Software Toolkit for Fast High-Resolution TMS Modeling

Sergey N. Makarov^{1,2}, Gregory M. Noetscher¹, Edward H. Burnham¹, Dung Ngoc Pham¹,
Aung Thu Htet¹, Lucia Navarro de Lara², Tommi Raji³, and Aapo Nummenmaa²

¹Electrical & Computer Eng. Dept., Worcester Polytechnic Institute, Worcester, MA 01609 USA

²Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital, Harvard Medical School, Boston, MA 02115 USA

³Center for Brain Stimulation, Shirley Ryan AbilityLab, Chicago, IL 60611; Department of Physical Medicine and Rehabilitation, Department of Neurobiology, Northwestern University, Chicago IL 60611 USA

Correspondence:

makarov@wpi.edu, gregn@wpi.edu, ehburnham@wpi.edu, dnpham@wpi.edu,
aungthuhtet@alum.wpi.edu, lnavarrodelara@mgh.harvard.edu, tommi.raji@northwestern.edu,
nummenma@nmr.mgh.harvard.edu

Abstract:

Computational modeling of Transcranial Magnetic Stimulation (TMS) is a tradeoff between computational speed vs. spatial precision. In this study, we introduce a software toolkit for high-resolution TMS modeling, which may offer the best of both. The toolkit employs the recently developed boundary element fast multipole method (BEM-FMM) with accurate solution computations close to tissue interfaces. It operates within the MATLAB platform and is designed for a broad, medically-oriented computational research community. To enable easy, subject-specific operation, the package is compatible with human head models generated with automated tools such as SimNIBS and FreeSurfer.

Both coil design in free space as well as efficacy and focality of TMS for a specific subject could be modeled by the package, including optional user-defined parametric loops. Detailed and widely used coil models, generated in both CAD and wire formats, may include several hundred thousand elementary current elements and observation spaces with approximately 1 M field points; the corresponding computational times are on the order of 1 sec. Detailed head models with approximately 1 M triangular facets and a mesh resolution of 0.6 points per mm² are simulated in approximately 1.5 min which is arguably the fastest time to date. Further reduction of computational times is foreseen. The toolkit is augmented with a population of 16 ready-to-use head models for performing simulations for computational studies that do not involve MRI data collection. The toolkit is also augmented with a coil geometry generator capable of creating accurate coil models.

Keywords:

Transcranial Magnetic Stimulation, Boundary Element Method, Fast Multipole Method, High-Resolution Modeling

1. Introduction

For all three chief neurostimulation modalities – transcranial magnetic stimulation (TMS), transcranial electric stimulation (TES), and intracortical microstimulation (ICMS) – simulating the electric fields within a patient-specific head model is the major and often only way to foster spatial targeting and obtain a quantitative measure of the required stimulation dose (Bikson et al., 2018). Specifically, the net TMS E-field consists of primary and secondary components. The primary component – the E-field directly induced by the coil – can be determined using scalar and vector potentials, which are calculated based on the geometric features of the TMS coil. The secondary component – due to induced charges on tissue interfaces – can either be found using the boundary element method (BEM), the finite element method (FEM) or the finite difference method (FDM).

The above three methods differ in terms of computational speed and spatial precision. The boundary element method is fast with low-resolution models but its speed quickly deteriorates towards higher-resolution models. Salinas et al., 2007 and Salinas et al., 2009 have written two fast computer programs in C++ and interfaced them with MATLAB. The first program (3D coil generator) simulates a coil's geometry; the second program (E-field generator) performs numerical calculations of the induced E-field in the volume of interest. The Helsinki BEM MATLAB library (Stenroos et al., 2007) and its refinements (Stenroos & Sarvas, 2012; Stenroos & Nenonen, 2012; Stenroos, 2016; Stenroos & Nummenmaa, 2016; Koponen et al., 2017) have been used for the purpose of TMS modeling as well. Different mathematical BEM formulations can be employed for TMS modeling (Gomez et al., 2018a).

The chief finite element TMS solver is the well-known, open-source transcranial brain stimulation modeling software SimNIBS (Thielscher et al., 2015; Opitz et al., 2015; Nielsen et al., 2018), whose most recent version, v2.1 (Saturnino et al., 2018), currently uses the open-source FEM software getDP. There are also many general-purpose, open-source solvers for finite element modeling, from high-level environments such as getDP (Dular et al., 1988), Deal.II (Bangerth et al., 2007), and FEniCS (Logg et al., 2012), to lower-level environments such as PETSc (Balay et al., 2018). Those solvers provide a practical and well-tested choice for creating problem-specific software solutions.

Commercial FEM software has options for adaptive mesh refinement and for a variety of accurate CAD coil geometries but tends to be slow compared to SimNIBS. Several studies (Miranda et al., 2003; Deng et al., 2011; Cho et al., 2010; Makarov et al., 2016; Wang et al., 2018) have used the commercial FEM solver ANSYS Maxwell 3D. Silva et al., 2008 and Salvador et al., 2011 used the commercial FEM solver COMSOL. Deng et al., 2013 used the FEM package MagNet (Infolytica, Inc., Canada) for a comprehensive focality study of 50 different TMS coil designs.

Multigrid fast and efficient FDM TMS simulators have been developed by Laakso & Hirata, 2012 and Laakso et al., 2018. The commercial FDM software SEMCAD X was used for TMS modeling by Rastogi et al., 2017.

The problem with BEM, FEM, and FDM, is that they offer compromises between spatial accuracy and computational speed. Here, we use a new TMS modeling method, based on the coupling the boundary element method and the fast multipole method (BEM-FMM; Makarov et al., 2018; Htet et al., 2019), that offers both speed and accuracy at the same time. This approach is an integration of the BEM formulation in terms of surface charge density and FMM invented by Rokhlin and Greengard (Rokhlin, 1985; Greengard & Rokhlin, 1987; Gimbutas & Greengard, 2015). The algorithm still possesses the major advantage of the conventional BEM – high speed – but is simultaneously capable of processing a very large number of surface-based unknowns. The

BEM-FMM approach has demonstrated faster computational speed (10-1000 times faster) and superior accuracy for high-resolution head models as compared to both the standard boundary element method and the finite element method used in SimNIBS or commercial FEM software ANSYS Maxwell 3D (Htet et al., 2019). These advantages of the BEM-FMM approach have been recently independently and thoroughly confirmed (Gomez et al., 2018).

The present study introduces the open-source software package for high-resolution modeling TMS fields via BEM-FMM. The computational platform is basic MATLAB[®] starting with version 2018a and with major FMM subroutines and other critical numerical subroutines converted to MATLAB executables, for both Windows and Linux. The software package is augmented with a population of sixteen high-resolution 2 manifold CAD compatible head models (Htet et al., 2018) with the edge length of cortical surface meshes of 1.4 mm from the Human Connectome Project (Van Essen et al., 2012). It is also augmented with a flexible coil geometry generator and a base collection of ten different coil design examples. Appendix A contains a concise user's manual.

2. Materials and Methods

2.1 Numerical algorithm

Fast multipole method (FMM). We integrate and use the open-source FMM library FMMLIB3 (FMMLIB3, 2017; Gimbutas & Greengard, 2015). In this version, there is no *a priori* limit on the number of levels of the FMM tree, although after about thirty levels, there may be floating point issues (L. Greengard, private communication). The required number of levels is determined by the maximum permissible least-squares error or method tolerance, which is specified by the user. The FMM is a FORTRAN program compiled for MATLAB. The tolerance level `iprec` of the FMM algorithm is set at 0 or 1 (the relative least-squares error is guaranteed not to exceed 0.5% or 0.05%, respectively). This FMM version allows for a straightforward inclusion of a controlled number of analytical neighbor integrals on triangular patches to be precisely evaluated as specified below.

