

This is a self-archived version of an original article. This version may differ from the original in pagination and typographic details.

Author(s): Helenius, A. P.; Puurtinen, T. A.; Kinnunen, K. M.; Maasilta, I. J.

Title: Simultaneous Noise and Impedance Fitting to Transition-Edge Sensor Data Using Differential Evolution

Year: 2020

Version: Published version

Copyright: © The Authors 2020

Rights: CC BY 4.0

Rights url: <https://creativecommons.org/licenses/by/4.0/>

Please cite the original version:

Helenius, A. P., Puurtinen, T. A., Kinnunen, K. M., & Maasilta, I. J. (2020). Simultaneous Noise and Impedance Fitting to Transition-Edge Sensor Data Using Differential Evolution. *Journal of Low Temperature Physics*, 200(5-6), 213-219. <https://doi.org/10.1007/s10909-020-02489-0>



Simultaneous Noise and Impedance Fitting to Transition-Edge Sensor Data Using Differential Evolution

A. P. Helenius¹ · T. A. Puurtinen¹ · K. M. Kinnunen¹ · I. J. Maasilta¹

Received: 5 August 2019 / Accepted: 6 June 2020
© The Author(s) 2020

Abstract

We discuss a robust method to simultaneously fit a complex multi-body model both to the complex impedance and the noise data for transition-edge sensors. It is based on a differential evolution (DE) algorithm, providing accurate and repeatable results with only a small increase in computational cost compared to the Levenberg–Marquardt (LM) algorithm. Test fits are made using both DE and LM methods, and the results compared with previously determined best fits, with varying initial value deviations and limit ranges for the parameters. The robustness of DE is demonstrated with successful fits even when parameter limits up to a factor of 10 from the known values were used. It is shown that the least squares fitting becomes unreliable beyond a 10% deviation from the known values.

Keywords Thermal model · Genetic algorithm · Differential evolution · Transition-edge sensor

1 Introduction

Transition-edge sensors (TES) are versatile, state-of-the-art radiation detectors [1, 2], currently used in many applications, such as particle-induced X-ray emission spectroscopy [3, 4], and ground- and space-borne telescopes [5, 6]. However, the modelling of transition-edge sensors and finding fits to data has sometimes proven quite challenging in practice, due to the complexities of the thermal circuit of the device [7–12], as two- and three-block thermal models [13] need to be employed at times. Fitting these models by the commonly used least squares fitting methods or with certain initial guesses manually, as was done in references [7, 8], has proven to be tedious or even unreliable.

Here, we propose a different approach to fit TES thermal models, which is independent of the initial parameters given, and can fit both the complex impedance and

✉ A. P. Helenius
ari.p.helenius@jyu.fi

¹ Nanoscience Center, Department of Physics, University of Jyväskylä, 40014 Jyväskylä, Finland

the noise data simultaneously, even for three-block models, producing more reliable results than the more commonly used Levenberg–Marquardt method [15]. It is based on the differential evolution (DE) algorithm [14], a branch of genetic algorithms. In this study, we fit previously measured data from Ref. [7] both with the DE algorithm and with the LM algorithm and use the published manual fits in that paper as the control to evaluate the performance.

2 Two- and Three-Block Thermal Models

In Ref. [7], it was shown that for good fitting of the complex impedance and noise data, three-block thermal models had to be employed. The models chosen for study here are the so-called hanging (H) model for two-block and intermediate–hanging (IH) model of Ref. [13], see Fig. 1. In the IH model, in addition to the heat capacity of the TES sensor element, C_{tes} , there are two additional heat capacities: one intermediate, C_2 , and one hanging, C_1 . For the H model, only one additional heat capacity C_1 is connected. The full equations for the complex impedance and for all the noise terms are lengthy and can be found in full detail from Ref. [13].

3 Genetic Algorithm and Differential Evolution

Differential evolution (DE) is a high-performance, yet simple, optimizer algorithm based on mutation and crossover of the trial argument vectors of the fit function [14]. DE is initialized by selecting a number of initial population entities (typically D -dimensional vectors x_i) that are evaluated with the cost function. In each step, trial vectors are randomly mutated $y_i = x_i + F(x_j - x_k)$, where $F \in [0, 2]$ is the mutation factor, and a crossover is performed by randomly mixing the vector elements $(y_i)_k \rightarrow (y_j)_k$ of two distinct entities y_i and y_j . The cost function is then re-evaluated for the decision whether the trial is kept or discarded from the population.

