Fitting a single equivalent-current-dipole model to MEG data with exhaustive search optimization is a simple, practical and very robust method given the speed of modern computers

Keith S Cover & Bob W van Dijk

VU University Medical Centre

VU medisch centrum

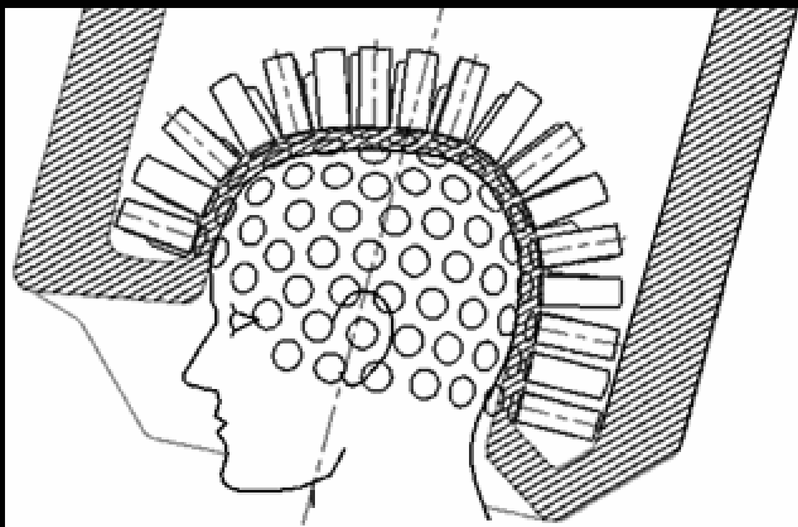
vl·e

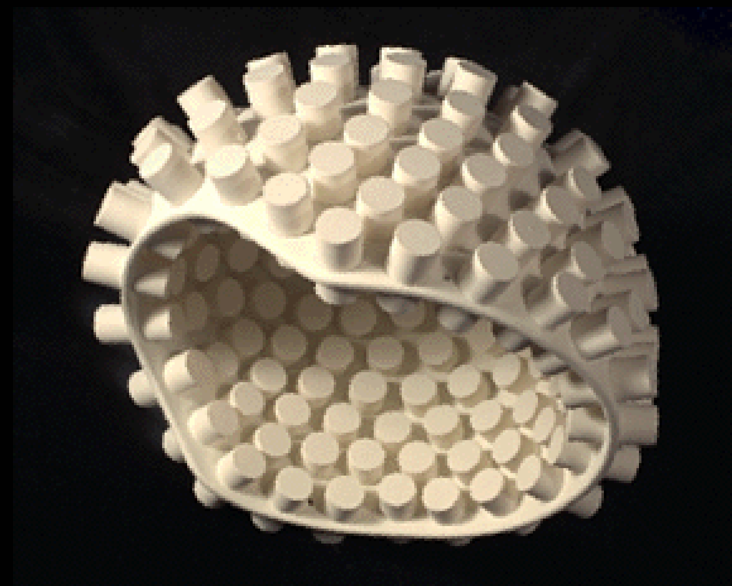
Outline

- What is MEG?
- ECD Fitting
- GRID Implementation of ECD fitting
- Lessons learned
- Acknowledgements

What is MEG?

What is MEG?