Boundary element method. We use the surface-charge approach mathematically called the adjoint double layer formulation (Rahmouni et al., 2018). Here, the unknown quantity is the charge density $\rho(\mathbf{r}')$ induced at conductivity boundaries. The corresponding integral equation is obtained by writing the total electric field $\mathbf{E}^t$ in a form that considers the primary solenoidal field $\mathbf{E}^p$ and a conservative contribution of the secondary induced surface charge density $\mathbf{E}^s$, i.e.

$$\mathbf{E}^t(\mathbf{r}) = \mathbf{E}^p(\mathbf{r}) + \mathbf{E}^s(\mathbf{r}), \quad \mathbf{E}^s(\mathbf{r}) = \int_S \frac{1}{4\pi\epsilon_0} \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|^3} \rho(\mathbf{r}') d\mathbf{r}' \quad (1)$$

where ϵ_0 is the permittivity of vacuum. Taking the limit of Eq. (1) as $\mathbf{r}$ approaches surface S from both sides (inside and outside) and using the continuity condition for the normal current component, $\sigma \mathbf{E}^t(\mathbf{r})$, one obtains the adjoint double-layer equation (Barnard et al., 1967; Makarov et al., 2016; Rahmouni et al., 2018) in the form

$$\frac{\rho(\mathbf{r})}{2} - \frac{\sigma_{in} - \sigma_{out}}{\sigma_{in} + \sigma_{out}} \mathbf{n}(\mathbf{r}) \cdot \int_S \frac{1}{4\pi} \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|^3} \rho(\mathbf{r}') d\mathbf{r}' = \frac{\sigma_{in} - \sigma_{out}}{\sigma_{in} + \sigma_{out}} \mathbf{n}(\mathbf{r}) \cdot \epsilon_0 \mathbf{E}^p(\mathbf{r}) \quad (2)$$

where $\mathbf{n}(\mathbf{r})$ is the outer normal vector to the particular boundary and $\sigma_{in/out}$ are the two distinct tissue conductivity values on either side of the boundary. The first (simpler) task of the numerical solution is to compute the excitation – the primary electric field $\mathbf{E}^p(\mathbf{r})$ – everywhere at tissue conductivity boundaries in Eq. (2). The second (more complicated) task is to compute the induced charge density by solving integral equation (2) of the second kind iteratively and thus compute the secondary electric field $\mathbf{E}^s$. Both fields add up.

Computation of primary electric and magnetic fields. Primary electric and magnetic fields $\mathbf{E}^p$ and $\mathbf{B}^p$ of a TMS coil are computed using FMM. The number of elementary current filaments is ultimately unlimited. Consider a small straight element of current $i_j(t)$ with segment vector $\mathbf{s}_j$ and center $\mathbf{p}_j$. Its magnetic vector potential, $\mathbf{A}^p$, at an observation point $\mathbf{c}_i$ is given by (Balanis, 2012)

$$\mathbf{A}^p(\mathbf{c}_i, t) = \frac{\mu_0}{4\pi} \frac{i_j(t)\mathbf{s}_j}{|\mathbf{c}_i - \mathbf{p}_j|} \quad (3)$$

where μ_0 is magnetic permeability of vacuum and index p again means the primary field. The electric field generated by this current element is

$$\mathbf{E}^p = -\frac{\partial \mathbf{A}^p}{\partial t} \quad (4)$$

Assuming harmonic excitation with angular frequency ω and omitting the redundant phase factor of $-j$, one has

$$\mathbf{E}^p = \frac{\mu_0 \omega}{4\pi} \frac{i_{0j}\mathbf{s}_j}{|\mathbf{c}_i - \mathbf{p}_j|} \quad (5)$$

Now consider multiple straight short current segments with moments $i_{j0}\mathbf{s}_j$ forming the coil. Also consider multiple observation points $\mathbf{c}_i$ which coincide, for example, with triangle centroids of the head surface mesh. For every observation point, the electric field in Eq. (5) is computed via FMM as a potential of a single layer repeated three times, i.e., separately for each component of the field. The corresponding pseudo charges used in FMM function `lfmm3dpart` of the FMM library (FMMLIB3, 2017; Gimbutas & Greengard, 2015) are chosen in the form (to within a constant)

$$q_{xj} = i_{0j}s_{xj}, \quad q_{yj} = i_{0j}s_{yj}, \quad q_{zj} = i_{0j}s_{zj} \quad (6)$$

The same element of current generates a magnetic field $\mathbf{B}^p$ given by Biot-Savart law

$$\mathbf{B}^p = \frac{\mu_0}{4\pi} \frac{i_{0j}\mathbf{s}_j \times (\mathbf{c}_i - \mathbf{p}_j)}{|\mathbf{c}_i - \mathbf{p}_j|^3} \quad (7)$$

For multiple segments and multiple observation points, Eq. (7) cannot be evaluated using the FMM framework directly. However, it can be rewritten in the form

$$\mathbf{B}^p = \frac{\mu_0}{4\pi} \frac{i_{0j}}{|\mathbf{c}_i - \mathbf{p}_j|^3} \begin{pmatrix} \mathbf{s}_j^x \cdot (\mathbf{c}_i - \mathbf{p}_j) \\ \mathbf{s}_j^y \cdot (\mathbf{c}_i - \mathbf{p}_j) \\ \mathbf{s}_j^z \cdot (\mathbf{c}_i - \mathbf{p}_j) \end{pmatrix} \quad (8)$$

$$\mathbf{s}_j^x = \begin{pmatrix} 0 \\ -s_{zj} \\ +s_{yj} \end{pmatrix}, \quad \mathbf{s}_j^y = \begin{pmatrix} +s_{zj} \\ 0 \\ -s_{xj} \end{pmatrix}, \quad \mathbf{s}_j^z = \begin{pmatrix} -s_{yj} \\ +s_{xj} \\ 0 \end{pmatrix} \quad (9)$$

which is equivalent to the electric potential of a double layer (layer of electric dipoles). This potential is to be computed three times and with three different sets of pseudo dipole moments given by Eq. (9) and used in the FMM function `lfmm3dpart` (FMMLIB3, 2017; Gimbutas & Greengard, 2015). This is another standard FMM task.

Computation of secondary electric field. The fast multipole method (Rokhlin, 1985; Greengard & Rokhlin, 1987) speeds up computation of a matrix-vector product by many orders of magnitude. In the present problem, such a matrix-vector product appears when an electric field from many point sources $\rho(\mathbf{r}')$ in space has to be computed at many observation points $\mathbf{r}$. Namely, discretization of the surface integral in Eq. (1) or Eq. (2) and use of the zeroth-order piecewise constant basis functions (pulse bases) yields

$$\mathbf{E}^s(\mathbf{c}_i) = \sum_{j=1}^N \frac{A_j \rho_j}{4\pi\epsilon_0} \frac{\mathbf{c}_i - \mathbf{c}_j}{|\mathbf{c}_i - \mathbf{c}_j|^3} \quad (10)$$

where $A_i, \mathbf{c}_i, i = 1, \dots, N$ are the areas and centers of triangular surface facets t_i ; ρ_i are triangle surface charge densities in C/m², and $A_j \rho_j$ are elementary charges located at $\mathbf{c}_j$. Eq. (10) is straightforwardly computed via the same FMM library, as an electric field of a given charge distribution. Such a computation is done at every step of an iterative solution for integral equation (2).

Approximation (10) is inaccurate for the neighbor facets. Using the Petrov-Galerkin method with the same pulse bases as testing functions, we employ a refined computation:

$$\mathbf{E}^s(\mathbf{c}_i) = \sum_{j \notin V(i)} \frac{A_j \rho_j}{4\pi\epsilon_0} \frac{\mathbf{c}_i - \mathbf{c}_j}{|\mathbf{c}_i - \mathbf{c}_j|^3} + \frac{1}{A_i} \sum_{j \in V(i)} \rho_j \iint_{t_{ij}} \frac{(\mathbf{r} - \mathbf{r}')}{4\pi\epsilon_0 |\mathbf{r} - \mathbf{r}'|^3} d\mathbf{r}' d\mathbf{r} \quad (11)$$

where $V(i)$ encompasses a neighborhood of facets around observation triangle t_i . The inner integrals in Eq. (11) are computed analytically (Wilton et al., 1984; Wang et al., 2003; Makarov et al., 2016) while the outer integrals use a Gaussian quadrature of 10th degree of accuracy (Cools, 2003). $V(i)$ includes $K \ll N$ nearest triangular facets. Inclusion of a small number of precomputed potential neighbor integrals (three to sixteen) improves the convergence of the iterative solution; inclusion of a larger number has a negligible effect. The present BEM-FMM approach performs precise analytical integration over $K = 12 - 16$ closest neighbor facets.

The same approach is used for volume field computations, when an observation point is located close to the charged boundary. In this case, the contribution of charges on neighbor facets is accurately averaged as prescribed by the second term on the right-hand side of Eq. (11).

Iterative solution. Eq. (2) is solved iteratively using the generalized minimum residual method or GMRES (Barrett et al., 1994; Saad, 2003) implemented by Drs. P. Quillen and Z. Hoffnung of MathWorks, Inc. Its overall performance and convergence are excellent, especially for complicated head geometries.

Algorithm accuracy and speed. The accuracy of the algorithm has been tested using side-by-side comparisons with analytical solutions, the open-source FEM solver getDP employed by SimNIBS, and the commercial FEM software ANSYS Maxwell with adaptive mesh refinement (Makarov et al., 2018; Htet et al., 2019). Both spherical and anatomically realistic head models were tested. For high-resolution models, the BEM-FMM algorithm ran significantly (10-1000 times) faster than the FEM and provided a better solution accuracy (Htet et al., 2019).

For the head models described below with approximately 0.9 M facets and with an average edge length of 1.4 mm in the cortex, run times necessary to achieve a relative residual of 10⁻³ and obtain a good solution accuracy are about 90 sec (4.5 sec per iteration for 20 iterations or less). These data were compiled on an Intel Xeon E5-2698 v4 CPU (2.20 GHz) server, 256 Gb RAM, MATLAB 2018a. Lowering the relative residual does not improve accuracy.

