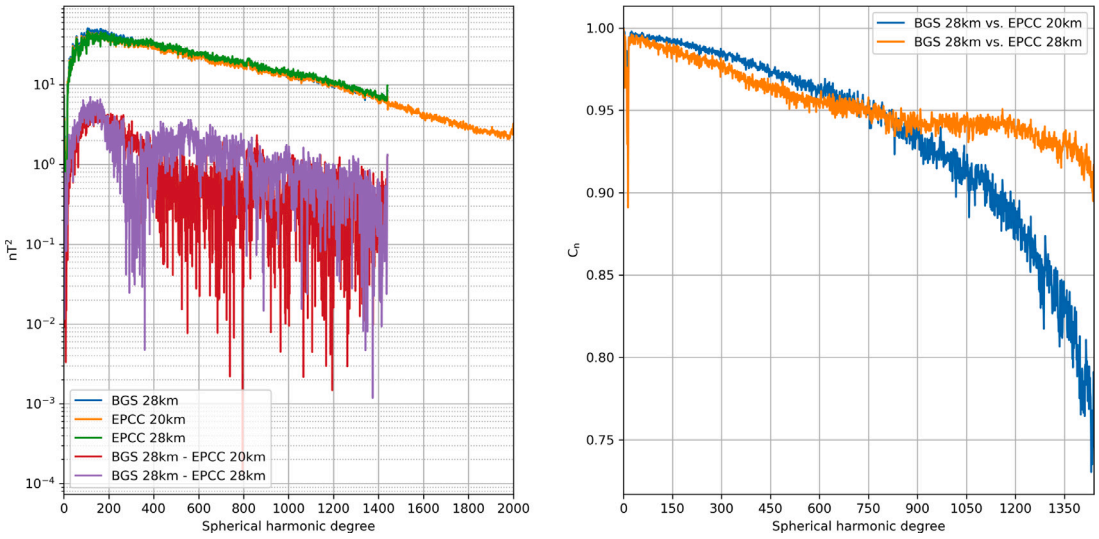

Fig. 6.2. Lowes-Mauersberger spectra (left) produced by WMAM v4.2 (28 km, green; 20 km, orange) and compared against the BGS 28 km benchmark model (blue). Spectral differences between the BGS 28 km benchmark and the WMAM models are also shown (28 km, purple; 20 km, red). Correlation per spherical harmonic degree (right) between BGS 28 km benchmark and WMAM (28 km, orange; 20 km, blue).