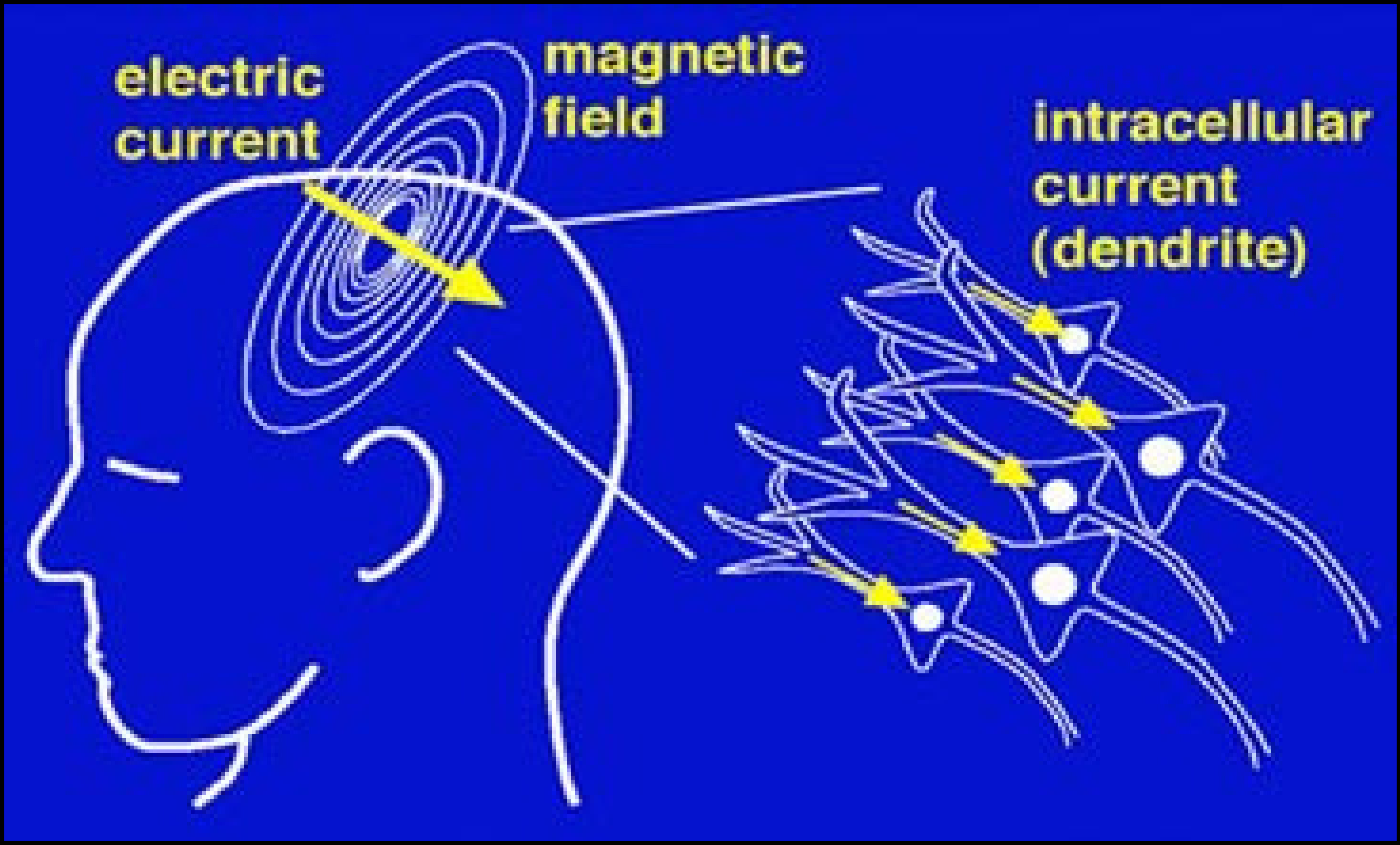




What is MEG?

- Acquire data at 151 coils simultaneously
- Sample rate 300Hz to 3000Hz
- Acquire for 5 to 60 minutes
- Data sets from 50MB to 1GB

ECD Fitting



Fitting Equivalent Current Dipoles (ECDs)

