



Original software publication

Total Mapping Toolbox (TOMATO): An open source library for cardiac magnetic resonance parametric mapping



Konrad Werys^{a,*}, Iulius Dragonu^b, Qiang Zhang^a, Iulia Popescu^a, Evan Hann^a,
Henrike Puchta^a, Agata Kubik^c, Dogan Polat^d, Cody Wu^a, Niall O. Moon^a,
Ahmet Barutcu^a, Vanessa M. Ferreira^a, Stefan K. Piechnik^a

^a Oxford Centre for Clinical Magnetic Resonance Research, John Radcliffe Hospital, Oxford, UK

^b Siemens Healthcare Ltd, Frimley, UK

^c Institute of Cardiology, Warsaw, Poland

^d University of Texas Southwestern Medical Center, Department of Radiology, Dallas, TX, USA

ARTICLE INFO

Article history:

Received 18 July 2019

Received in revised form 14 November 2019

Accepted 26 November 2019

Keywords:

Optimisation

Model fitting

Parametric mapping

Longitudinal relaxation time

T1

Magnetic resonance imaging

ABSTRACT

TOMATO (Total Mapping Toolbox) is a C++ library for the calculation of parametric maps in cardiac magnetic resonance imaging (MRI). As an open source project, TOMATO allows transparent and standardised cardiac longitudinal relaxation time (T1) mapping in clinical applications. With C++ implementation, TOMATO can easily interface and translate between research software environments and commercial vendors' closed-source C++ environments on scanners as well as post-processing software. To complement the core library implementation, a ready-to-use command line tool has been provided.

© 2019 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Code metadata

Current code version	v0.4
Permanent link to code/repository used of this code version	https://github.com/ElsevierSoftwareX/SOFTX_2019_219
Legal Code Licence	MIT
Code versioning system used	git
Software code languages, tools, and services used	TOMATO library: C++ TOMATO executable: C++, ITK, libyaml
Compilation requirements, operating environments & dependencies	TOMATO library: none TOMATO executable: C++, ITK, libyaml
If available Link to developer documentation/manual	https://mrkonrad.github.io/TOMATO/html https://mrkonrad.github.io/tomato_docs
Support email for questions	konrad.werys@cardiov.ox.ac.uk

Software metadata

Current software version	v0.4
Permanent link to executables of this version	https://github.com/MRKonrad/TOMATO/releases
Legal Software Licence	MIT
Computing platforms/Operating Systems	TOMATO library: Windows/MacOS/Linux TOMATO executable: Windows/MacOS/Linux
Installation requirements & dependencies	TOMATO library: CMake TOMATO executable: CMake, ITK, libyaml
If available, link to user manual—if formally published include a reference to the publication in the reference list	https://github.com/MRKonrad/TOMATO https://mrkonrad.github.io/tomato_docs
Support email for questions	konrad.werys@cardiov.ox.ac.uk

* Corresponding author.

E-mail address: konrad.werys@cardiov.ox.ac.uk (K. Werys).

1. Motivation and significance

Magnetic Resonance Imaging (MRI) is a widely used medical diagnostic technology for non-invasive high-resolution visualisation of anatomy, function and tissue characterisation. Longitudinal (T1) and transverse (T2) relaxation times are fundamental magnetic resonance properties that allow distinguishing between various tissues and states of pathology [1]. Over the past 10 years in the field of cardiovascular magnetic resonance (CMR) imaging, T1 and T2-mapping techniques have matured as advanced quantitative tissue characterisation techniques to provide novel, non-invasive insights into a wide range of heart diseases [2,3]. Recently, CMR mapping was deemed by the European Society of Cardiology Heart Failure Association as one of six most innovative technologies for the non-invasive evaluation of patients with heart failure [4]. Quantitative T1- and T2-mapping have also recently been incorporated into international guidelines for detecting myocardial inflammation non-invasively using CMR [5].