Three different functions are simultaneously fitted to the measured data. The complex impedance is broken down to its real and imaginary parts and used as the first two fit functions. The third function comes from the total TES current noise,

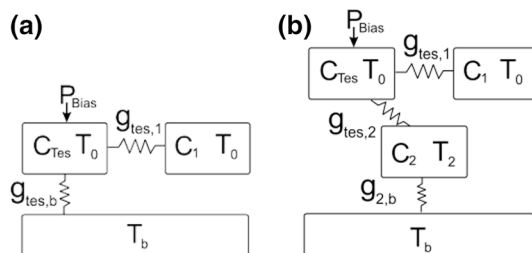


Fig. 1 **a** The two-block hanging model and **b** the three-block thermal model used in this study, with additional intermediate, C_2 , and hanging, C_1 , heat capacities. Parameters g_i describe the differential thermal conductances connecting the heat capacities, and T_i are the steady state temperatures of the blocks

including a constant $(4 \text{ pA}/\sqrt{\text{Hz}})^2$ SQUID noise component. The cost function to be minimized for both the DE and LM fitting methods is the sum of squared errors (considered as dimensionless numbers), summed over the real and imaginary parts of the complex impedance and the noise. For the noise data and fit, an additional \log_{10} is taken prior to the subtraction. In order to normalize the relative weighting between the impedance and noise datasets, the number of impedance data points is increased to match the noise. All the calculations are done with Python 3.7, NumPy version 1.15.1, and SciPy version 1.1.0, and both of the optimization algorithms are from the package `scipy.optimize`: *least_squares* and *differential_evolution*. The DE strategy was “best2bin,” and following parameters were used for the algorithm: population size 15, mutation 1.8, recombination 0.1, tolerance 10^{-7} , and absolute tolerance 0. No seed was chosen, in order to see whether the fit found is always the same. Polishing was used, which runs *scipy.optimize.minimize* with limited memory Broyden–Fletcher–Goldfarb–Shanno algorithm, and initial population was determined by “latinhypercube.”

4 Calculations and Results

A total of six fitting parameters were chosen for the three-block model to be free parameters for the fitting tests: All the three heat capacities of the model, the two steady state temperatures T_0 and T_2 , and one of the thermal conductances, $g_{\text{tes},2}$. $g_{\text{tes},1}$, α_l and β_l were kept fixed at the values determined in Ref. [7], and $g_{2,b}$ is not free anymore, if $g_{\text{tes},2}$, T_0 and T_2 are set, as the overall dynamic conductance to the bath is known from the I-V measurements [13].

For the estimation of initial values, the Corbino geometry of the devices of Ref. [7] allows for a reasonable estimation of the heat capacities C_{tes} and C_1 , but very little is known beforehand on C_2 , the “excess” heat capacity. The TES (T_0) and the intermediate block (T_2) temperatures have certain limits that they follow, but are typically not exactly known. A good initial guess for the TES temperature can be calculated from the I-V curves, based on the bias point dissipated power $P = IV$ and its measured bath temperature T_{bath} dependence, $P = K(T_0^n - T_{\text{bath}}^n)$, by $T_0 = \left(\frac{P}{K} + T_{\text{bath}}^n\right)^{\frac{1}{n}}$. The intermediate block temperature is somewhere between the TES temperature and the bath temperature, depending on the values of $g_{\text{tes},2}$ and $g_{2,b}$. Figure 2 shows the obtained DE (red) and LM (blue) fits of complex impedance and noise at several bias points, with a 10% deviation of the parameter limits from the manual control fit [7] (green). We see that both methods, in this case, give reasonable fits, with the DE method giving slightly better results.

We compared both algorithms with the same lower and upper parameter limits ($\pm 1\%$, $\pm 10\%$, or $\pm 50\%$) with several different runs (initial conditions). For the LM fit, the initial guess is randomized from the control fit parameters [7] by an additional factor of $\pm 1\%$, $\pm 10\%$, or $\pm 20\%$, respectively. The LM fit for $\pm 1\%$ is good, but with the $\pm 10\%$ limits, LM algorithm gets stuck in local minima and loses its consistency, as shown in Fig. 3. However, the DE algorithm is robust even with $\pm 50\%$ limits (Fig. 3). This underlines the importance of the accuracy of the initial parameters

