Toolkits and Techniques for Debugging Inverse Problems

A Boyle[1], WRB Lionheart[2], A Adler[3]

[1] *University of Ottawa, Ottawa, Canada* [2] *University of Manchester, Manchester, UK*
[3] *Carleton University, Ottawa, Canada*
Corresponding author: aboyle2@uottawa.ca

Inverse problems lead to inherently low resolution images which can be difficult to verify. Regularization can stabilize solutions, but lead to bias. On occasion, image reconstruction will go poorly. Typically, the algorithms successfully produce images, but the reconstructed images have artefacts which may lead to wrong interpretations. Our work is motivated by this "debugging" process, leading to two questions: How should one go about determining whether a result is satisfactory? And, if the result is wrong, what caused the failure?

In this work, we report our processes, and the techniques used to find issues in the specific context of impedance imaging and, more generally, for inverse problems. We focus on the algorithmic aspects: the challenges in validating inverse problem codes as well as their inputs and outputs. This represents early work on a novel aspect of inverse problems [1]. This work is naturally related to theories of software and algorithm debugging [2, 3], defect and root cause analysis [4, 5] and business process analysis [6, 7]. Methods for debugging software or hardware are very heuristic by nature; they tend to be problem dependent. We propose a set of tools to enable a "debugging" workflow for evaluating an inverse problem result. A systematic process such as this can help the dissection of an algorithm to locate a specific cause within the code base or algorithm inputs. Our workflow provides debugging tools at three stages for evaluating: 1) data and model quality, 2) algorithm behaviour, and 3) regional image fit.

In examining the data, we wish to avoid conflating the reconstruction algorithm with the quality of the data. To achieve some degree of independence between the two, we initially rely solely on the forward model's correct implementation to examine the data (Figure 1). Data that agrees with the model to a certain degree will result in a "reasonable" reconstruction while a significant data-model mismatch may cause a "suspicious" artifact. A suspicious artifact may either be ignored, by discarding or de-weighting measurements, or the cause of the problem can be resolved by adjusting the model to make the data useful in the reconstruction.

By developing these debug tools, we aim to improve the rigour and quality of image reconstruction in a comprehensive framework by enabling quick debug cycles that can feedback to improved data acquisition strategies and better interpretive outcomes.

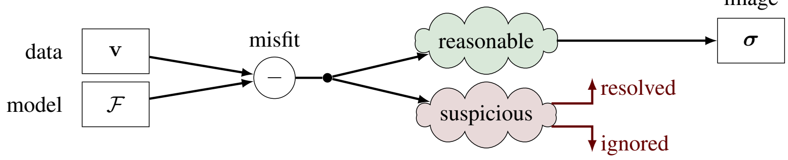

Figure 1: Explanations for data quality and model mismatch

[1] A. Boyle. Ph.D. thesis, Carleton University, Ottawa, Canada (2016).
[2] S. Gill. *Proc. Royal Soc. London A*, 206(1087) (1951) 538.
[3] A. Zeller. *Why programs fail*. Morgan Kaufmann Publishers, Inc, 2nd edn. (2009).
[4] K. Ishikawa. *Guide to quality control*. Asian Productivity Organization Tokyo (1982).
[5] P. Wilson, L. Dell, G. Anderson. *Root cause analysis*. ASQ Quality Press (1993).
[6] M. Harry, R. Schroeder. *Six Sigma: The Breakthrough Management Strategy*. Random House (2000).
[7] T. Ohno. *Toyota Production System: beyond large-scale production*. Productivity Press (1988).

Keywords: debug, impedance imaging, tomography, EIT, ERT, data misfit