11-2017

Reproducible Research for Computing in Science & Engineering

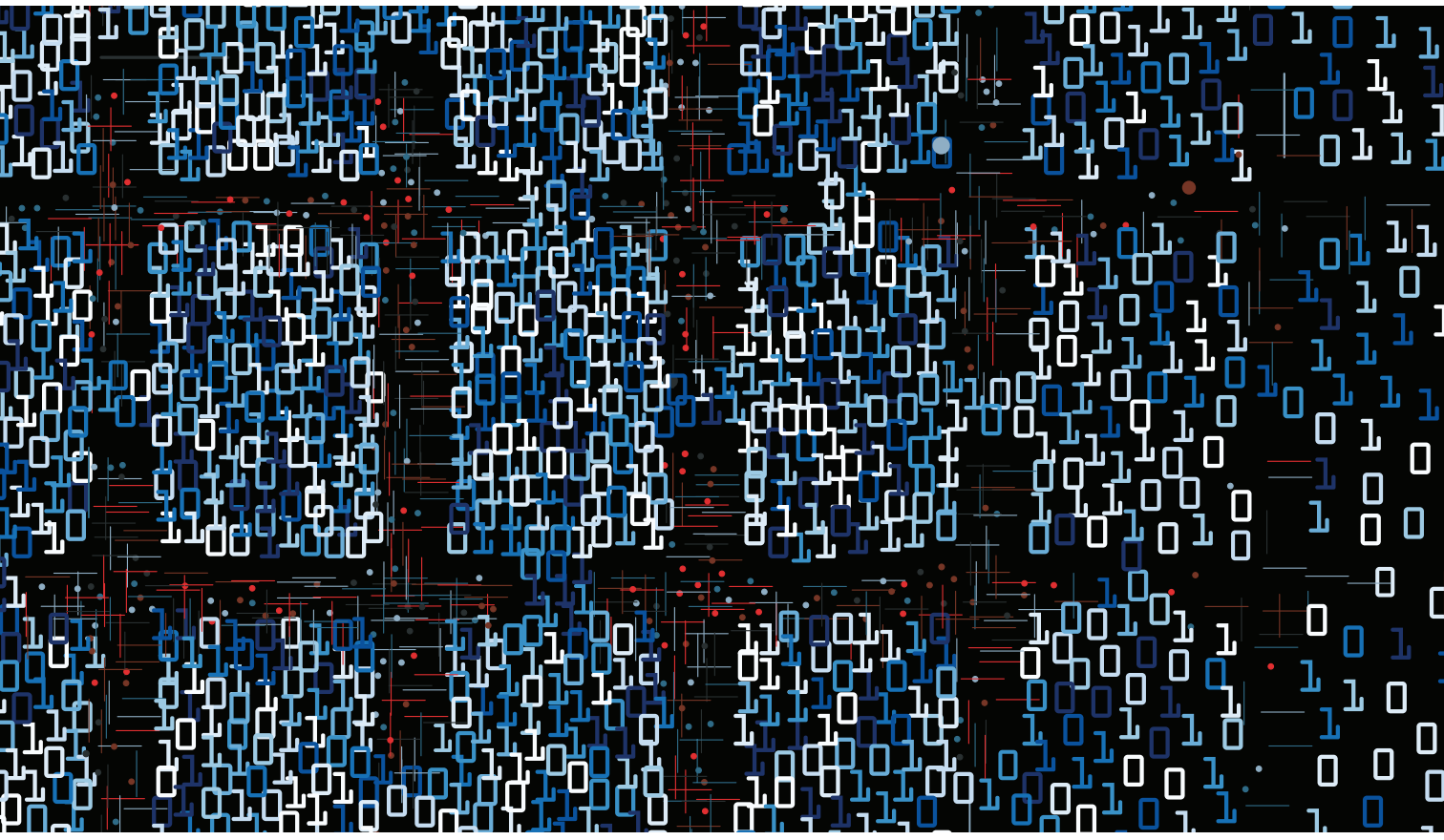
Lorena A. Barba
George Washington University