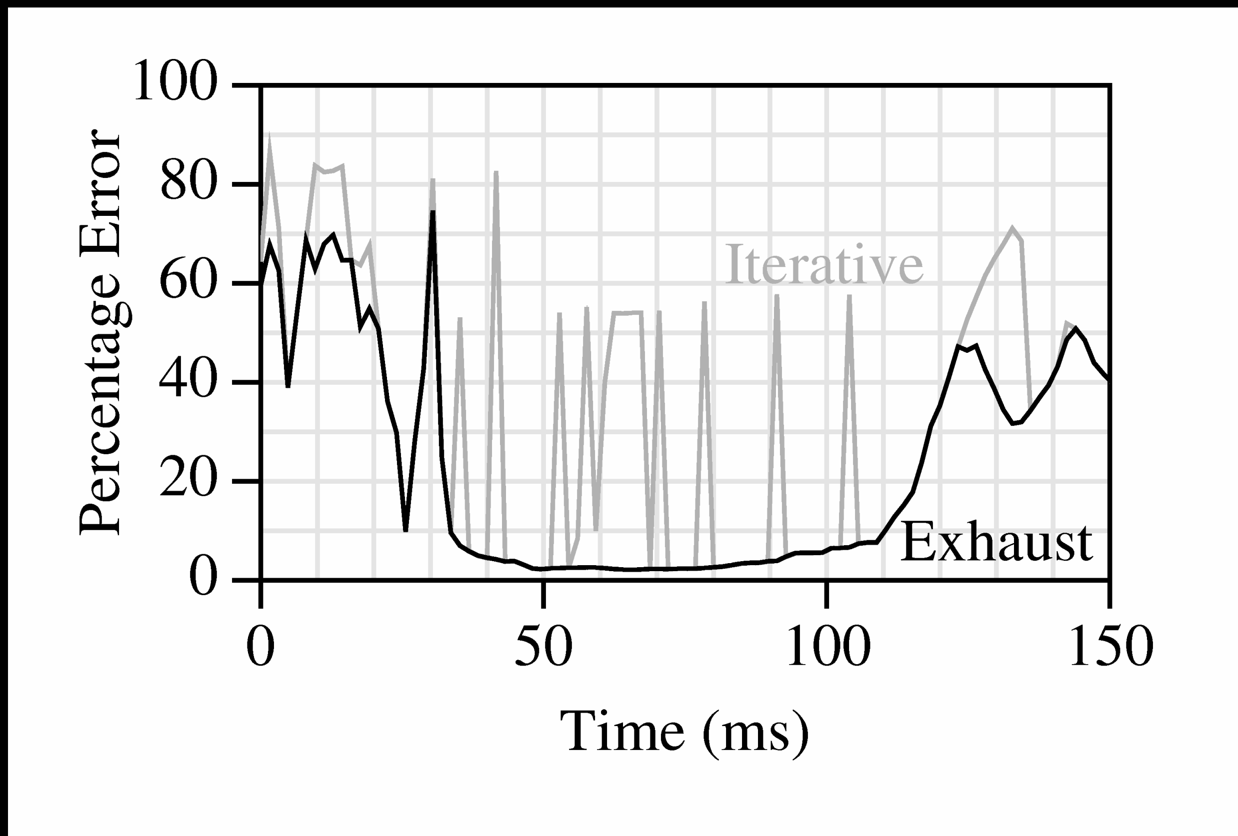
ECD Fitting

- Fit ECD model to data at each time point
- Least squares fit to the MEG data is a common practice
- 6 parameters to fit
  - 3 locations
  - 3 orientations

ECD Fitting

- Least squares
  - convex in 3 orientation parameters
  - multiple local minima in 3 location parameters
- Use exhaustive search over 3 location parameters (roughly 2,000,000 locations)
- Much more robust solution than iterative

ECD Fitting



- Exhaustive search reliably finds best fit

GRID Implementation

- Initially implemented as single thread Java
- Can fit 1 point every 8 seconds
- Would like to fit at least 20 different 3 million point data sets (ECD detection)
- Will take 130,000 cpu-core hours
- Use the GRID!!

GRID Implementation

- Initially implemented on DAS2
- Divide task into time intervals using lock files to share load over nodes (1 hour jobs)
- Took many months of coding
- Implementation not very portable
- Moving implementation to JavaGAT for portability

GRID Implementation

- Slow to debug - no IDE
- Difficult to add new features compared to single computer
- Handling the large volume of data produced requires special software tools
- Response times can be on order of hours

Lessons Learned

- Using the GRID to speed up the processing time of single large data sets requires a great deal of time and effort
- Changing design is very expensive compared to single computer
- Stick to trivially parallel jobs whenever possible - especially when trying new ideas
- JavaGAT is a very useful GRID package

Acknowledgements

Jan C. de Munck, VUmc

Jeroen Verbunt, VUmc

Vumc

VL-e

[redacted]