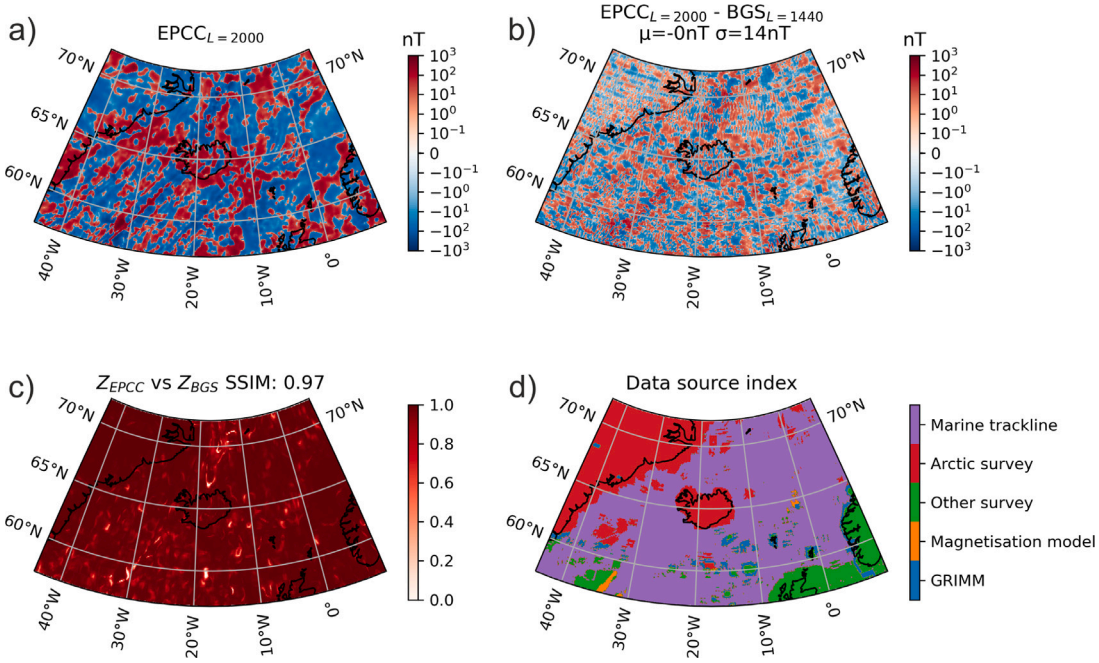

Fig. 6.3. Validating the new 20 km model around Iceland: a) modelled vertical magnetic field at 20 km resolution; b) difference between 20 km ($L = 2000$) and 28 km ($L = 1440$) models; c) spatial correlation between 20 km and 28 km models; d) distribution of data sources in region.

more signal is accommodated by the additional small scale features in the WMAM $L = 2000$ model.

We demonstrate the coherence of these additional smaller scale signals in Fig. 6.3, which shows four panels with contextual information of the crustal magnetic field in the North Atlantic Ocean around Iceland and Greenland. Fig. 6.3a shows the vertical field magnetic anomaly (i.e. the field remaining once the core and external fields have been subtracted). Fig. 6.3b shows the differences between the 20 km and 28 km resolution models are small scale and do not appear to have an obvious correlation to geological structures seen in Fig. 6.3a. The differences between models have zero mean and a standard deviation of 14 nT, in the region depicted. A clearer view is given by the spatial correlation of the 20 km and 28 km maps in Fig. 6.3c. Here we see that the spatial correlation is very high, with an overall structural similarity index value (SSIM, Wang et al., 2004) of 0.97. The majority of the region shows excellent correlation, indicating that the smaller spatial scale information in the 20 km model has mainly modified the amplitude of signals, i.e. there is no new or spurious structure introduced. Indeed, Fig. 6.3d confirms there is no correlation between the regions of varying quality data sources (see Lesur et al., 2016) and the differences between the two models. The areas of lowest correlation are small features aligned trending south-west to north-east, and correspond to known geological features — the ocean bottom magnetic striping caused by the opening of the mid-Atlantic rift (Vine and Matthews, 1963), which is better captured by the additional small scales of the 20 km model.

Table 7.1
The read-only arrays defined by WMAM v5.0 which are mapped to the GPU devices once only before the main application loop. L is the model degree and N_{ppr} is the number of points per MPI rank.

Name	Type	Shape	Purpose
`ppos`	real	$[8, N_{ppr}]$	input data
`cov`	real	$[N_{ppr}]$	covariance matrix
`jcov`	integer	$[N_{ppr}+2]$	one half of covariance index matrix, '`ijcov`' (Table 5.2)
`d2a`	real	$[L+1]$	array for holding pre-computed values

7. Code development for GPU

This section covers the changes made to WMAM required to meet the last of the three objectives listed in Section 2. We previously identified two loops where WMAM spends the majority of its execution, one is the main part of the `cpt_dat_vals()` subroutine and the other has a similar prominence within `ssqgh_d()`. It proved straightforward to assign the iterations of those loops to OpenMP threads. We now discuss how the loops were offloaded to GPU.

7.1. MPI

Each MPI task has exclusive access to one GPU. This means that for the Cirrus machine (EPCC, 2025e), the WMAM code runs four MPI tasks per node, as each Cirrus GPU node has four NVIDIA V100 GPUs. Such an arrangement can be established at runtime via the `omp_set_default_device()` subroutine.

7.2. Data mapping

Several of the top-level arrays (Table 5.2) are only ever read once they have been populated from input file data, i.e. the arrays are constant after initialisation.

For that reason, those arrays, identified by Table 7.1 are mapped to the device just once using an OpenMP `target data` region that surrounds the main application loop of the inversion algorithm (see the `opt_pr_p3()` subroutine). In addition, it is also necessary to allocate two arrays on each device of size $L+1$, which will later be declared thread-private — this is done with an OpenMP `target enter data` directive placed just inside the first data region. Those two arrays are used by the device code to store interim calculations.