In particular, cardiac T1-mapping has rapidly accumulated a large body of evidence for various clinical applications [6], and is currently used in a number of large-scale multi-centre studies, such as the UK Biobank imaging component ($n = 100,000$) [7], the Hypertrophic Cardiomyopathy Registry HCMR ($n > 2770$) [8], Canadian alliance for healthy hearts and minds cohort study ($n \sim 9700$) [9] and Hamburg City Health cohort study [10]. However, the measured T1 depends on multiple factors, including hardware and software acquisition parameters, and image post-processing methods [2,3]. This can result in poor consistency between normal range values reported from different centres, which use different approaches to measure T1 [11].

T1-map post-processing faces the issue of a lack of transparency in the choice and implementation of T1 calculation algorithms. Even though the algorithms are described in the literature, different implementations in research environments like MATLAB and closed commercial implementations on different scanners and post-processing software can contribute to discrepancies in the results. The development of an open-source library for mapping calculation can increase the transparency of the reconstruction algorithm choices and pave the way towards achieving better consistency and standardisation within the field.

To our knowledge, the following open-source T1 calculation tools have been published: MRMap (developed in IDL) [12] qMR-Lab (Matlab) [13] and QUIT (C++) [14]. MRMap and qMRLab use high level interactive language environments that require paid subscriptions. As a consequence, they are excellent educational tools for use in academia, but are limited by access to IDL or Matlab, which is especially problematic in the environment of an MRI scanner. QUIT focuses on the implementation of T1 calculation methods, used to image brains, whereas TOMATO focuses on methods used to image hearts.

Based on our experience over the past 10 years in the development and validation of the ShMOLLI T1-mapping technique [15], we present the Total Mapping Toolbox (TOMATO), an open source C++ library for cardiac T1-mapping reconstruction. TOMATO is implemented in C++, which is the industry standard used by most of the scanner vendors and post-processing companies in the cardiac MRI field. This choice allows for direct and easy translation between a prototype algorithm and a vendor environment. Additionally, use of C++ allows fast calculation compatible with a broad range of numerical libraries such as VNL [vxl.github.io] and GSL [gnu.org/software/gsl].

We have designed TOMATO to be flexible and easy to extend, to facilitate its use in clinical applications and trials. TOMATO promotes consistency by opening the “black box” of T1 calculation algorithms that can be used by different vendors. In addition, TOMATO enables translation from research prototyping to production with a transparent and fully traceable reference to the open source reference.

2. Software description

2.1. Software requirements specification and dependencies

The design of **TOMATO library (TomatoLib)** is based on the following set of requirements:

1. ShMOLLI calculation in accordance to the existing data
2. Ability to choose different calculation and fitting algorithms, and easily switch between them
3. Ability to easily add new algorithms and extend the existing ones
4. Fast calculation
5. Compatibility with the major operating systems (Windows, Linux, MacOS).

The development process was focused on fulfilling requirements 1–3. In its base form TomatoLib can be compiled as a standalone code without any non-standard library dependencies. For the backward compatibility with different scanner systems, the library was developed in C++98 (ISO/IEC 14882:1998 [isocpp.org]). Possibility of switching between fitting procedures allows a user finding the algorithm, which is the most suitable for their needs. For example, one that is the fastest or one that can be easily modified. Different fitting procedures can be used in TomatoLib, including algorithms from numerical libraries VNL [vxl.github.io] (included in ITK [itk.org]) and LMfit [https://jugit.fz-juelich.de/mlz/lmfit]. During TomatoLib build step, a user can select one or more of these libraries and they will be automatically downloaded and included in the project (requirement 2). Universal build system CMake [cmake.org] was used to manage the build process (requirement 5).

TomatoLib library allows single-, and multi-pixel map reconstructions in a form intended to be readily embedded in an external calculation tool. To demonstrate this, we have designed a **TOMATO executable (TomatoExe)** with the following requirements:

1. Direct handling of DICOM files as input and output
2. Ability to easily configure the input files and processing algorithms
3. Compatibility with the main operating systems (Windows, Linux, MacOS).

TomatoExe depends on TomatoLib for calculations, ITK library [itk.org] for input and output DICOM operations (requirement 1.) and libyaml [github.com/yaml/libyaml] library for the human-readable configuration files written in YAML (requirement 2.). The YAML configuration files include information about the location of input files and output directories and allow configuration of the calculation algorithm. They are easily configurable and can be used as an interface to other environments.