2.2 Head models

Model format. Any head model comprised of *.stl (stereolithography) files for each individual brain compartment could be imported. Such models are routinely generated by SimNIBS (Saturnino et al., 2018) and FreeSurfer (Fischl, 2012). For example, any of the 50 head models from the Population Head Model Repository created by Lee et al., 2016 (see also Lee et al., 2018; The Population Head Model Repository, 2017) using SimNIBS software and the Connectome MRI Database may be used. In this case, the typical model has about 0.7 M triangular facets in total (skin resolution was lowered via post-processing).

Built in head models. The toolkit is augmented with a collection of sixteen high-resolution, 2-manifold CAD compatible head models within the MATLAB platform described in (Htet et al., 2018). Each model contains skin, skull, CSF (cerebrospinal fluid), GM (gray matter), cerebellum, WM (white matter), and ventricles head compartments in the form of triangulated manifold shells with nearly uniform triangle density. Every surface object can be 3D printed. The models are compatible with a variety of modeling software, including ANSYS Electronics Desktop, CST Studio Suite, Sim4Life/SEMCAD, and other electromagnetic simulation software packages.

The collection is based on MRI data for sixteen subjects from the Human Connectome Project (Van Essen et al., 2012) segmented using the SimNIBS processing pipeline (Saturnino et al., 2018). The MRI data were kindly provided by the Human Connectome Project WU-Minn Consortium (PIs: David Van Essen and Kamil Ugurbil; 1U54MH091657, see S1200 Reference Manual 2018). The average number of triangular surface facets in a model is 866,000, the average triangle quality is 0.25, the average edge length for all surfaces is 1.48 mm, and the average surface mesh density or resolution is 0.57 points per mm². This mesh resolution (1 to 1.5 mm) corresponds approximately to the high-quality 3 T images.

Head model visualization and pre-processing. The head models are visualized in MATLAB either as a surface mesh for every brain compartment (Fig. 1a) or as head cross-sections (2-manifold polygons for skin, scalp, and CSF) in any principal plane – coronal (XZ), sagittal (YZ), and transverse (XY) – located anywhere in space. As an example, Fig. 1a shows white matter, gray matter, and CSF (inner skull) shells for Connectome subject # 110411. Next, Fig. 1b shows three cross-section cuts obtained for the same subject at $z = -0.0185\text{ m}$, $y = 0$, $x = 0.01\text{ m}$. Furthermore, the integral mesh characteristics, including mesh quality, can be computed.

For every head model we define a cortical layer as a volume within the GM shell but excluding WM volume. A dense point cloud within the cortical layer is additionally generated and used to estimate normal and tangential fields specifically in the cortical layer.

Precomputed global BEM-FMM parameters for the head model. The BEM-FMM algorithm needs to accurately compute electrostatic double surface field integrals for neighbor triangular facets t_i, t_j in the form:

$$\iint_{t_i t_j} \frac{(\mathbf{r} - \mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|^3} d\mathbf{r}' d\mathbf{r} \quad (12)$$

For each head model, these integrals are precomputed and saved. Their number could potentially be large. The safe default value is given by twelve neighbor integrals for every triangle t_i excluding the self-integral, which is exactly zero. This value could be lowered to four if the corresponding file size appears too large. The typical file sizes to save the integrals are approximately 250 Mbytes.

Fig. 1a. White matter, gray matter, and CSF (inner skull) shells for Connectome subject # 110411.

Fig. 1b. Three cross-section cuts obtained for subject # 110411 at $z = -0.0185 \text{ m}$, $y = 0$, $x = 0.01 \text{ m}$. Intracranial volume is marked yellow; skull volume is marked orange; and scalp volume is marked red.

2.3 Coil models

Coil CAD model. The base of a coil model is the conductor (or wire) centerline. The centerline of a conductor (or a multiple thereof) is created in MATLAB as set of 3D nodes distributed along a curve (circle, ellipse, spiral, etc.). After the centerline has been created, the CAD model for the coil conductor(s) is constructed by

- i. Automatic creation of an unstructured triangular mesh for conductor's cross-section similar to that shown in Fig. 2a,b. This mesh could have an arbitrary density and is either rectangular or elliptical in shape;
- ii. Sweeping this cross-section mesh along curve's centerline and creating a structured triangular surface mesh for the conductor's side surface. The cross-section remains perpendicular to conductor's centerline;
- iii. Creating start/end caps for non-closed conductors and;
- iv. Joining the meshes if necessary and eliminating duplicated nodes.

Fig. 3 shows simplified coil models created in this way. Fig. 3a is a simplified MRi-B91 TMS-MRI coil model (MagVenture, Denmark) with elliptical conductors of a rectangular cross-section; Fig. 3b is a simplified MagPro C-B60 coil model (MagVenture, Denmark); Fig. 3c is a double shape-eight spiral coil with an elliptical cross-section and two bootstrapped interconnections; Fig. 3d is a simplified Cool-40 Rat small animal coil model (MagVenture, Denmark); Fig. 3e is a three-axis multichannel TMS coil array radiator (Navarro de Lara et al, 2018).

Once created in MATLAB, the CAD coil model may be exported to any other computational package in STL format. Note that coil models created in this way do not yet define the normal vectors of surface triangles. Therefore, after converting to STL format, a mesh healing operation is required; ANSYS Electronics Desktop and other packages implement this step.

Fig. 2. Filaments of current (red) inside conductor's volume and conductor's surface CAD model following cross-section triangulation. a) Modeling uniform current distribution through a conductor's cross-section (Litz wire); b) Modeling the skin effect (a solid copper conductor at a higher frequency).

Computational conductor model. The volumetric current flow is modeled as a set of infinity-thin current filaments or segments. The current filaments (shown red in Fig. 2a,b) are defined as short straight lines joining centroids of triangles of the cross-sectional mesh, which is replicated along the conductor's centerline as many times as required. The cross-section is always perpendicular to the conductor's centerline; the filaments are therefore parallel to the centerline of the conductor. The underlying computational wire grid is located strictly inside the solid conductor as illustrated in Fig. 2a,b.

The number of filaments passing through conductor's cross-section can be made arbitrarily large. When a time-varying electric current flows in a volumetric conductor, two cases are possible: a) Litz-wire conductor with uniform current flow through its cross-section or b) current flow in a thin skin layer close to the surface. In the first case, the current filaments are uniformly distributed over the conductor's cross-section as shown in Fig. 2a. In the second case, the current filaments are distributed close to its surface as shown in Fig 2b. Each filament carries current proportional to the area of the base triangle; the total conductor current is 1 A. Once the current filaments are internally defined, the primary field computations follow Eqs. (5), (8), and (9).

Fig. 3. Some solid CAD models created using the MATLAB-based coil geometry generator. Fig. 3a is a simplified MRi-B91 TMS-MRI coil model (MagVenture, Denmark) with elliptical conductors of a rectangular cross-section; Fig. 3b is a simplified MagPro C-B60 coil model (MagVenture, Denmark); Fig. 3c is a generic double shape-eight spiral coil model with an elliptical cross-section and two bootstrapped interconnections; Fig. 3d is a simplified Cool-40 Rat small animal coil model (MagVenture, Denmark); Fig. 3e is a three-axis multichannel TMS coil array radiator (Navarro de Lara et al, 2018).

2.4 Software organization

Fig. 4 shows the software organization chart of the toolkit along with the major computation blocks (1 through 6). The blocks using the FMM computational algorithm, in some way or another, are marked grey. These program blocks do not have to be executed sequentially. They are described in more detail in Appendix A along with the program flow.

Fig. 4. Software organization chart. Major computational blocks are labeled 1 through 6.

3. Results

3.1 FMM-based coil modeling in free space

The toolkit includes editable scripts for ten pre-constructed coil models, some of which are shown in Fig. 3. Other coil models or coil arrays can be constructed in a similar way, by defining the conductor centerline(s) and then automated cross-section sweeping. Of special note is the three-axis coil in Fig. 3e, which is in fact an array of three independent radiators with three independent excitations.

Coil modeling is done by computing and plotting respective electric and magnetic fields along a line or in a plane. Operating frequency (CW excitation is assumed) and conductor current must be specified. Fig. 5 shows typical output field results. Table 1 (Makarov et al., 2019) reports timing results for $\mathbf{E}^p$ and $\mathbf{B}^p$ computations. Here and below in the text, computations are performed using an Intel Xeon E5-2698 v4 CPU 2.2 GHz server, Windows Server 2016 operating system, and MATLAB 2018a.

As an example, a conical-shape coil with 50 single coaxial loops of a circular cross-section has been considered. Each of the loops is divided into 100 straight filaments. We assume a Litz-wire conductor and introduce 30 interpolation points nearly uniformly distributed over conductor's cross-section. This results in a coil model that contains 150,000 elementary current dipoles in total.