7.3. Loop offloading

The loops within the `cpt_dat_vals()` and `ssqgh_d()` subroutines — hereafter referred to as `cpt` and `ssqgh` – are both split in two: the first loop iterates over the input points whereas the second iterates over the sampled points. Using the OpenMP `target` directive, a loop's iterations are executed on the GPU as a set of threads or kernel, see Listing (BGS, 2025c). It is important that the threads within the same kernel do the same number of computations as this minimises the time that threads are idle. Hence, as it is known that input and sampled points require different amounts of computation (Section 5.1), the two types of data point are handled by separate GPU kernels.

The `cpt` loops can now be offloaded straightforwardly. All kernel threads can share arrays stored in the global memory of the GPU. However, the parameter array ('`bc`') needs to be mapped to the GPU each time control passes to the `cpt` subroutine (the parameter set changes as the algorithm converges). As regards output, each thread writes to a unique element of an array, the size of which matches the number of data points (input + sampled).

The situation is more complex for the `ssqgh` loops, owing to the fact that each kernel thread calculates a pair of values for every model parameter, amounting to $16\,L(L+2)$ bytes. The NVIDIA V100 GPUs on Cirrus have 16 GB of global memory and 80 streaming multiprocessors (SM). Thus, for a model of degree $L = 1440$, there would only be sufficient memory to have 2–3 threads per SM, severely under-utilising the GPU device cores. A single pair of parameter arrays is therefore held in GPU global memory with each GPU thread updating the array elements atomically via the OpenMP `atomic update` construct. In this way, each element of the two global arrays is a sum of contributions from all GPU threads.

Table 8.1
The performance of WMAM v5.0 CPU on Cirrus for various model degrees. The runtimes and energies are averages of three runs (measured deviations were < 3% of mean). The speedup is relative to the WMAM v4.2 runtimes, see Table 6.1.

Degree	Nodes	Tasks	Runtime [s]	Speedup	Energy usage [kWh]
200	1	4	109	1.03	0.012
300	2	8	338	1.01	0.076
720	4	16	4 380	1.46	2.047
1440	8	32	40 818	1.52	39.073

Table 8.2
The performance of WMAM v5.0 GPU on Cirrus. Runtimes and energies are averaged over three runs (measured deviations were < 3% of mean, except for degree 200 model, which exhibited variances of 4-6%). Total energy usage is that consumed by the CPU host(s) and the GPU devices. The speedup is relative to WMAM v5.0 CPU (Table 8.1).

Degree	Nodes	Tasks	Runtime [s]	Speedup	Energy usage [kWh]		
					Host	*Device*	*Total*
200	1	4	23	4.74	0.002	0.002	0.004
300	2	8	56	6.04	0.008	0.016	0.024
720	4	16	965	4.54	0.283	0.675	0.958
1440	8	32	11 164	3.66	6.731	14.933	21.664

8. GPU results

We ran CPU and GPU flavours of WMAM v5.0 on the Cirrus GPU nodes for a range of model degrees. WMAM v5.0 CPU was compiled using GCC 10.2.0 and OpenMPI 4.1.6, and WMAM v5.0 GPU was compiled using NVIDIA nvfortran 24.5 and a version of OpenMPI 4.1.6 specially built for CUDA 12.4. As with WMAM v4.2, all code was compiled at optimisation level 3. The Cirrus GPU nodes contain 384 GB of memory and two 20-core Intel Xeon Gold 6148 (Cascade Lake) series processors. WMAM v5.0 CPU was run with four MPI tasks per node, 10 OpenMP threads per task and one host CPU core per OpenMP thread; whereas WMAM v5.0 GPU was run with four MPI tasks per node and one NVIDIA V100 GPU per task.

For Table 8.1, the energies and runtimes were obtained following job completion via the Slurm (SchedMD, 2025) (v22.05.11) `sacct` command[5] with the energy data coming from the Intel Running Average Power Limit (RAPL) counters. Thus, the energy data presented in Table 8.1 covers host hardware only, i.e. the two Cascade Lake processors.