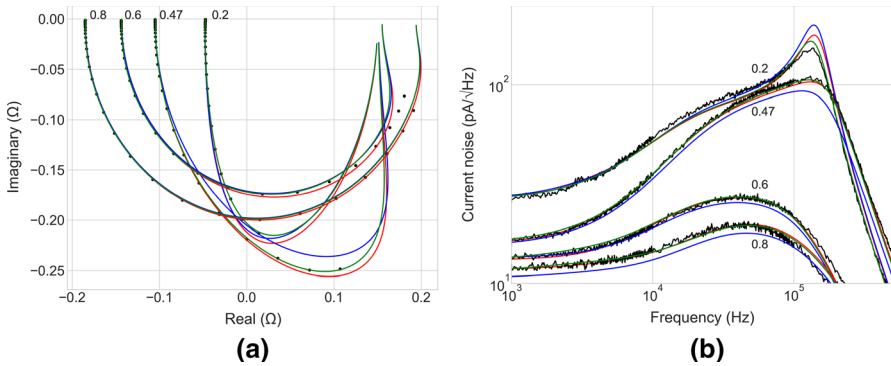


Fig. 2 (Color online) **a** Impedance fits (curves) and data (points). Data are only taken up to 100 kHz, fits shown up to 1 MHz. **b** Noise fits (colored curves) and data (black). The electrical cutoff due to the measurement circuitry is visible above 100 kHz frequencies. Manual fits (green lines), DE (red lines), and LM (blue lines), with 10% deviation in limits. The bias points range from 0.2 to 0.8 R/R_N

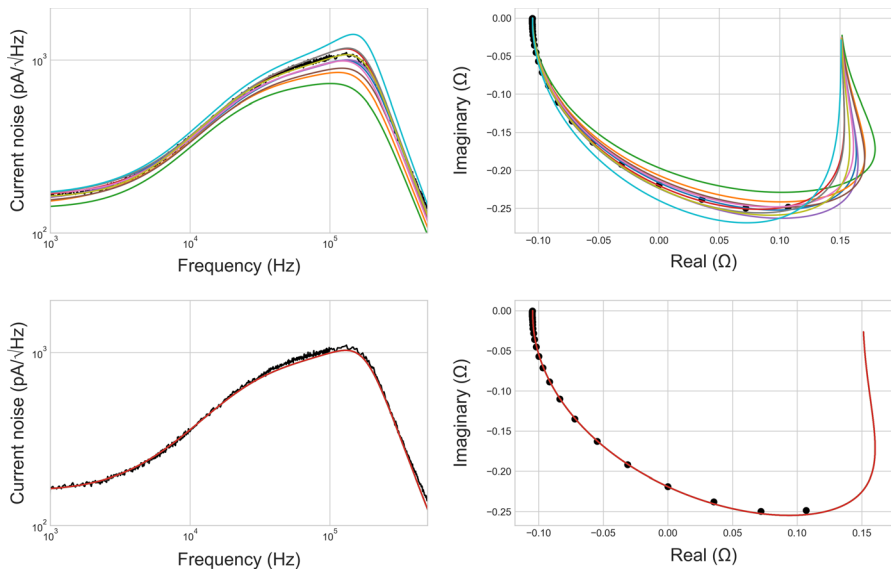


Fig. 3 (Color online) Ten randomized LM fits with 10% (top) and ten randomized DE fits with 50% limits (bottom) deviation from the initial values, for the same dataset. Black dots and lines represent the data for impedance and noise, respectively. The upper limit for the number of iterations for both LM and DE was 5000

for Levenberg–Marquardt algorithm and, conversely, the robustness of the differential evolution algorithm. A two-block H model with similar parameters was also explored with artificially generated noisy data. DE still performs better, but the difference between the performance of the algorithms is not as significant.

Relative weighting between the noise and impedance data was also explored for the DE algorithm. Examples of fits and their residuals are shown in Fig. 4, with

Table 1 Four differently weighted scenarios with 20 run average (relative standard error) of fitting parameters.

Z/N (50% dev)	T_{tes} (mK)	T_{ex} (mK)	$g_{tes,2}$ (nW/K)	GoF: Re (Im)(arb.u.)
95/5	163.1 (0.13%)	119.8 (0.46%)	1.07 (0.33%)	0.040 (0.051)
75/25	163.0 (0.20%)	120.2 (0.52%)	1.06 (0.63%)	0.041 (0.051)
50/50	162.9 (0.54%)	120.2 (1.6%)	1.06 (1.3%)	0.044 (0.053)
25/75	163.1 (0.59%)	119.8 (1.4%)	1.06 (1.7%)	0.052 (0.055)
100/0	163.0 (0.19%)	85.7 (43.9%)	1.25 (0.16%)	0.038 (0.081)
Z/N (50% dev)	C_{tes} (fJ/K)	C_{abs} (fJ/K)	C_{ex} (fJ/K)	GoF: Noise (Sum)(arb.u.)
95/5	74 (1.1%)	267 (0.11%)	447 (1.6%)	0.0836 (0.175)
75/25	74 (1.4%)	269 (0.14%)	446 (0.84%)	0.0842 (0.176)
50/50	72 (2.7%)	269 (0.26%)	446 (1.8%)	0.0843 (0.181)
25/75	70 (3.2%)	269 (0.11%)	449 (0.22%)	0.0852 (0.192)
100/0	74 (1.3%)	267 (0.58%)	400 (18.5%)	– (0.119)