2.2. Software architecture

To obtain a T1-map, a series of images with different acquisition parameters are acquired. The images are then combined into parametric maps, by fitting a T1 relaxation model for each pixel, predetermined by the model defined by magnetic resonance physics and presented in the equation in Fig. 1.

In TomatoLib (Fig. 2), the user has the option to choose the fitting procedure (Fitter class), the model (FunctionsT1 class), and how the data is processed before and after the fitting (Calculator class). Additionally, we provide a way to calculate the fitting starting point (StartPointCalculator class) and a way to recover the sign of the data from magnitude and phase data (SignCalculator class).

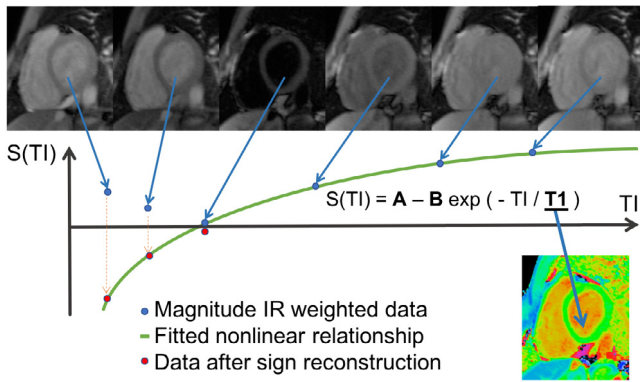


Fig. 1. Concept of generating T1-maps. A series of images with different acquisition parameters (MR weighting) is required to fit parameters to known model function. Parameters of the model are presented as maps and can be used in clinical diagnosis.

To provide a platform to choose and configure a calculation, as well as the fitting algorithm, the five core classes (Fitter, FunctionsT1, Calculator, StartPointCalculator and SignCalculator) have been implemented as abstract classes. The concrete implementation code is contained within the child classes. Such modularisation enables extending the library, by adding new child classes to the five core classes.

2.2.1. Calculator classes

Calculator class is managing the calculation process from a single pixel point of view. The purpose of Calculator class is to calculate (calculate() method) the parameters used in the map, based on the input signal. It can be configured by providing pointer to other TomatoLib objects like Fitter or FunctionsT1. Examples of solid implementations of this class include:

- **CalculatorT1Shmolli** – ShMOLLI implementation with the conditional fitting algorithm [15]. It has a fixed number of input samples. It can be configured to give exactly the same results as [15] or a user may experiment with different configurations of fitters, sign and start point calculators

- **CalculatorT1Molli** – calculation using MOLLI algorithm [16]. A user can choose which fitting algorithm should be used for example Levenberg–Marquardt as in [16] or Amoeba as in [15,17] or which sign reconstruction method [16,18].

2.2.2. Fitter classes

The purpose of Fitter class is to perform fitting (performFitting() method) of a given model (FunctionsT1 class) parameters to given input samples to output fitted model parameters. A Fitter code can be implemented from scratch, like in AmoebaNr class, where a Nelder–Mead algorithm code is based on Numerical Recipes book [19]. Alternatively, a fitter code can be a wrapper over an external library optimiser, like in AmoebaVnl class, where a Nelder–Mead algorithm from VNL library is used.

2.2.3. FunctionsT1 classes

The purpose of the FunctionsT1 class is to provide the model (calcModelValue(T *parameters, T time) method), corresponding cost function of the fitting problem (calcCostValue(T *parameters) method) and corresponding least squares function calculation (calcLSResiduals(T *parameters, T *residuals) method). Additionally, derivative calculation for gradient based fitters can be implemented. Some fitters need special formulation of these functions, so following the concept of the adapter design pattern [20], we provide adapter classes for example: FunctionsT1AdapterVnlCost and FunctionsT1AdapterVnlLeastSquares.

2.2.4. SignCalculator classes

Typically, the input data is the magnitude of the signal (blue points on Fig. 1). Without more information the sign of the signal (red points) can be calculated by trying a different combination of the signal sign as proposed in [16]. If additional information is provided, like the phase of the signal, the sign of the signal can be calculated and the calculation time can be reduced [15,18]. The purpose of SignCalculator class is to calculate (calculateSign() method) the sign of the signal based on magnitude and phase data.