The same procedure is followed for Table 8.2, but with additional calls to the NVIDIA System Management Interface program, `nvidia-smi` (Nvidia Corporation, 2025), so as to retrieve time-stamped power readings for the GPUs during the run. Instances of the `nvidia-smi` command were run per GPU node such that power readings from all GPUs were made every second, see Listing (BGS, 2025d). The total energy consumption was calculated by passing the power readings from each GPU to a numerical integration method[6] provided by the SciPy Python package and then adding the result to the energy reported by the Slurm `sacct` command. We tested whether the GPU power profiles

[5] `sacct -j <jobid> --format=JobID%20,JobName%20,Partition,Elapsed,ConsumedEnergyRaw`.

[6] SciPy v1.10.1 `integrate.cumulative_trapezoid()`.

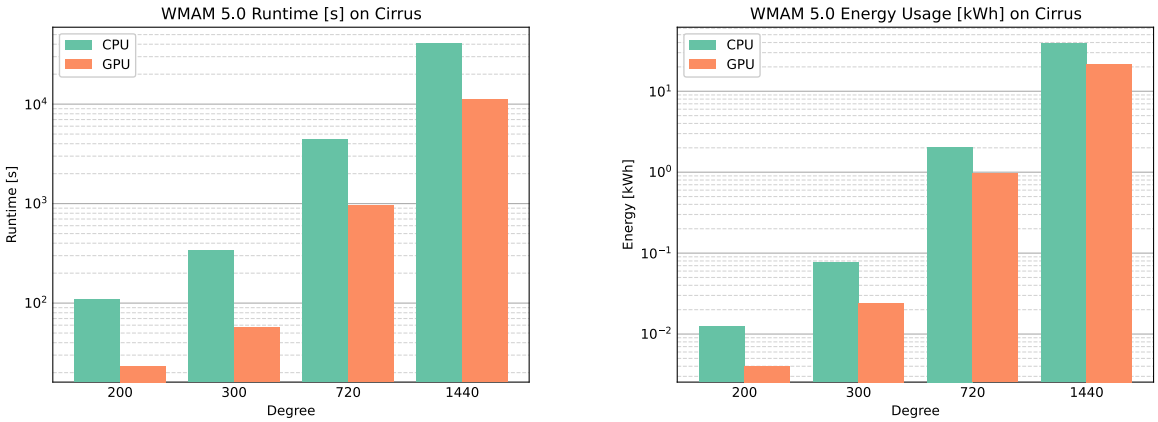

Fig. 8.1. The WMAM v5.0 CPU/GPU runtime and energy usage for degrees 200, 300, 720 and 1440. Both the CPU and GPU runs used 4 MPI tasks per Cirrus GPU node. The CPU runs used 10 Intel Cascade Lake cores per task, whereas the GPU runs used one NVIDIA V100 GPU per task. Starting from 1, the Cirrus GPU node count doubles for successive degree values (the GPU count also doubles but starts from 4).

were being undersampled by lowering the read frequency to once per millisecond for all model sizes. This produced no significant deviation in GPU energy, confirming the accuracy of the energy values presented in Table 8.2.

To clarify, Table 8.1 records the energy due to the host CPU processors only as those runs did not use GPUs. The runs referenced by Table 8.2 however do use the GPUs, which, of course, must be accessed via the CPU processors and so, the total energy given in Table 8.2 is the sum of the host and device energies.

In Fig. 8.1, we compare the CPU and GPU performance. At first glance, it appears that the gap between GPU and CPU narrows as we increase the model degree. This impression is due to the logarithmic scaling. The performance boost due to GPU is in fact more or less consistent for the different model sizes. The energy saving relative to WMAM v5.0 CPU ranges from 45% to 68%, and the reduction in runtime is between 73% and 83%. Essentially, this comparison pits one NVIDIA V100 GPU against ten Intel Cascade Lake CPUs.

The NVIDIA V100 range of GPUs was launched in 2017, and since then NVIDIA have released the A100 and H100/200 GPUs, as well as the Grace Hopper Superchip that features unified host and device memory. The main benefits of these later architectures are the greater memory size and improved memory performance. For example, the 16 GB available on the V100 is too little to compute a degree-2000 WMAM model. This memory restriction also means that the WMAM v5.0 CPU code benefits from an optimisation not available to the GPU version, specifically, the use of an additional thread-private array for caching powers of the radius.

