

Open Source Gradient Coil Design Tool “CoilGen”

Team name: CoilGen

Team:

Philipp Amrein^{1 2}

Feng Jia¹

Maxim Zaitsev¹

Sebastian Littin¹

¹Medical Physics, Department of Radiology, Faculty of Medicine, University Freiburg, Killianstr. 5a, 79106 Freiburg, Germany.

² Neoscan Solutions, Joseph-von-Fraunhofer-Straße 6, 39106 Magdeburg, Germany

CoilGen on github: <https://github.com/Philipp-MR/CoilGen>

PyCoilGen on github: <https://github.com/kev-m/pyCoilGen> (Translation mainly by Kevin Meyer)

How can CoilGen advance MRI Research?

Gradient coils are used in MRI for spatial encoding and/or diffusion weighting. Along with shim coils, they are main components of every MRI machine. Besides CoilGen the following open source tools which allow to optimize MRI gradient and/ or shim coils are known to us:

LUMC tool by Patrick Fuchs:

<https://github.com/LUMC-LowFieldMRI/GradientDesignTool>

Thin-wire coil design:

https://github.com/Sebastian-MR/ThinWire_MRIGradientCoilDesign

Gradient Matlab scripts (Gael Bringout): <https://github.com/gBringout/CoilDesign/>

All of these tools enable generating a stream function which is used to approximate a surface current density with discrete wires. However, all these tools require manual and tedious post-processing. This is necessary to interconnect iso-contours of the stream function to a continuous wire path.

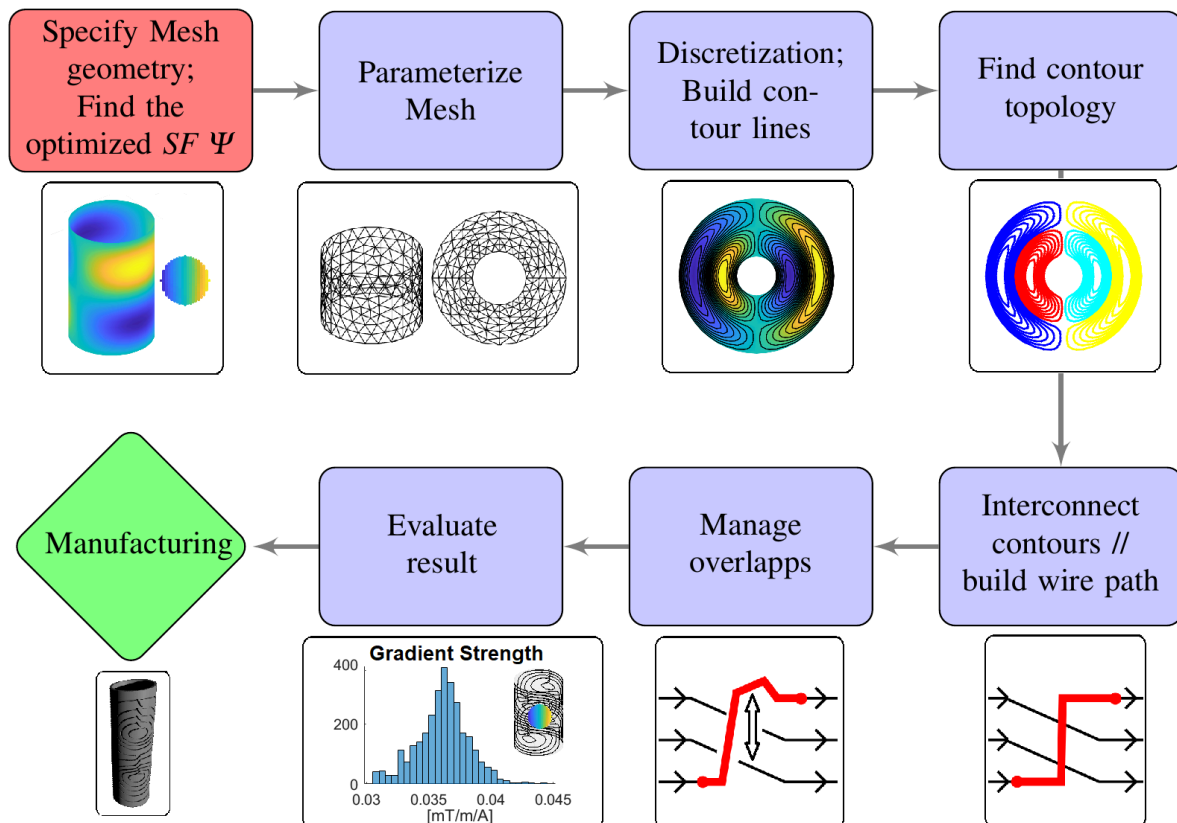
Besides coil design similar to other design tools, CoilGen has advanced post-processing steps. This significantly speeds up the design and enables for rapid prototyping of shim and gradient coils. Due to the availability in MATLAB and completely open source in Python it can be easily accessed.

Multiple gradient coils have been designed, e.g. recently for the MRI4all Hackathon at NYU and at the ezyMRI Nerdfest right before ISMRM in Singapore.

What are the functionalities and capabilities of CoilGen?

CoilGen enables a coil design for arbitrary target magnetic fields on arbitrary target surfaces. The tool uses the boundary element method for deriving interconnected coil layouts to approximate a surface current density.

Beyond traditional coil design, CoilGen has advanced post processing functions included. A topological analysis is performed and interconnections within and among groups are made. Options to directly derive designs for double-layer PCBs and to export .stl files of a solid wire are included. This significantly reduces the necessary post processing to achieve a realizable coil design of shim and gradient coils. CoilGen was initially implemented using MATLAB. By now it has been translated into Python and is openly available as PyCoilGen.



Flowchart depicting the individual steps of the layout generation algorithm. The red node represents the start for which a pre-optimized stream function is either imported or generated by the provided optimization functionality for a specific target field. The following blue nodes are the postprocessing steps which are proposed in this work for an automated solution.