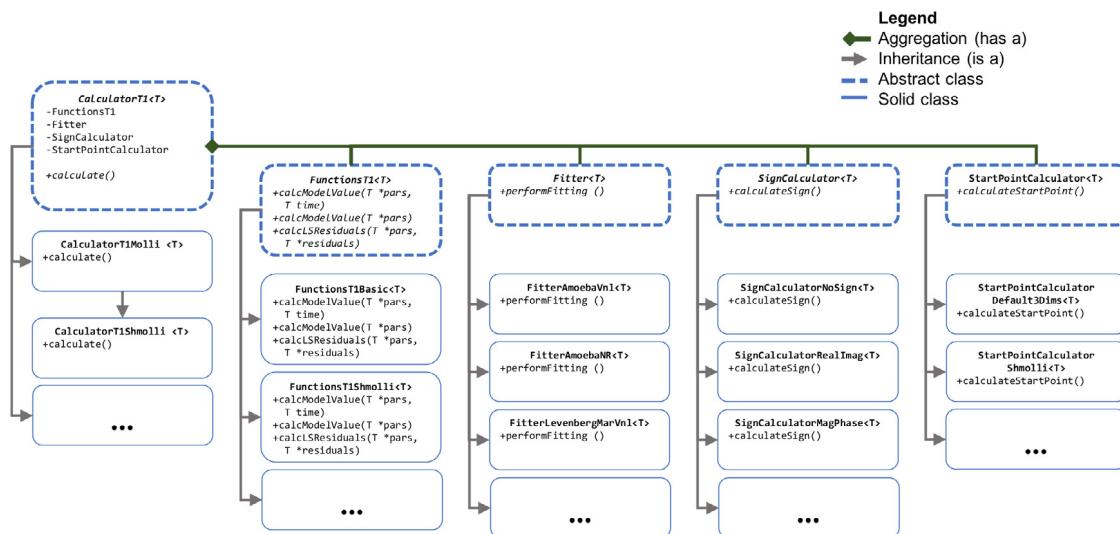


Fig. 2. Simplified class diagram of the TOMATO library. Abstract classes (dashed borders) provide abstract methods (italic) that have to be implemented in the solid classes (solid borders), as they will be called during the algorithm execution. CalculatorT1 class has object of classes FunctionsT1, Fitter, SignCalculator and StartPointCalculator. The users can choose any combination of the solid classes when configuring the algorithm, or they can provide their own implementation of these modules as indicated by the three horizontal dots.

2.2.5. StartPointCalculator classes

The purpose of this class is to calculate an initial estimate of the output model parameters that is used by the Fitter as a starting point, in order to speed up and improve the robustness of the subsequent nonlinear optimisation.

2.2.6. ImageCalculator classes

The purpose of this class is to calculate the T1-map from a multipixel data provided in an input array. This is where parallel CPU calculations are enabled for the systems that support multithreading support library. Its use is optional and it can be replaced by specialised thread handling system like ITK Filters as in TomatoExe.

2.3. Sample code snippets

Below we present a sample function for one-pixel calculation using the ShMOLLI algorithm. Other examples are provided in the code.

```
Ox::CalculatorT1Results<double> CalculateShmolli(
    int nSamples, double* sigMag, double* sigPha, double* invTimes) {

    // initialise the necessary objects
    Ox::CalculatorT1Shmolli<double> calculatorT1Shmolli;
    Ox::FunctionsT1Shmolli<double> functionsObject;
    Ox::FitterAmoebaNr2<double> fitter;
    Ox::SignCalculatorShmolli<double> signCalculator;
    Ox::StartPointCalculatorShmolli<double> startPointCalculator;

    // configure
    calculatorT1Shmolli.setFunctionsT1(&functionsObject);
    calculatorT1Shmolli.setFitter(&fitter);
    calculatorT1Shmolli.setSignCalculator(&signCalculator);
    calculatorT1Shmolli.setStartPointCalculator(&startPointCalculator);

    // set the data
    calculatorT1Shmolli.setNSamples(nSamples);
    calculatorT1Shmolli.setInvTimes(invTimes);
    calculatorT1Shmolli.setSigMag(sigMag);
    calculatorT1Shmolli.setSigPha(sigPha);

    calculatorT1Shmolli.calculate();

    return calculatorT1Shmolli.GetResults();
}
```

3. Illustrative examples

In Fig. 3, we present an example of TomatoExe output with different Fitting algorithms. This approach can be used to optimise the calculation parameters in order to shorten the calculation time, as we have shown in our previous work [21].