The DiRAC Extreme Scaling Service known as Tursa (EPCC, 2025h), features NVIDIA A100 GPUs, which have sufficient memory (40 GB per GPU) for the WMAM GPU code to accommodate the extra thread-private array. We ran the optimised WMAM v5.0 GPU code on thirty-two A100 GPUs (across eight GPU nodes).[7] For the degree-1440 model, the average runtime over three runs was $3\,379 \pm 8.9$ s, about 56 mins. Compared to the Cirrus GPU result, the runtime has reduced by a factor of 3.3. Unfortunately, the Slurm installation on Tursa is not set up to measure the energy use due to the host hardware. We instead estimate this value by taking the host energy measured for Cirrus GPU nodes (6.731 kWh) and divide by 3.3 to give 2.04 kWh. This value can then be added to the energy measured (via `nvidia-smi` for the Tursa GPUs), 7.180 ± 0.025 kWh, which gives 9.22 kWh. Hence, relative to the WMAM v5.0 CPU performance, the speedup has increased from 3.66 to 12.08 and the energy reduction has improved from 45% to 76%.

We also ran the optimised WMAM v5.0 GPU code on a single Grace Hopper (GH200) Superchip, supported internally at EPCC, which has more than enough unified memory (480 GB on host and 96 GB on device) to run the 1440 model.[8] The average runtime for this model (again over three runs) was $43\,208 \pm 3$ s, just over 12 h, and the energy consumed by the GH200 Superchip was 6.683 ± 0.004 kWh. Compared to the Tursa A100 GPUs, this represents a further speedup (per GPU) of 2.5 times, together with an additional 28% reduction in energy usage.

9. Conclusions

Inverse modelling of the lithosphere to a high spherical harmonic degree is a computationally intensive problem. The outputs can be used for scientific research as well as for applications in navigation. We substantially refactored the WMAM code such that it can take advantage of the parallelism offered by the ARCHER2 supercomputer. As a result it is possible to compute a degree-1440 model in less than an hour using 64 ARCHER2 nodes (8 192 CPU cores). The same number of ARCHER2 nodes is sufficient for WMAM to compute a degree-2000 model in under 4 h. Profiling the WMAM code reveals that the vast majority of the runtime is spent within just two loops. This presents a clear starting point for achieving further performance gains via GPU offloading. We demonstrate and quantify these gains by benchmarking WMAM on three generations of NVIDIA GPU hardware. Using the host part of a Cirrus GPU node as our baseline (Table 8.1), we find that, on a per-GPU basis, successive iterations of NVIDIA hardware consistently reduce runtime and energy use. Examining all of the runtimes measured indicates that an NVIDIA GPU node is equivalent to multiple ARCHER2 nodes. For Cirrus ($V100 \times 4$) this multiple is two, for Tursa ($A100 \times 4$) it is seven and for GH200 ($\times 1$) it is four.

This work represents a significant advancement for computation of high degree models in geomagnetism and the wider potential field community. We have made the code publicly available to allow further research into lithospheric modelling.

CRediT authorship contribution statement

Michael Bareford: Writing – review & editing, Writing – original draft, Software, Investigation, Data curation. **William Brown:** Writing – review & editing, Validation, Conceptualization. **Ciarán Beggan:** Writing – review & editing. **Callum Watson:** Writing – review & editing, Visualization, Validation. **Mark Bull:** Writing – review & editing, Software, Investigation.

[7] On Tursa, WMAM was compiled using nvfortran 23.5 and a version of OpenMPI 4.1.5 specially built for CUDA 12.1.

[8] On the GH200 Superchip, WMAM was compiled using nvfortran 24.3.

Computer code availability

The WMAM code is stored within a publicly accessible GitHub repository (BGS, 2025a), which has been available since July 2025. This repository also contains the source code for two libraries that are statically linked to WMAM, namely, GlobLibI and SLATEC. All three software items are written in Fortran. Various utilities for preparing input data and for postprocessing WMAM output files are also stored in the repo. Some of these utilities are written in C, while others are written in Bash or Python 3.9 (or above). How the repository is organised is outlined on the landing page with links to lower-level README files.