We now keep this coil geometry fixed but vary the number of observation points in a coronal plane whose size is twice the coil size as per Tables 1 and 2, respectively. In these tables, we present mesh generation time for the wire grid, FMM execution times, and execution times for a simple yet vectorized MATLAB code that directly computes Eq. (5) and Eq. (7) for the required set of observation points. All results have been averaged over several runs.

By analyzing the representative FMM computations and the underlying wire model generation times from Tables 1 and 2, we observe that coil evaluation can generally be performed in less than 1 sec on an ordinary server. On the other hand, the direct vectorized MATLAB loop would run 200-6,000 times slower.

Fig. 5. a) Observation plane geometry for MRi-B91 TMS-MRI coil model. b, c) Electric and magnetic fields in the plane with 0.25 M observation points. Conductor cross-section is marked in red. Conductor current is 5 kA and CW frequency is 3 kHz.

We also observe that the FMM code indicates only a very modest increase in time for the reported size of the coil model and the sizes of the observation domain. No specific effort to parallelize the algorithm has yet been made. However, MATLAB automatically performs multithreading pertinent to linear algebra operations available in LAPACK (Linear Algebra PACKage) and some level-3 BLAS (Basic Linear Algebra Subprograms) matrix operations, allowing them to execute faster on multicore-enabled machines.

With regard to coil optimization, parametric loops versus coil geometry parameters could straightforwardly be organized and executed relatively fast. Reference Makarov et al., 2019

discusses corresponding examples. Additionally, an example pertinent to steering in a multi-channel coil array is also discussed.

Table 1. Timing results for E^p computations. An accurate coil model with 150,000 elementary current dipoles is considered.

Observation points	Wire grid generation time, sec	FMM execution time, sec	Vectorized MATLAB loop, sec
100×100	0.32	0.44	92
200×200	0.32	0.47	394
300×300	0.32	0.57	892
500×500	0.32	0.93	2503

Table 2. Timing results for B^p computations. An accurate coil model with 150,000 elementary current dipoles is considered.

Observation points	Wire grid generation time, sec	FMM execution time, sec	Vectorized MATLAB loop, sec
100×100	0.31	0.49	242
200×200	0.31	0.52	968
300×300	0.31	0.62	2250
500×500	0.31	0.99	6320

3.2 Electric field computations within the head

Computational times. For head models with ~0.9 M facets described above, the main iterative algorithm runs in approximately 90 sec (4.5 sec per iteration for 20 iterations). Restoration of surface electric fields (just inside or outside a conducting boundary) requires one step of the iterative solution, i.e. 4-5 sec. Restoration of line fields is also fast. However, detailed field computations with a fine resolution (0.3-0.4 mm) in a plane crossing many conductivity boundaries as well as field computations in a cortical volume require 30-40 sec. This is because the contributions of surface charges on nearby triangles need to be accurately averaged for every observation point in question, similar to Eq. (11). The run time data are for an Intel Xeon E5-2698 v4 CPU (2.20 GHz) server, 256 Gb RAM, MATLAB 2018a.

Computational example. In the following computation example, we will consider Connectome head model # 110411 and MRI-B91 TMS-MRI coil model targeting the area of the primary motor cortex and located above the precentral gyrus of the right hemisphere. The tissue conductivity values are those of Engwer et al., 2017 and Piastra et al., 2018. Conductor current is 5 kA and CW frequency is 3 kHz. Fig. 6a shows the model, the coil position above the skin surface, the coil centerline (red) which starts at the coil bottom, as well as positions and sizes of three principal observation planes. All three planes pass through the point which is an intersection of the gray matter shell with the coil centerline.

Fields along a line. Fig. 6b shows three Cartesian components of the electric field computed along the coil centerline. Any other line may be considered as well. The prominent field discontinuity at tissue interfaces is due to surface charges residing at the conductor boundaries. This discontinuity is precisely tracked with our computational method. Comparison of line fields with the commercial FEM software ANSYS Maxwell, which uses adaptive mesh refinement, has been performed in (Htet et al., 2019). There, ten head models from the Population Head Model Repository (Lee et al., 2016; Lee et al., 2018; The Population Head Model Repository, 2017) have been tested and a very good agreement between the two approaches was established.

Fig.6. a) – Computation geometry, position of the observation line (coil centerline), and positions of three observation planes. b) Electric field distribution along the coil centerline.

Surface fields just inside/outside brain compartments (conductivity boundaries). For any model shell (skin, skull, CSF, GM, WM, etc.) the fields and surface charge distribution can be extracted. The magnetic field is simply the continuous coil's field in free space, $\mathbf{B}^t(\mathbf{r}) = \mathbf{B}^p(\mathbf{r})$, since the secondary field of eddy currents is ignored in TMS studies. The total electric field, however, is discontinuous at interfaces. When approaching an interface from outside/inside it is given by

$$\mathbf{E}_{just\frac{outside}{inside}}^t(\mathbf{r}) = \mathbf{E}^p(\mathbf{r}) - \int_S \frac{1}{4\pi\epsilon_0} \frac{\mathbf{r}' - \mathbf{r}}{|\mathbf{r} - \mathbf{r}'|^3} \rho(\mathbf{r}') d\mathbf{r}' \pm \mathbf{n} \frac{\rho(\mathbf{r})}{2\epsilon_0}, \mathbf{r} \in S \quad (13)$$

where the notations have been explained in Section 2.1. Since the charge distribution is already known from the BEM-FMM solution, the integral on the right-hand side of Eq. (13) is computed using the FMM (with accurate neighbor integral contributions). Then, the electric field just inside or outside the interface is obtained by simply choosing the proper sign in front of the last term on the right-hand side of Eq. (13).

Figs. 7a-c show surface electric fields just inside the gray matter shell, i.e. at the top of the cortical layer. In Fig. 7a, the magnitude of total electric field is shown. Fig. 7b presents the magnitude of the continuous field component tangential to the surface and Fig. 7c is the most important discontinuous signed normal electric field. The color scale expands from 0 to 160 V/m, 0 to 120 V/m, and -150 to 150 V/m, respectively.

Fig. 7. Surface electric fields just inside the gray matter shell: the top level of the cortical layer. a) Magnitude of total electric field (0-160 V/m). b) Magnitude of tangential electric field (0-120 V/m). c) Signed normal electric field component (-150 to 150 V/m) in the direction of the inner normal vector, i.e. into the gray matter volume. d) Induced surface electric charge density on the gray matter surface; the color scale expands from -1 to $+1 \times 10^{-9}$ C/m².

Total normal and tangential fields given in Fig. 7b,ca are defined in the form:

$$\mathbf{E}_{\perp}^t(\mathbf{r}) = -\mathbf{n} \cdot \mathbf{E}^t(\mathbf{r}); \mathbf{E}_{\parallel}^t(\mathbf{r}) = \mathbf{E}^t(\mathbf{r}) + \mathbf{n} \cdot \mathbf{E}_{\perp}^t(\mathbf{r}) \quad (14)$$

i.e. the signed normal field $\mathbf{E}_{\perp}^t$ is considered positive if it is directed into the cortical layer. The total and tangential fields in Fig. 7a,b indicate rather familiar behavior referenced in many literature sources. The normal field in Fig. 7c is more irregular. It is mostly concentrated inside certain sulci, where the dominant coil field component becomes perpendicular to the cortical surface. A sulcus that is perpendicular to the dominant field component thus becomes “polarized”.

Quite remarkably, the inner normal field component in Fig. 7c almost perfectly correlates with the underlying induced surface charge density distribution on the gray matter surface shown in Fig 7d. There, the color scale expands from -1 to $+1 \times 10^{-9} \text{ C/m}^2$. The correlation measure is the relative error, ΔE , given by

$$\Delta E = \left\| \mathbf{E}_{\perp}^t - \frac{\rho(\mathbf{r})}{\epsilon_0} \right\| / \|\mathbf{E}_{\perp}^t\| \quad (15)$$

The error value (15) for the present example for the entire gray matter shell is 5.3%. This error value decreases further when only the area under the coil is considered. Such an observation might be significant for real-time TMS field computations since computing and storing the charge distributions is far more efficient than computing and storing the field distributions. This question, however, is beyond the scope of the present study.

Fields in principal planes. Those computations require more time since the potential integrals are no longer precomputed and must be calculated in real time. Fig. 8a illustrates the electric field distribution for the y-component of the total electric field (the component that is directed out of figure plane) in the coronal plane from Fig. 6 with a grid resolution of $0.4 \times 0.3 \text{ mm}$.

Fig. 8. Y-component of the total electric field (directed out of figure plane) in the coronal plane from Fig. 6 with a grid resolution of $0.4 \times 0.3 \text{ mm}$ (0.25 M observation points). The color scale expands from 0 to 105 V/m. a) Field in the entire plane. b) Field in a zoomed in cortical area with the size of $40 \times 40 \text{ mm}$.