4. Impact

TOMATO originated as part of the pipeline for T1-map analysis in the Hypertrophic Cardiomyopathy Registry (HCMR) [8], which included over 2770 patient datasets using ShMOLLI T1-mapping. TOMATO provides a consistent and transparent system for T1-map image post-processing, ensuring that the same T1 calculation method was used among all of the cases in a multicentre study like HCMR [8] or UK Biobank [7]. In the future, TOMATO may be used for standardised post-processing for mapping methods between various hardware and software manufacturers, which may eliminate one of the factors of method variability across study sites [22].

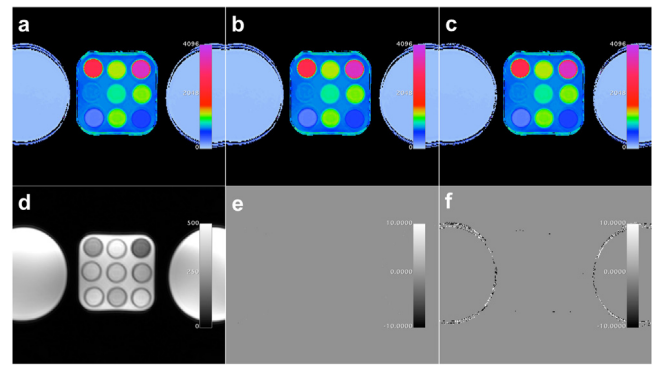


Fig. 3. TomatoExe calculations. (a) Original DICOM T1-map produced on the scanner as a reference image. (b) T1-map calculated using TomatoExe with default ShMOLLI calculation parameters. (c) T1-map calculated using TomatoExe with Levenberg–Marquardt fitting algorithm. (d) Raw magnitude images used for T1-maps calculation. (e) Difference between a. and b. (f) Difference between a. and c. Note that a. and b. are perfect match as seen in e., but a. and c. give different results in the pixels close to the edge as shown in f.

Additionally, in TOMATO, we reduced the time of offline T1-map calculation from minutes (in our MATLAB implementation) to seconds, similar performance to vendor's proprietary implementation. This degree of acceleration allowed our centre the unique capability to efficiently process large-scale datasets offline, like the 33,000 T1-maps in HCMR [8] and the initial 5000 T1-maps in UK Biobank [7] (target 100,000 T1-maps). TOMATO facilitates further improvement of the T1-mapping algorithms and has already been used in the development of a quality assurance protocol within our centre, advancing towards standardisation and widespread clinical applicability of CMR mapping techniques.

Currently, we are planning to integrate TOMATO into the T1-mapping reconstruction pipeline of one of the largest MR scanner vendors, into the software of one of the post-processing commercial companies, and into the qMRLab [13]. The open source character of TOMATO allows for better standardisation between different scanner vendors, through traceable versions and method choices. With additional implementation of T2, T2*, T1rho parametric equations, the TOMATO library may be used for these multiparametric tissue characterisation techniques, towards standardisation of quantitative CMR imaging biomarkers.

5. Conclusions

Total Mapping Toolbox (TOMATO) is an open source, flexible and fast parametric mapping calculation library developed in the C++ language, with practical implementations as a command line calculation tool. provides not only a fast, but more consistent and objective way of comparing different reconstruction options, with traceability. The TOMATO library addresses the need of better standardisation and transparency in MRI data processing, fulfilling one of the important quality control criteria in high-quality research, clinical decision-making, trials and industry standards. With future implementation of T1, T2, T2*, T1rho models, the TOMATO library aims to ultimately widen the application and standardisation of these quantitative CMR imaging biomarkers to address real-world clinical needs.