Multiple versions of the WMAM code were used to produce the results presented in this paper. Versions 1.2 and 4.2 are stored on the main branch of the WMAM repo. These can be retrieved by running the git clone command and then "`git checkout vX.Y`" from the repo folder. WMAM 5.0 exists on a separate branch; this is accessed by specifying "`-b gpu-openmp`" when cloning the repo (an explicit checkout command is not required).

Versions 1.2 and 4.2 of WMAM were run on ARCHER2 and compiled using the HPE Cray Compiler CCE 15.0.0 with inter-process communications being handled by the Cray MPICH 8.1.23a library (EPCC, 2025b). WMAM 5.0 was run on the now retired Cirrus SGI ICE XA system (EPCC, 2025e) where it was built for CPU (EPCC, 2025c) and GPU (EPCC, 2025d) depending on which preprocessor constants had been defined in the build script, see preceding citations. WMAM 5.0 was built using GCC gfortran 10.2.0 when compiling for CPU and NVIDIA nvfortran 24.5 for GPU. For both hardware environments, MPI communications were handled by OpenMPI 4.1.6.

The hardware requirements for running WMAM should be met by any HPC system featuring CPU processors and GPU accelerators that has installed a Linux-based OS along with a job scheduling system such as Slurm (SchedMD, 2025), which allows the user to run the WMAM code in parallel across multiple hardware devices, be they CPU or a combination of CPU and GPU.

The code developers responsible for implementing and testing the changes to the WMAM software discussed here are Michael Bareford and Mark Bull of EPCC (EPCC, 2025g), University of Edinburgh.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work used the ARCHER2 UK National Supercomputing Service (https://www.archer2.ac.uk) and the Cirrus UK National Tier-2 HPC Service at EPCC (http://www.cirrus.ac.uk) funded by the University of Edinburgh and EPSRC (EP/P020267/1). This work also used the DiRAC Extreme Scaling service (Tursa) at the University of Edinburgh, managed by the EPCC on behalf of the STFC DiRAC HPC Facility (www.dirac.ac.uk). The DiRAC service at Edinburgh was funded by BEIS, UKRI and STFC capital funding and STFC operations grants. DiRAC is part of the UKRI Digital Research Infrastructure. Lastly, we thank Dr Vincent Lesur, the original developer, for providing the initial version of the WMAM code.

Data availability

See the section of the manuscript entitled Computer code availability.

References

Alken, P., Thébault, E., Beggan, C.D., Amit, H., Aubert, J., Baerenzung, J., Bondar, T.N., Brown, W.J., Califf, S., Chambodut, A., et al., 2021. International Geomagnetic Reference Field: The thirteenth generation. Earth, Planets Space 73, 1–25. http://dx.doi.org/10.1186/s40623-020-01288-x.

Backus, G.E., 1970. Non-uniqueness of the external geomagnetic field determined by surface intensity measurements. J. Geophys. Res. (1896-1977) 75 (31), 6339–6341. http://dx.doi.org/10.1029/JA075i031p06339.

Beamish, D., White, J.C., 2011. Aeromagnetic data in the UK: a study of the information content of baseline and modern surveys across Anglesey, North Wales. Geophys. J. Int. 184 (1), 171–190. http://dx.doi.org/10.1111/j.1365-246X.2010.04852.x.

BGS, 2025a. eCSE04-8 GitHub Repo. https://github.com/wb-bgs/WMAM. (Online Accessed 3 April 2025).

BGS, 2025b. GlobLibI README.md in eCSE04-8 GitHub Repo. https://github.com/wb-bgs/WMAM/tree/main/libs/globlibi. (Online Accessed 3 April 2025).

BGS, 2025c. Loop Offload. https://github.com/wb-bgs/WMAM/blob/main/listings/loop-offload.md. (Online Accessed 28 October 2025).

BGS, 2025d. Reading NVIDIA GPU power. https://github.com/wb-bgs/WMAM/blob/main/listings/nvidia-smi.md. (Online Accessed 25 June 2025).

BGS, 2025e. Sampled dot product. https://github.com/wb-bgs/WMAM/blob/main/listings/sampled-dot-product.md. (Online Accessed 28 October 2025).