Noticeable difference can be found between fitting impedance only, noted by additional Z in ratios, and fitting both noise and impedance data simultaneously. Last columns for both levels are goodness-of-fit (GoF) values in arbitrary units

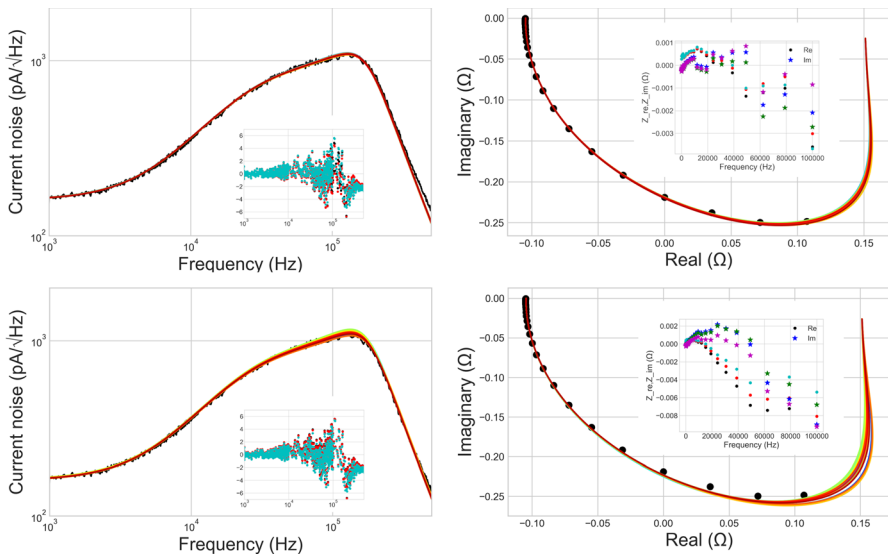


Fig. 4 (Color online) Weight ratios 75%, 25% (top) and 25%, 75% (bottom) for impedance and noise, respectively, 20 runs per set. Inset figures show residuals of three randomly chosen runs versus frequency. Variance in impedance fits is easily visible when noise is weighted more heavily. Better results are found when weighting is in favor of impedance. Noise tends to fit decently with almost any weight ratio; however, the best total fits are given when weight ratio is heavily in favor of impedance. Roughly, 1700 iterations and 50% deviation were used for these runs

averages and standard errors for the fit parameters, and overall goodness-of-fit measures (cost function metric) are shown in Table 1 for all ratios studied. The observed general trend is that increasing the weighting ratio to favor impedance increases the consistency of the fits. We also see from Table 1 that fitting only the impedance does give quite different fit parameter values, which do not describe the noise data well.

Finally, in Fig. 5, we show the effect of increasing the parameter limits significantly for the DE algorithm, to demonstrate its robustness. For the factor of ten deviated DE fits, the fit tends to change the phonon and internal fluctuation components the most, by changing the temperature of the extra heat capacity (from ~ 120 mK up to ~ 164 mK, which is equal to the TES temperature) and by increasing $g_{\text{tes},2}$ by a factor of eleven. Looking at both fits individually without a prior knowledge of the model, they both look like reasonable physical solutions. Thus, one should help the algorithm with all the intuition available. In this case, we could have limited the TES temperature to a smaller deviation from a calculated value, or in general, one could limit the fit parameters individually based on prior knowledge.

5 Conclusions and Outlook

We have implemented a robust method for simultaneous fitting of complex impedance and noise data of TES detectors by a differential evolution (DE) algorithm. A three-block thermal model could be fitted to the data manually, but with DE, the problem of choosing the initial parameters and the tediousness and unreliability of the fitting process is removed. When the number of fitting parameters is small and the limits are close to the actual physical values, the Levenberg–Marquardt algorithm will be faster, finding the solution in a matter of seconds, whereas DE takes roughly a minute. However, a few extra minutes, in the case of less accurate initial estimates, is a small price to pay for the robustness and reliability that DE offers.