The y-component is simultaneously the dominant primary field component of the coil. Only the field within brain compartments excluding scalp and skull is displayed in Fig. 8. The color scale expands for 0 to 105 V/m.

By observing Fig. 8, it might appear at first sight that the field component of interest is well localized just underneath the coil. However, this observation is only valid for this particular plane and for this particular (primarily tangential to the cortex) field component.

Volumetric fields within the cortical volume. We define the cortical layer as a volume within the GM shell but excluding the WM volume. A point cloud within the cortical layer is automatically generated and used to estimate normal and tangential fields specifically in the cortical layer. This point cloud contains at least five inner observation points across the layer, which are uniformly spaced along the normal direction of every triangle of the GM shell. Both normal and tangential field components are evaluated at every such observation point and respective “hot spots” within the cortical layer are identified and labeled with a small prominent blue ball.

Fig. 9a shows examples of hot spots where the magnitude of the total normal field component is greater than or equal to 90% of the maximum total normal field in the cortex; Fig. 9b shows a similar result but for the 80% threshold. The underlying white matter shell is also shown. One observes that the hot spots for the peak fields may be very well localized in space as long as the required field is 80% or higher of the maximum normal field (for this case the maximum normal field is approximately 140 V/m). In this sense, the focality of the coil could be well established for a specific position/subject. However, these hot spots may occur not directly beneath the coil as expected, but at a rather significant distance from the coil centerline as seen in Fig. 9a,b. This observation underscores the value of numerical modeling.

4. Discussion and Conclusions

In the present study, we have described the performance and capabilities of a software toolkit for high-resolution TMS modeling. The toolkit employs the boundary element fast multipole method with accurate field evaluation close to conducting interfaces. It operates entirely within the MATLAB platform and is designed for a broad, medically-oriented audience without specialized programming skills. To enable fast, subject-specific modeling, we have constructed a simulation capability compatible with automated human head CAD model generation tools such as SimNIBS or FreeSurfer. The toolkit is augmented with the collection of 16 head models and with a coil geometry generator script.

Both coil design in free space as well as efficacy and focality of TMS for a specific subject could be modeled by the package, including optional user-defined MATLAB parametric loops. Detailed coil models, presented in both CAD and wire formats, may include several hundred thousand elementary current elements and observation spaces with approximately 1 M field points; the corresponding computational times are on the order of 1 sec. Detailed head models with approximately 1 M triangular facets and a mesh resolution of 0.6 points per mm² are simulated in approximately 1.5 min. Further reduction of the computational time is foreseen by using real-valued FMM arithmetic.

The fields within the head are examined along a line, just inside or outside a brain compartment surface, in three principal planes, or in the entire cortical volume. Volumetric field computations with a large number of observation points close to charged boundaries may require up to 1 min.

Although the accuracy of the present numerical approach has already been justified, an additional error test may be needed in a specific case. To do so, a refined set of head models has been included into the toolkit with 3.5 M triangular facets on average and with a mesh resolution of 1.2 points

per mm^2 . These were obtained after barycentric oversampling (4×1) and subsequent surface-preserving Laplacian smoothing (Vollmer et al., 1999) of the base models. Those models may be characterized by a better solution convergence; they run in approximately 200-250 sec.

Fig. 9. Hot spots – observation points within the cortical volume where the magnitude of the total normal field component is greater than or equal to 90% (a) or 80% (b) of the maximum total normal field in the cortex. The underlying white matter shell is shown.

In conclusion, we anticipate that the described TMS modeling toolkit will be a useful academic research tool. The authors would welcome user feedback and look forward to further improving software performance and capabilities.

Acknowledgements

The authors wish to thank Dr. Leslie Greengard of the Courant Institute of Mathematical Sciences, New York, NY for multiple insightful discussions and support in our utilization of the FMM library. This work has been partially supported by the National Institutes of Health under award numbers R43AR071220, R00EB015445, NINDS R44NS090894, and R01MH111829.

Code availability:

The code will be made available through

- i. MATLAB Central. 2019 Onl.: <https://www.mathworks.com/matlabcentral/> and;
- ii. Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital. 2019 Onl.: <https://www.nmr.mgh.harvard.edu/research/software>

These links are not yet active. To obtain the toolkit code, please send a short note to the first author (makarov@wpi.edu).

References

- [1] Balanis CA. *Advanced Engineering Electromagnetics*, Wiley, New York, 2012. 2nd Ed., ISBN-10: 0470589485.
- [2] Balay S, Abhyankar S, Adams MF, Brown J, Brune P, Buschelman K, Dalcin L, Eijkhout V, Gropp WD, Kaushik D, Knepley MG, May DA, McInnes LC, Mills RT, Munson T, Rupp K, Sanan P, Smith BF, Zampini S, Zhang H, *PETSc Users Manual*. Argonne National Laboratory. 2018. ANL-95/11 – Rev. 3.9. Onl: <http://www.mcs.anl.gov/petsc>.
- [3] Bangerth W, Hartmann R, Kanschat, G. Deal.II—A general-purpose object-oriented finite element library. *ACM Transactions on Mathematical Software (TOMS)*. 2007 Aug;33(4). Article #24. doi: 10.1145/1268776.1268779.
- [4] Barnard ACL, Duck IM, Lynn MS. The application of electromagnetic theory to electrocardiology: I. Derivation of the integral equations. *Biophys J.*, 2007;7(5):443-62. doi: 10.1016/S0006-3495(67)86598-6.
- [5] Barrett R, Berry M, Chan TF, Demmel J, Donato JM, Dongarra J, Eijkhout V, Pozo R, Romine C, Van der Vorst H. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, 2nd Ed., Philadelphia, USA, Society for Industrial and Applied Mathematics. 1994. Onl: <http://www.netlib.org/templates/templates.pdf>.
- [6] Bikson M, Brunoni AR, Charvet LE, Clark VP, Cohen LG, Deng ZD, Dmochowski J, Edward DJ, Frohlich F, Kappenman ES, Lim KO, Loo C, Mantovani A, McMullen DP, Parra LC, Pearson M, Richardson JD, Rumsey JM, Sehatpour P, Sommers D, Unal G, Wassermann EM, Woods AJ, Lisanby SH. Rigor and reproducibility in research with transcranial electrical stimulation: An NIMH-sponsored workshop. *Brain Stimul.* 2018 May - Jun;11(3):465-480. doi: 10.1016/j.brs.2017.12.008. Epub 2017 Dec 29.
- [7] Blumberger DM, Vila-Rodriguez F, Thorpe KE, Feffer K, Noda Y, Giacobbe P, Knyahnytska Y, Kennedy SH, Lam RW, Daskalakis ZJ, Downar J. Effectiveness of theta burst versus high-frequency repetitive transcranial magnetic stimulation in patients with depression (THREE-D): a randomised non-inferiority trial. *Lancet*, 2018;391(10131):1683-1692. doi: 10.1016/S0140-6736(18)30295-2.