BGS, 2025f. SLATEC README.md in eCSE04-8 GitHub Repo. https://github.com/wb-bgs/WMAM/tree/main/libs/slatec. (Online Accessed 3 April 2025).

BGS, 2025g. Variants of XYZsph_bi0 subroutine. https://github.com/wb-bgs/WMAM/blob/main/listings/variants-of-XYZsph_bi0.md. (Online Accessed 28 October 2025).

EPCC, 2025a. ARCHER2 hardware. https://docs.archer2.ac.uk/user-guide/hardware/. (Online Accessed 3 April 2025).

EPCC, 2025b. Building WMAM on ARCHER2. https://github.com/wb-bgs/WMAM/blob/main/apps/WMAM/build.sh. (Online Accessed 30 October 2025).

EPCC, 2025c. Building WMAM on Cirrus for CPU. https://github.com/wb-bgs/WMAM/blob/gpu-openmp/apps/WMAM/build-cirrus-gnu.sh. (Online Accessed 30 October 2025).

EPCC, 2025d. Building WMAM on Cirrus for GPU. https://github.com/wb-bgs/WMAM/blob/gpu-openmp/apps/WMAM/build-cirrus-nvfortran.sh. (Online Accessed 30 October 2025).

EPCC, 2025e. Using the Cirrus GPU Nodes. https://github.com/EPCCed/legacy-cirrus-docs/blob/main/docs/user-guide/gpu.md. (Online Accessed 11 December 2025).

EPCC, 2025f. Cirrus EX4000. https://www.cirrus.ac.uk/. (Online Accessed 11 December 2025).

EPCC, 2025g. EPCC Contact Us. https://www.epcc.ed.ac.uk/contact. (Online Accessed 27 October 2025).

EPCC, 2025h. Tursa hardware. https://epcced.github.io/dirac-docs/tursa-user-guide/hardware/. (Online Accessed 27 May 2025).

Finlay, C.C., Kloss, C., Olsen, N., Hammer, M.D., Tøffner-Clausen, L., Grayver, A., Kuvshinov, A., 2020. The CHAOS-7 geomagnetic field model and observed changes in the South Atlantic Anomaly. Earth, Planets Space 72 (1), 156. http://dx.doi.org/10.1186/S40623-020-01252-9.

Fong, K.W., Jefferson, T.H., Suyehiro, T., Walton, L., 1993. Guide to the SLATEC Common Mathematical Library. https://www.netlib.org/slatec/guide. (Online Accessed 9 May 2025).

Gauss, C.F., 1839. Resultate Aus Den Beobachtungen Des Magnetischen Vereins Im Im Jahre 1838. Weidmannsche Buchhandlung, Leipzig.

Glassmeier, K.-H., Tsurutani, B.T., 2014. Carl Friedrich Gauss - General Theory of Terrestrial Magnetism – a revised translation of the german text. Hist. Geo Space Sci. 5 (1), 11–62. http://dx.doi.org/10.5194/hgss-5-11-2014.

Hemant, K., Maus, S., 2005. Geological modeling of the new CHAMP magnetic anomaly maps using a geographical information system technique. J. Geophys. Res.: Solid Earth 110 (B12), http://dx.doi.org/10.1029/2005JB003837.

Langel, R.A., 1987. In: Jacobs, J.A. (Ed.), Geomagnetism, vol. 1, Academic Press, chapter 4. The Main Field.

Langel, R.A., Hinze, W.J., 1998. The Magnetic Field of the Earth's Lithosphere: The Satellite Perspective. Cambridge University Press.

Lesur, V., Hamoudi, M., Choi, Y., Dyment, J., Thébault, E., 2016. Building the second version of the World Digital Magnetic Anomaly Map (WDMAM). Earth, Planets Space 68 (1), 27. http://dx.doi.org/10.1186/s40623-016-0404-6.

Lowes, F.J., 1974. Spatial Power Spectrum of the Main Geomagnetic Field, and Extrapolation to the Core. Geophys. J. Int. 36 (3), 717–730. http://dx.doi.org/10.1111/j.1365-246X.1974.tb00622.x.