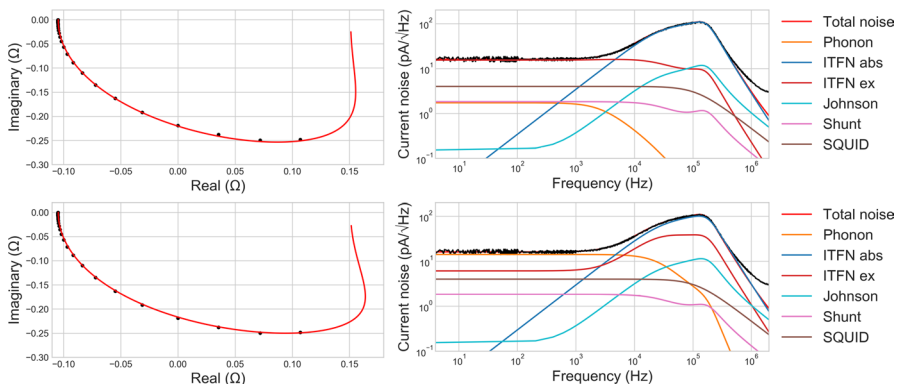


Fig. 5 (Color online) Top: 50% deviation fits with noise components visible. Bottom: fits with 10x deviation. Two different solutions found with different weighting of phonon and extra heat capacity internal fluctuation noise components. Slight difference is also noticeable in the impedance fits on the left with 10x having a larger kink prior to the tilting back to Z_{∞} . For both, 5000 iterations were used

It was also found that the fits for the tested dataset were more consistent, if impedance data were weighted more heavily. However, only taking into account the impedance leads to much worse noise fits and large parameter deviations from the cases where both impedance and noise are simultaneously fitted.

In addition, when running the calculations with larger deviations, multiple different solutions may arise. This may require user intervention, in some cases, to drive the system toward a more physical solution. Nevertheless, DE-based fitting algorithms can help avoid some of the caveats commonly encountered in multivariable nonlinear fitting problems, and in particular, it is a reasonable tool for the data analysis of TES detectors.

Acknowledgements Open access funding provided by University of Jyväskylä (JYU).

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. K.D. Irwin, G.C. Hilton, *Cryogenic Particle detection*, edited by Ch. Enss (Springer, New York, 2005)
2. J.N. Ullom, D.A. Bennett, *Supercond. Sci. Technol.* **28**, 084003 (2015). <https://doi.org/10.1088/0953-2048/28/8/084003>
3. M.R.J. Palosaari et al., *J. Low Temp. Phys.* **176**, 285 (2014). <https://doi.org/10.1007/s1090-013-1004-5>
4. M.R.J. Palosaari et al., *Phys. Rev. Appl.* **6**, 024002 (2016). <https://doi.org/10.1103/PhysRevApplied.6.024002>
5. Y. Inoue et al., *Proc. SPIE* **9914**, 99141I (2016). <https://doi.org/10.1117/12.2231961>
6. W.B. Doriese et al., *IEEE Trans. Appl. Supercond.* **29**, 5 (2019). <https://doi.org/10.1109/TASC.2019.2905577>
7. K.M. Kinnunen, M.R.J. Palosaari, I.J. Maasilta, *J. Appl. Phys.* **112**, 034515 (2012). <https://doi.org/10.1063/1.4745908>
8. M.R.J. Palosaari et al., *J. Low Temp. Phys.* **167**, 129 (2012). <https://doi.org/10.1007/s1090-012-0471-4>
9. M.A. Lindeman et al., *Rev. Sci. Instrum.* **78**, 043105 (2007). <https://doi.org/10.1063/1.2723066>
10. D.J. Goldie et al., *J. Appl. Phys.* **105**, 074512 (2009). <https://doi.org/10.1063/1.3097396>
11. E. Taralli et al., *Supercond. Sci. Technol.* **23**, 105012 (2010). <https://doi.org/10.1088/0953-2048/23/10/105012>
12. N.A. Wakeham et al., *J. Appl. Phys.* **125**, 164503 (2019). <https://doi.org/10.1063/1.5086045>
13. I.J. Maasilta, *AIP Adv.* **2**, 042110 (2012). <https://doi.org/10.1063/1.4759111>
14. R. Storn, K. Price, *J. Glob. Optim.* **11**, 314–359 (1997). <https://doi.org/10.1023/A:1008202821328>
15. W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, *Numerical Recipes the Art of Scientific Computing*, 3rd edn. (Cambridge University Press, Cambridge, 2007)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.