- [8] Collection of Sixteen High-Quality Human Head CAD Models. 2019. MATLAB Central Onl: <https://www.mathworks.com/matlabcentral/fileexchange/69517-collection-of-sixteen-high-quality-human-head-cad-models>
- [9] Cools R. An Encyclopaedia of Cubature Formulas. *J. Complexity*, 2003. 19:445-453, Retrieved from: <http://www.cs.kuleuven.ac.be/~nines/research/ecf/ecf.html>.
- [10] Cho YS, Suh HS, Lee WH, Kim TS. TMS modeling toolbox for realistic simulation. *Conf Proc IEEE Eng Med Biol Soc*. 2010;2010:3113-6. doi: 10.1109/IEMBS.2010.5626096.
- [11] de Munck JC. A linear discretization of the volume conductor boundary integral equation using analytically integrated elements. *IEEE Trans Biomed Eng*. 1992;39(9):986-90. doi: 10.1109/10.256433.
- [12] Deng ZD, Lisanby SH, Peterchev AV. Electric field strength and focality in electroconvulsive therapy and magnetic seizure therapy: a finite element simulation study. *J. of Neural Engineering*. 2011 Feb;8(1): 1-13. doi: 10.1088/1741-2560/8/1/016007.
- [13] Deng ZD, Lisanby SH, Peterchev AV. Electric field depth-focality tradeoff in transcranial magnetic stimulation: simulation comparison of 50 coil designs. *Brain Stimulation*. 2013 Jan;6(1):1-13. doi: 10.1016/j.brs.2012.02.005.
- [14] *Development of Neuromodulation*. International Neuromodulation Society. 2018. Onl: <https://www.neuromodulation.com/about-neuromodulation>.
- [15] Dular P, Geuzaine C, Henrotte F, Legros W. A general environment for the treatment of discrete problems and its application to the finite element method. *IEEE Transactions on Magnetics*. 1988 Sep;34(5):3395–3398. doi: 10.1109/20.717799.
- [16] Dune FEM Software. Onl: <https://www.dune-project.org/>; www.duneuro.org.
- [17] Engwer C, Vorwerk J, Ludewig J, Wolters CH. A Discontinuous Galerkin Method to Solve the EEG Forward Problem Using the Subtraction Approach. *SIAM J. Sci. Comput*. 2017;39(1):B138–B164. doi: 10.1137/15M1048392.
- [18] Ferguson AS, Zhang X, Stroink G. A Complete Linear Discretization for Calculating the Magnetic Field Using the Boundary Element Method. *IEEE Trans Biomed Eng*. 1994;41(5):455-60. doi: 10.1109/10.293220.
- [19] *Fast Multipole Method (FMM) library in R³*. V. 1.2.1. Nov. 9th 2017. Online: <https://github.com/zgimbutas/fmmlib3d>
- [20] Fischl, B. FreeSurfer. *Neuroimage*. Aug 2012;62(2): 774-781. doi: 10.1016/j.neuroimage.2012.01.021. Onl.: <http://surfer.nmr.mgh.harvard.edu/>.
- [21] Gimbutas Z, Greengard L. Simple FMM Libraries for Electrostatics, Slow Viscous Flow, and Frequency-Domain Wave Propagation. *Communications in Comput. Phys*. 2015;18(2):516-528. Retrieved from: <https://dx.doi.org/10.4208/cicp.150215.260615sw>.
- [22] Gomez LJ, Yucel AC, Michielssen E. The ICVSIE: A General Purpose Integral Equation Method for Bio-Electromagnetic Analysis. *IEEE Trans Biomed Eng*. 2018a Mar;65(3):565-574. doi: 10.1109/TBME.2017.2704540.
- [23] Gomez L, Dannhauer M, Koponen L, & Peterchev A.V. Conditions for numerically accurate TMS electric field simulation. *bioRxiv 505412* 2018b. doi: <https://doi.org/10.1101/505412>.
- [24] Greengard L, Rokhlin V. A fast algorithm for particle simulations. *J. Comput. Phys*. 1987;73(2):325-348. doi: 10.1016/0021-9991(87)90140-9.
- [25] Hasgall PA, Di Gennaro F, Baumgartner C, Neufeld E, Lloyd B, Gosselin MC, Payne D, Klingenberg A, Kuster N. *IT'IS Database for thermal and electromagnetic parameters of*

- biological tissues*. Version 4.0, May 15, 2018. doi: 10.13099/VIP21000-04-0. Onl: www.itis.ethz.ch/database.
- [26] Htet AT, Noetscher G, Burnham EH, Nummenmaa A, Makarov SN. Collection of Sixteen High-Quality Human Head CAD Models Generated with SimNIBS 2.1 Using Connectome Subject Data within MATLAB® Platform. *bioRxiv*. 2018 doi: <https://doi.org/10.1101/480103>.
 - [27] Htet AT, Saturnino GB, Burnham EH, Noetscher G, Nummenmaa A, Makarov SN. Comparative performance of the finite element method and the boundary element fast multipole method for problems mimicking transcranial magnetic stimulation (TMS). *J Neural Eng*. 2019;16:1-13. doi: <https://dx.doi.org/10.1088/1741-2552/aafbb9>.
 - [28] Human Connectome Project. *S1200 Reference Manual*. April 10 2018. Onl: https://www.humanconnectome.org/storage/app/media/documentation/s1200/HCP_S1200_Release_Reference_Manual.pdf
 - [29] Koponen LM, Nieminen JO, Mutanen TP, Stenroos M, Ilmoniemi RJ. Coil optimisation for transcranial magnetic stimulation in realistic head geometry. *Brain Stimulation*. 2017 Jul - Aug;10(4):795-805. doi: 10.1016/j.brs.2017.04.001.
 - [30] Laakso I, Hirata A. Fast multigrid-based computation of the induced electric field for transcranial magnetic stimulation. *Phys Med Biol*. 2012 Dec 7;57(23):7753-65. doi: 10.1088/0031-9155/57/23/7753.
 - [31] Laakso I, Murakami T, Hirata A, Ugawa Y. Where and what TMS activates: Experiments and modeling. *Brain Stimulation*. 2018 Jan - Feb;11(1):166-174. doi: 10.1016/j.brs.2017.09.011.
 - [32] Lee E, Duffy W, Hadimani R, Waris M, Siddiqui W, Islam F, Rajamani M, Nathan R, Jiles D. Investigational Effect of Brain-Scalp Distance on the Efficacy of Transcranial Magnetic Stimulation Treatment in Depression. *IEEE Trans. Magn.*, 2016;52(7):1-4. doi: 10.1109/TMAG.2015.2514158.
 - [33] Lee EG, Rastogi P, Hadimani RL, Jiles DC, Camprodon JA. Impact of non-brain anatomy and coil orientation on inter- and intra-subject variability in TMS at midline. 2018 Sep;129(9):1873-1883. doi: 10.1016/j.clinph.2018.04.749. Appendix A. Supplementary data.
 - [34] Logg A, Mardal K, Wells, G. (Eds.). *Automated Solution of Differential Equations by the Finite Element Method. The FEniCS Book*. 2012. Springer. New York. ISBN: 3642230989.
 - [35] Makarov SN, Yanamadala J, Piazza MW, Helderma AM, Thang NS, Burnham EH, Pascual-Leone A. Preliminary Upper Estimate of Peak Currents in Transcranial Magnetic Stimulation at Distant Locations From a TMS Coil. *IEEE Trans Biomed Eng*. 2016a Sep;63(9):1944-1955. doi: 10.1109/TBME.2015.2507572.
 - [36] Makarov SN, Noetscher GM, Nazarian A. *Low-Frequency Electromagnetic Modeling of Electrical and Biological Systems Using MATLAB*. Wiley, New York. 2016b. ISBN: 978-1-119-05256-2.
 - [37] Makarov SN, Noetscher GM, Raij T, Nummenmaa A. A Quasi-Static Boundary Element Approach with Fast Multipole Acceleration for High-Resolution Bioelectromagnetic Models. *IEEE Trans. Biomed. Eng*. 2018;65(12):2675-2683. doi: 10.1109/TBME.2018.2813261.

- [38] Makarov SN, Navarro de Lara L, Noetscher GM, Nummenmaa A. Modeling Primary Fields of TMS Coils with the Fast Multipole Method. *bioRxiv*. 2019. doi: <https://doi.org/10.1101/514919>.
- [39] Miranda PC, Hallett M, Basser PJ. The electric field induced in the brain by magnetic stimulation: a 3-D finite-element analysis of the effect of tissue heterogeneity and anisotropy. *IEEE Trans Biomed Eng*. 2003; 50: 1074-1085. doi: <https://doi.org/10.1109/TBME.2003.816079> PMID: 1294327.
- [40] Navarro de Lara LI, Rad LG, Makarov SN, Stockmann J, Wald LL, Nummenmaa A. Simulations of a birdcage coil B1+ field on a human body model for designing a 3T multichannel TMS/MRI head coil array. *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. Honolulu, HI, 2018:4752-4755.
- [41] Navarro de Lara L, Mascarenas A, Pratt K, Paulson D, Makarov S, Okada Y, Nummenmaa A. A novel three-axis TMS coil design for multichannel stimulation systems. *2019 OHBM Annual Meeting*. Rome, Italy, June 9-13. Accepted.
- [42] Nielsen JD, Madsen KH, Puonti O, Siebner HR, Bauer C, Madsen CG, Saturnino GB, Thielscher A. Automatic skull segmentation from MR images for realistic volume conductor models of the head: Assessment of the state-of-the-art. *Neuroimage*. 2018 Jul 1;174:587-598. doi: 10.1016/j.neuroimage.2018.03.001.
- [43] Nummenmaa A, Stenroos M, Ilmoniemi RJ, Okada YC, Hämäläinen MS, Raij T. Comparison of spherical and realistically shaped boundary element head models for transcranial magnetic stimulation navigation. *Clin Neurophysiol.*, 2013;124(10):1995-2007. doi: 10.1016/j.clinph.2013.04.019.
- [44] Opitz A, Paulus W, Will S, Antunes A, Thielscher A. Determinants of the electric field during transcranial direct current stimulation. *Neuroimage*. 2015 Apr 1;109:140-50. doi: 10.1016/j.neuroimage.2015.01.033. Epub 2015 Jan 19.
- [45] Piastra MC, Nüßing A, Vorwerk J, Bornfleth H, Oostenveld R, Engwer C, Wolters CH. The Discontinuous Galerkin Finite Element Method for Solving the MEG and the Combined MEG/EEG Forward Problem. *Frontiers in Neuroscience*, 2018;12(Article 30):1-18. doi: 10.3389/fnins.2018.00030.
- [46] Rahmouni L, Adrian SB, Cools K, Andriulli FP. Conforming discretizations of boundary element solutions to the electroencephalography forward problem. *Comptes Rendus Physique*, 2018;19(1-2):7-25. doi: <https://doi.org/10.1016/j.crhy.2018.02.002>.
- [47] Rastogi P, Lee EG, Hadimani RL, Jiles DC. Transcranial Magnetic Stimulation-coil design with improved focality. *AIP ADVANCES* 2017;7, 056705. doi: 10.1063/1.4986705.
- [48] Rokhlin V. Rapid Solution of Integral Equations of Classical Potential Theory. *J. Computational Physics*, 1985;60(2):187–207. doi: 10.1016/0021-9991(85)90002-6.
- [49] Saad Y. *Iterative Methods for Sparse Linear Systems*. 2nd Ed., Society for Industrial and Applied Mathematics. 2003. ISBN 978-0-89871-534-7.
- [50] Salinas FS, Lancaster JL, Fox PT. Detailed 3D models of the induced electric field of transcranial magnetic stimulation coils. *Phys Med Biol*. 2007 May 21;52(10):2879-92.
- [51] Salinas FS, Lancaster JL, Fox PT. 3D Modeling of the total electric field induced by transcranial magnetic stimulation using the boundary element method. *Phys Med Biol.*, 2009;54(12):3631-47. doi: 10.1088/0031-9155/54/12/002.