Maus, S., Barckhausen, U., Berkenbosch, H., Bournas, N., Brozena, J., Childers, V., Dostaler, F., Fairhead, J.D., Finn, C., von Frese, R.R.B., Gaina, C., Golynsky, S., Kucks, R., Lühr, H., Milligan, P., Mogren, S., Müller, R.D., Olesen, O., Pilkington, M., Saltus, R., Schreckenberger, B., Thébault, E., Caratori Tontini, F., 2009. EMAG2: A 2–arc min resolution Earth Magnetic Anomaly Grid compiled from satellite, airborne, and marine magnetic measurements. Geochem. Geophys. Geosyst. 10 (8), Q08005. http://dx.doi.org/10.1029/2009GC002471.

Maus, S., Yin, F., Lühr, H., Manoj, C., Rother, M., Rauberg, J., Michaelis, I., Stolle, C., Müller, R.D., 2008. Resolution of direction of oceanic magnetic lineations by the sixth-generation lithospheric magnetic field model from CHAMP satellite magnetic measurements. Geochem. Geophys. Geosyst. 9 (7), http://dx.doi.org/10.1029/2008GC001949.

Nvidia Corporation, 2025. NVIDIA System Management Interface program. https://docs.nvidia.com/deploy/nvidia-smi/index.html. (Online Accessed 15 April 2025).

Olsen, N., Ravat, D., Finlay, C.C., Kother, L.K., 2017. LCS-1: a high-resolution global model of the lithospheric magnetic field derived from CHAMP and Swarm satellite observations. Geophys. J. Int. 211 (3), 1461–1477. http://dx.doi.org/10.1093/gji/ggx381.

Olsen, N., Stolle, C., 2012. Satellite geomagnetism. Annu. Rev. Earth Planet. Sci. 40, 441–465. http://dx.doi.org/10.1146/annurev-earth-042711-105540.

Sabaka, T.J., Olsen, N., Purucker, M.E., 2004. Extending comprehensive models of the earth's magnetic field with Ørsted and CHAMP data. Geophys. J. Int. 159 (2), 521–547. http://dx.doi.org/10.1111/j.1365-246X.2004.02421.x, arXiv:https://academic.oup.com/gji/article-pdf/159/2/521/5990459/159-2-521.pdf.

SchedMD, 2025. Slurm Workload Manager Documentation. https://slurm.schedmd.com/documentation.html. (Online Accessed 27 October 2025).

Thébault, E., Hulot, G., Langlais, B., Vigneron, P., 2021. A spherical harmonic model of earth's lithospheric magnetic field up to degree 1050. Geophys. Res. Lett. 48 (21), http://dx.doi.org/10.1029/2021GL095147, e2021GL095147.

Thébault, E., Vervelidou, F., 2015. A statistical spatial power spectrum of the earth's lithospheric magnetic field. Geophys. J. Int. 201 (2), 605–620. http://dx.doi.org/10.1093/gji/ggu463.

VI-HPS, G., 2025. Cube GUI User Guide. https://apps.fz-juelich.de/scalasca/releases/cube/4.7/docs/guide/html/. (Online Accessed 3 April 2025).

VI-HPS, Forschungszentrum Jülich GmbH, Germany, 2025. SCORE-P - Scalable Performance Measurement Infrastructure for Parallel Codes. https://www.vi-hps.org/projects/score-p/. (Online Accessed 3 April 2025).

Vine, F.J., Matthews, D.H., 1963. Magnetic anomalies over oceanic ridges. Nature 199 (4897), 947–949. http://dx.doi.org/10.1038/199947a0.

Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. 13 (4), 600–612. http://dx.doi.org/10.1109/TIP.2003.819861.

Wessel, P., Luis, J.F., Uieda, L., Scharroo, R., Wobbe, F., Smith, W.H.F., Tian, D., 2019. The Generic Mapping Tools Version 6. Geochem. Geophys. Geosystems 20 (11), 5556–5564. http://dx.doi.org/10.1029/2019GC008515.

Williams, S., Gubbins, D., Whittaker, J., Seton, M., 2025. Magnetization of oceanic lithosphere from modeling of satellite observations. J. Geophys. Res.: Solid Earth 130 (4), http://dx.doi.org/10.1029/2024JB030370.