Declaration of competing interest

- Stefan Piechnik has rights US patent US20120078084A1: Stefan K. Piechnik, Matthew D. Robson, Andreas Greiser SYSTEMS AND METHODS FOR SHORTENED LOOK LOCKER

INVERSION RECOVERY (Sh-MOLLI) CARDIAC GATED MAPPING OF T1, total income since 2010 less than 10000 USD. Patent owned by University of Oxford. All rights transferred to Siemens Healthcare LTD

- Iulius Dragonu is an employee of Siemens Healthcare LTD, Frimley, UK
- Konrad Werys is an employee of Circle Cardiovascular Imaging Inc, Canada from 1st August 2019

Other authors do not have known conflict of interests.

Acknowledgements

This research is supported by the British Heart Foundation (grant number PG/15/71/31731). SN, VMF and SKP acknowledge support from the Oxford NIHR Biomedical Research Centre and from the Oxford British Heart Foundation Centre of Research Excellence. VMF is supported by the British Heart Foundation. Part of the code is protected by the US patent US20120078084A1.

There are two Tomato version available: TomatoOpenSource compiled with publicly available code and TomatoFull containing additionally private code used for AmoebaPrivateNr2 fitting algorithm using Nelder-Mead algorithm based on Numerical Recipes. Due to Numerical Recipes licence limitations this code cannot be made publicly available.