- [52] Salvador R, Silva S, Basser PJ, Miranda PC. Determining which mechanisms lead to activation in the motor cortex: A modeling study of transcranial magnetic stimulation using realistic stimulus waveforms and sulcal geometry. *Clin Neurophysiol.* 2011 Apr;122(4):748-58. doi: 10.1016/j.clinph.2010.09.022.
- [53] Sarvas J. Basic mathematical and electromagnetic concepts of the biomagnetic inverse problem. *Phys Med Biol.*, 1987;32(1):11-22. PMID: 3823129.
- [54] Saturnino GB, Puonti O, Nielsen JD, Antonenko D, Madsen KH, Thielscher A. SimNIBS 2.1: A Comprehensive Pipeline for Individualized Electric Field Modelling for Transcranial Brain Stimulation. *bioRxiv* 500314; 2018. doi: <https://doi.org/10.1101/500314>.
- [55] Silva S, Basser PJ, Miranda PS. Elucidating the mechanisms and loci of neuronal excitation by transcranial magnetic stimulation using a finite element model of a cortical sulcus. *Clin Neurophysiol.* 2008 Oct;119(10):2405-13. doi: 10.1016/j.clinph.2008.07.248.
- [56] Stenroos M, Mäntynen V, Nenonen J. A MATLAB library for solving quasi-static volume conduction problems using the boundary element method. *Comput Methods Programs Biomed.*, 2007;88(3):256-63. doi: 10.1016/j.cmpb.2007.09.004.
- [57] Stenroos M. Integral equations and boundary-element solution for static potential in a general piece-wise homogeneous volume conductor. *Phys Med Biol.*, 2016;61(22):N606-N617. Epub 2016 Oct 25. doi: 10.1088/0031-9155/61/22/N606.
- [58] Stenroos M, Nummenmaa A. Incorporating and Compensating Cerebrospinal Fluid in Surface-Based Forward Models of Magneto- and Electroencephalography. *PloS One*, 2016;11(7), e0159595. <http://doi.org/10.1371/journal.pone.0159595>.
- [59] Stenroos M, Sarvas J. Bioelectromagnetic forward problem: isolated source approach revis(it)ed. *Phys Med Biol.*, 2012;57(11):3517-35. doi: 10.1088/0031-9155/57/11/3517.
- [60] *The Population Head Model Repository*. IT'IS Foundation website. 2017. doi: 10.13099/VIP-PHM-V1.0. Onl: <https://www.itis.ethz.ch/virtual-population/regional-human-models/phm-repository/>.
- [61] Thielscher A, Antunes A, Saturnino GB. Field modeling for transcranial magnetic stimulation: A useful tool to understand the physiological effects of TMS?. *Conf Proc IEEE Eng Med Biol Soc.* 2015;222-5. doi: 10.1109/EMBC.2015.7318340.
- [62] Van Essen DC, Ugurbil K., Auerbach E, Barch D, Behrens TE, Bucholz R, Chang A, Chen L, Corbetta M, Curtiss SW, Della Penna S, Feinberg D, Glasser MF, Harel N, Heath AC, Larson-Prior L, Marcus D, Michalareas G, Moeller S, Oostenveld R, Petersen SE, Prior F, Schlaggar BL, Smith SM, Snyder AZ, Xu J, Yacoub E. The Human Connectome Project: A data acquisition perspective. *NeuroImage*, 2012;62(4):2222–2231. doi: 10.1016/j.neuroimage.2012.02.018. PMID: 22366334.
- [63] Vollmer J, Mencl R, Müller H. Improved Laplacian smoothing of noisy surface meshes. *Comput. Graph. Forum*, 1999;18(3):131-138. <https://doi.org/10.1111/1467-8659.00334>.
- [64] Wang B, Shen MR, Deng ZD, Smith JE, Tharayil JJ, Gurrey CJ, Gomez LJ, Petershev AV. Redesigning existing transcranial magnetic stimulation coils to reduce energy: application to low field magnetic stimulation. *J Neural Eng.* 2018 Jun;15(3):036022. doi: 10.1088/1741-2552/aaa505.
- [65] Wang Z, Volakis J, Saitou K, Kurabayashi K. Comparison of semi-analytical formulations and Gaussian-quadrature rules for quasi-static double-surface potential integrals. *IEEE Antennas and Propagation Magazine.* 2003;45(6):96-102. doi: 10.1109/MAP.2003.1282185.

- [66] Wilton DR, Rao SM, Glisson AW, Schaubert DH, Al-Bundak OM, Butler CM. Potential integrals for uniform and linear source distribution on polygonal and polyhedral domains. *IEEE Trans. Antennas and Propagation*, 1984;32(3):276-281. doi: 10.1109/TAP.1984.1143304.
- [67] Wolters C, Koestler H, Moeller C, Haerdtlein J, Grasedyck L, Hackbusch W. Numerical mathematics of the subtraction method for the modeling of a current dipole in EEG source reconstruction using finite element head models. *SIAM Journal on Scientific Computing*, 2007;30(1):24-45. doi. 10.1137/060659053.

Appendix A. Concise User's Manual

A1. Low-level organization chart

The toolkit contains a number of MATLAB scripts organized within three subfolders – Model, Coil, and Engine – and a number of scripts located in the main folder, as shown in Fig. A1. The subfolder Model deals with a particular head model definition and precomputations. The subfolder Coil handles coil definition, construction, and testing/optimization. The subfolder Engine contains computational scripts which should not be changed unless necessary. The scripts from the main folder perform major numerical operations and output the fields.

Fig. A1. Low-level organization chart of the toolkit.

A2. Computational sequence of the main folder

The scripts of the main folder could be executed at any time since all data have already been precomputed. The first two scripts must be executed sequentially, i.e. bem0_import_model_data.m first and then bem1_process_data.m next. The first script imports model data into the MATLAB workspace and sets the MATLAB path. The second script initializes (i) operation frequency and coil current; (ii) tissue conductivities; (iii) coil position above the head by proper rotation and translation; (iv) coil centerline or another observation line and; (v) displays

combined head-coil geometry as shown in Fig. 6a of the main text. Coil position adjustment may be performed by running the script `bem1_process_data` multiple times. When performing mathematical (e.g. coil rotation) and graphical operations, this script calls a number of functions from subfolder `Engine`.

The next script to be executed is `bem2_charge_engine.m`. This script (i) computes the primary field of the coil on every head interface using FMM; (ii) computes the iterative solution of Eq. (2) of the main text for the induced surface charge density using the FMM, precomputed near-field potential integrals, and MATLAB GMRES, and; (iii) displays iteration time for every iteration step in the MATLAB command window and then plots the entire convergence history when completed. Since no charge conservation law is explicitly enforced in the current version, the resulting charge error (typically below 0.5%) is also computed and displayed. The maximum permissible number of iterations and the targeted relative residual are defined at the beginning of the script. It is not recommended to change these default values. The execution time is approximately 1.5 min. When performing mathematical (FMM) operations, this script calls original and derived FMM functions from subfolder `Engine`.

Upon completion of this step, any of the field computational scripts may be executed, in any sequence. The script `bem3_surface_field_e` computes (via the FMM) and displays generally discontinuous electric field just inside or outside any head compartment, as shown in Figs. 7a,b,c of the main text. Normal (signed), tangential (magnitude), and total (magnitude) field components may be plotted. The script `bem3_surface_field_m` performs a similar operation for the magnetic field which remains continuous across the boundaries. The script `bem3_surface_field_q` displays induced surface charge density as shown in Fig. 7d of the main text.

Scripts `bem4_line_field_e/m.m` compute the three components of the electric/magnetic field along the coil centerline or, alternatively along any arbitrary line using the FMM with accurately averaged contributions of the surface charge density on neighbor triangles. An example is given in Fig. 8 of the main text. There and in the following scripts, parameter R characterizes dimensionless (vs. average edge length) radius of the integration sphere with the center at the observation point within which precise integration over triangular facets is performed.

The following scripts starting with `bem4_plane_coronal_e.m`, `bem4_plane_coronal_m.m` etc. output electric or magnetic field (any of its Cartesian components or a magnitude) in three principal planes. Accuracy of the electric field computation in the vicinity of the compartment boundaries is again controlled by parameter R – dimensionless radius of the precise-integration sphere. Finally, the script `bem5_volume_cortical.m` displays domains of the maximum normal or tangential electric field within the cortical layer between gray matter and white matter shells. A representative example is given in Fig. 9 of the main text. When performing mathematical (FMM) and graphical operations, all of these scripts call a number of functions from subfolder `Engine`.

A3. Computational sequence of subfolder `Model`

The head model files are located in the dedicated folder `Model` with contents shown in Fig. A1. The primary set are *.stl (stereolithography) files for every individual brain compartment in the form of a faceted shell with triangular facets and the respective outer normal vectors. The number of shells may be arbitrary. Every stl file, either binary or ASCII, is then converted to an equivalent MATLAB data file seen in Fig. A1 with arrays of vertices, P , facets, t , and normal vectors, $\mathbf{n}$, using the script `model01_stl_to_mat`. Every MATLAB data file can be inspected and visualized using the `viewer` function from the same subfolder as shown in Fig. 1a of the main text. Next, the combined mesh for the entire head is created by appending individual meshes. The combined mesh

is stored in the MATLAB data file `CombinedMesh.mat`. This is done by running the script `model02_main_script.m`. There is also an additional and bigger data file `CombinedHeadP.mat` generated in the same folder, which contains precomputed double surface electrostatic integrals over triangles from Eq. (11) necessary for accurate BEM-FMM simulations. Computations of integrals are done in parallel and using the default number of 24 cores. The `parfor` loop of MATLAB may be adjusted or removed depending on the computer configuration.

The present package is augmented with 16 head models for 16 Connectome subjects (101309, 110411, 117122, 120111, 122317, 122620, 124422, 128632, 130013, 131722, 138534, 149337, 149539, 151627, 160123, 198451). These models (and their refined versions) described in (Htet et al., 2018) could be downloaded independently from the MATLAB Central link (Collection of Sixteen High-Quality Human Head CAD Models. 2019).

The following three scripts are `model03_cross_section_XY.m`, `model03_cross_section_XZ.m`, and `model03_cross_section_YZ.m`. They compute tissue mesh intersections with three principal planes specified by the user. The intersection contour for every tissue is found and saved in the respective MATLAB data file. For skin, skull, and CSF shells, such a contour is given by a 2 manifold polygon, which could be filled. An example is given in Fig. 1b.

The script `model06_intersection_test.m` finds tissue mesh intersection points with an arbitrary line. These data may be useful for coil positioning. Finally, the script `model07_cortical_volume.m` creates and saves a point cloud between gray matter and white matter shells: the cortical layer. This point cloud contains five or more inner observation points across the layer, which are uniformly spaced along the normal direction of every triangle of the GM shell. It is used to compute hot spots of the electric field within the cortex as shown in Fig. 9 of the main text. When performing mathematical (potential integrals, mesh intersections) and graphical operations, these scripts call a number of functions from subfolder `Engine`.

A4. Computational sequence of subfolder `coil`

First, a coil model (both wire and CAD) is generated using a dedicated MATLAB script, with *one* script per coil. The bulk of the manual work here is the definition of the conductor's centerline using either an analytical formula or a set of points in three dimensions. After that, the script automatically generates the volumetric computational wire grid coil model and the illustrative CAD coil model. The default coil axis is the *z*-axis.

The first task (wire model shown in Fig. 2 of the main text) is accomplished by the function `meshwire.m` of subfolder `Engine`. This function creates the wire mesh for an arbitrary *single* conductor. Either closed loops (`coil01_SingleRing.m`) or open conductors (`coil01_SingleRung.m`) may be considered. The computational wire grid coil model consists of straight, short, infinitely-thin current filaments or segments. The current filaments (shown in red in Fig. 2 of the main text) are defined as short straight lines joining centroids of triangles of the cross-sectional mesh, which is replicated along the conductor's centerline as many times as required. The cross-section is always perpendicular to the conductor's centerline; the filaments are therefore parallel to the centerline of the conductor. The result is structure `strcoil` with the following fields:

```
Pwire - nodes of elementary wires inside the conductor
Ewire - edges (start & end points) of elementary wires inside the conductor
Swire - weights of elementary wire segments given total current of 1A
```

Weights are necessary to ensure that the total current through the conductor's cross-section is 1 A.

Along with the function `meshwire.m`, the function `meshcoil.m` can be used. This function performs a similar task, but for a particular coil geometry consisting of a (large) number of concentric loops; it creates the wire coil model all at once. The input are intersection points of the loop centerlines with xz - and yz planes.

Either a Litz wire model (parameter `sk = 0`) or a skin-layer model (parameter `sk = 1`) can be considered in both functions `meshwire.m`, and `meshcoil.m`. In the former case, the current distribution across conductor's gross-section is nearly uniform. In the latter case, the wire grid is situated close to the surface of the conductor. The density of the wire grid depends on cross-section triangulation; it is controlled by parameter `m` – the number of cross-section subdivisions. The grid resolution in the direction of the conductor centerline is controlled by the original centerline discretization.

A CAD model for the coil conductor is constructed using the function `meshsurface.m` of subfolder `Engine` and by creating a structured triangular surface mesh for the conductor's side surface. This mesh is created based on the array of edges `e` for the conductor's perimeter (which may either be elliptical or rectangular); the perimeter contour stays the same when moving along the centerline. At every discrete step along the conductor's centerline (such steps are illustrated in Fig. A2), we add triangular facets in an ordered way. The corresponding MATLAB code accumulates side facets into array `t` as follows

```
t1(:, 1:2) = e; % Two lower nodes
t1(:, 3) = e(:, 1) + NE; % One upper node
t2(:, 1:2) = e + NE; % Two upper nodes
t2(:, 3) = e(:, 2); % One lower node
ttemp = [t1; t2];
t = [t; ttemp+NE*m];
```

where `NE` is the number of edges (and nodes) for the conductor's perimeter contour. At every step, we also add `NE` new nodes. Next, start/end caps for non-closed conductors are created and duplicated nodes are removed. As a result, the solid CAD model of a single coil conductor and the entire coil is created as shown in Fig. 3 of the main text.

Note that the coil model does not yet define the normal vectors of triangular surface patches, nor does it properly orient the triangles. Therefore, after converting to STL format, a mesh healing operation is required, for example in ANSYS Electronics Desktop. Fig. A2 shows a more detailed concept of the combined wire/CAD model.

Fig. A2. Filaments of current (red) within conductor's surface CAD model. a) – Uniform current distribution (Litz wire); b) – modeling the skin effect (a solid conductor at a high frequency).

At present, the coil geometry modeler is restricted to predominantly flat or moderately bent conductor loops or nearly planar curves. H-coils (see, for example, Deng, et al., 2013) with sharp conductor bends in all three planes may be subject to further investigation, along with coil inductance calculations.

After the coil model has been created, line plots (`coil04_tester_line_e/m.m`) as well as high-resolution 2D contour plots (`coil05_tester_plane_coronal_e/m.m` etc.) for any component of the electric and/or magnetic field in the coronal, sagittal, and transverse planes are computed via the FMM, similar to the plots shown in Fig. 5 of the main text. When performing mathematical (mesh generation, FMM computations) and graphical operations, these scripts call a number of functions from subfolder `Engine`. A coil mesh generator script and a field computation script may be further combined into one script and augmented with a parametric loop to enable coil analysis and design (Makarov et al., 2019).

A5. Software compatibility

The toolkit has been tested on both Windows and Linux platforms using MATLAB 2018a. Generally, no MATLAB toolboxes are necessary with one minor exception. The script `model02_main_script.m` from subfolder `Model` and the function `bemf5_volume_field.m` for computing volumetric fields from subfolder `Engine` both still use a fast built-in function `rangesearch` from the Statistics and Machine Learning Toolbox in order to identify the neighbor facets. This MATLAB toolbox should therefore be installed or a replacement for this function needs to be found.

A6. Known problems

The toolkit will not run with the older versions of MATLAB. Another difficulty might appear when a script cannot find the proper MATLAB path or the previously loaded data. In this case, it is recommended to start the process over, ideally with the first script `bem0_import_model_data.m` of the main folder as described in Section A2.