References

- [1] Margaret Cheng H-L, Stikov N, Ghugre NR, Wright GA. Practical medical applications of quantitative MR relaxometry. *J Magn Reson Imaging* 2012;36:805–24. <http://dx.doi.org/10.1002/jmri.23718>.
- [2] Moon JC, Messroghli DR, Kellman P, Piechnik SK, Robson MD, Ugander M, et al. Myocardial t1 mapping and extracellular volume quantification: A Society for Cardiovascular Magnetic Resonance (SCMR) and CMR Working Group of the European Society of Cardiology consensus statement. *J Cardiovasc Magn Reson* 2013. <http://dx.doi.org/10.1186/1532-429X-15-92>.
- [3] Messroghli DR, Moon JC, Ferreira VM, Grosse-Wortmann L, He T, Kellman P, et al. Clinical recommendations for cardiovascular magnetic resonance mapping of T1, T2, T2 and extracellular volume: A consensus statement by the Society for Cardiovascular Magnetic Resonance (SCMR) endorsed by the European Association for Cardiovascular Imaging. *J Cardiovasc Magn Reson* 2017;19:1–24. <http://dx.doi.org/10.1186/s12968-017-0389-8>.
- [4] Čelutkienė J, Plymen CM, Flachskampf FA, de Boer RA, Grapsa J, Manka R, et al. Innovative imaging methods in heart failure: a shifting paradigm in cardiac assessment. Position statement on behalf of the Heart Failure Association of the European Society of Cardiology. *Eur J Heart Fail* 2018;20:1615–33. <http://dx.doi.org/10.1002/ehf.1330>.
- [5] Ferreira VM, Schulz-Menger J, Holmvang G, Kramer CM, Carbone I, Sechtem U, et al. Cardiovascular magnetic resonance in nonischemic myocardial inflammation: Expert recommendations. *J Am Coll Cardiol* 2018;72:3158–76. <http://dx.doi.org/10.1016/j.jacc.2018.09.072>.
- [6] Liu JM, Liu A, Leal J, McMillan F, Francis J, Greiser A, et al. Measurement of myocardial native T1 in cardiovascular diseases and norm in 1291 subjects. *J Cardiovasc Magn Reson* 2017;19:1–10. <http://dx.doi.org/10.1186/s12968-017-0386-y>.
- [7] Matthews PM, Piechnik S, Weale P, Young AA, Zemrak F, Hudson S, et al. UK Biobank's cardiovascular magnetic resonance protocol. *J Cardiovasc Magn Reson* 2016;18. <http://dx.doi.org/10.1186/s12968-016-0227-4>.
- [8] Jerosch-Herold M, Geller N, Wu P, Desvigne-Nickens P, Appelbaum E, Neubauer S, et al. Hypertrophic Cardiomyopathy Registry: The rationale and design of an international, observational study of hypertrophic cardiomyopathy. *Am Heart J* 2015;170:223–30. <http://dx.doi.org/10.1016/j.ahj.2015.05.013>.
- [9] Anand SS, Tu JV, Awadalla P, Black S, Boileau C, Busseuil D, et al. Rationale, design, and methods for Canadian alliance for healthy hearts and minds cohort study (CAHHM) – A Pan Canadian cohort study. *BMC Public Health* 2016. <http://dx.doi.org/10.1186/s12889-016-3310-8>.
- [10] Bohnen S, Avanesov M, Jagodzinski A, Schnabel RB, Zeller T, Karakas M, et al. Cardiovascular magnetic resonance imaging in the prospective, population-based, Hamburg City Health cohort study: Objectives and design. *J Cardiovasc Magn Reson* 2018. <http://dx.doi.org/10.1186/s12968-018-0490-7>.
- [11] Popescu I. Standardization of MOLLI-based myocardial T1 mapping techniques based on published reports. SCMR 22nd Annu. Sci. Sess., Seattle, USA: 2019.
- [12] Messroghli DR, Rudolph A, Abdel-Aty H, Wassmuth R, Kühne T, Dietz R, et al. An open-source software tool for the generation of relaxation time maps in magnetic resonance imaging. 2010.
- [13] Cabana JF, Gu Y, Boudreau M, Levesque IR, Atchia Y, Sled JG, et al. Quantitative magnetization transfer imaging made easy with qMTLab: Software for data simulation, analysis, and visualization. *Concepts Magn Reson A Bridg Educ Res* 2015;44A:263–77. <http://dx.doi.org/10.1002/cmr.a.21357>.
- [14] Wood TC. QUIT: Quantitative imaging tools. *J Open Source Softw* 2018;3:656. <http://dx.doi.org/10.21105/joss.00656>.
- [15] Piechnik SK, Ferreira VM, Dall'Armellina E, Cochlin LE, Greiser A, Neubauer S, et al. Shortened Modified Look-Locker Inversion recovery (ShMOLLI) for clinical myocardial T1-mapping at 1.5 and 3 T within a 9 heartbeat breathhold. *J Cardiovasc Magn Reson* 2010;12:69. <http://dx.doi.org/10.1186/1532-429X-12-69>.
- [16] Messroghli DR, Radjenovic A, Kozerke S, Higgins DM, Sivananthan MU, Ridgway JP. Modified look-locker inversion recovery (MOLLI) for high-resolution T1 mapping of the heart. *Magn Reson Med* 2004;52:141–6. <http://dx.doi.org/10.1002/mrm.20110>.
- [17] Xue H, Shah S, Greiser A, Guetter C, Littmann A, Jolly M-P, et al. Motion correction for myocardial T1 mapping using image registration with synthetic image estimation. *Magn Reson Med* 2012;67:1644–55. <http://dx.doi.org/10.1002/mrm.23153>.
- [18] Xue H, Arai AE, Jolly M-P, Guehring J, Kellman P, Greiser A, et al. Phase-sensitive inversion recovery for myocardial T1 mapping with motion correction and parametric fitting. *Magn Reson Med* 2012;69:1408–20. <http://dx.doi.org/10.1002/mrm.24385>.
- [19] Press WH, Teukolsky SA, Vetterling WT, Flannery BP. *Numerical recipes : The art of scientific computing*. Cambridge University Press; 2007.
- [20] Vlissides J, Johnson R, Helm R, Gamma E. *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley; 2001.
- [21] Werys K, Moon N, Kwong RY, Jerosch-Herold M, Kramer CM, Neubauer S, et al. Accelerating reconstruction of T1-maps without sacrificing consistency for clinical applications. SCMR 22nd Annu. Sci. Sess., Bellevue: 2019.
- [22] Stikov N, Boudreau M, Levesque IR, Tardif CL, Barral JK, Pike GB. On the accuracy of T1 mapping: Searching for common ground. *Magn Reson Med* 2015;73:514–22. <http://dx.doi.org/10.1002/mrm.25135>.