George K. Thiruvathukal
Loyola University Chicago, gkt@cs.luc.edu

Recommended Citation

Lorena A. Barba and George K. Thiruvathukal, "Reproducible Research for Computing in Science & Engineering", Computing in Science & Engineering, vol. 19, no. 6, pp. 85-87, November/December 2017, doi:10.1109/MCSE.2017.3971172

Editors: Lorena A. Barba, labarba@gwu.edu | George K. Thiruvathukal, gkt@cisemagazine.org



Reproducible Research for *Computing in Science & Engineering*

Lorena A. Barba | George Washington University
George K. Thiruvathukal | Loyola University Chicago

In the final issue of 2000, *CiSE* published an article entitled "Making Scientific Computations Reproducible," by Matthias Schwab, Martin Karrenbach, and Jon Claerbout,[1] in which the authors described their system for reproducing the scientific computations coming out of their laboratory. Claerbout, a professor of geophysics at Stanford University, had been using and perfecting the system for more than a decade and even required his doctoral students to submit fully reproducible theses: specifically, every figure in the thesis should be reproducible programmatically, with a single command. To achieve it, the group developed standardized commands, a filing system for data and computer files, and a process for testing and reusing software. To this day, researchers around the world are still striving to measure up to Claerbout's standards.

Eight years later, Claerbout co-edited with Sergey Fomel a theme issue of *CiSE* dedicated to reproducible research.[2] Four articles composed this dedicated portion of the magazine, covering computational research in harmonic analysis, wave propagation, and bioinformatics, along with questions about copyright and licensing of research materials. The authors are a who's who of the reproducible research field: David Donoho, Randall LeVeque, Roger Peng, and Victoria Stodden, among others. A year later, the magazine's News section published a report of the Yale Law School's roundtable on data and code sharing,[3] which contained a set of recommendations inspired by the theme of transparency via open code and data. Another theme issue in July/August 2012 focused on reproducible research, this time emphasizing tools and strategies.

CiSE has thus published several influential pieces in the scholarly conversation about reproducible research. Now we want to make the topic a more permanent fixture in our magazine by launching a Reproducible Research Track.

CiSE has published several influential pieces in the scholarly conversation about reproducible research. Now we want to make the topic a more permanent fixture in our magazine by launching a Reproducible Research Track.

Seeking Three Contribution Types

The new *CiSE* track seeks contributions of three types: long-form articles reporting on original research that followed rigorous reproducibility practices (these will be peer reviewed using the normal process for general *CiSE* submissions); short case studies reporting experiences in reproducing or replicating past computational work (successes or failures—we especially invite researchers in industry or national labs to contribute case studies); and briefs on libraries, tools, and techniques.

Peer-reviewed articles in the RR track will adhere to special requirements. First, openness: any underlying code used to produce the results in the article will need to be free and open source (hereafter, open source), all data will also be openly available, and the manuscript itself will be deposited in a preprint server before submission. Second, reproducible computations: any computational results will need to be reproducible by executing the author-provided workflow (in the form of input data, analysis code, plotting scripts, and so on). Authors will prepare a reproducibility package consisting of a file set that includes all the parts needed to reproduce each figure or result. Finally, articles will include an appendix reporting on the software engineering and data management practices followed by the authors and their collaborators.

Open Source

Openness and transparency became inseparable from reproducibility after the Yale Law roundtable recommendations, which are unequivocal about the need to use open licenses and nonproprietary formats. Certainly, only if the authors made all code and data available could readers hope to reproduce their exact results. But open source software is not just about access. In a 2015 interview with Robert Talbert, Lorena Barba described open source software as a "human invention of tremendous impact" (https://medium.com/@roberttalbert/4-1-interview-lorena-barba-bfb3bd70a6a).

While copyright laws want to control how creative works are used, open source licenses offer a freeing workaround: they pre-authorize anyone who might want to use the copyrighted work. Freer licenses permit wider uses (such as commercial) and the creation of derivative works. Since building from the work of others is essential in science, with more and more scientific findings relying on software, open source licensing is a must. Accordingly, the RR track will require software associated with an article to be under an OSI-approved license (http://opensource.org) and data to be under a Creative Commons license or dedication. Authors will also need to deposit a preprint of their manuscript in an archival-quality service that provides a unique identifier (such as a DOI), as found in the arXiv, engrXiv, bioRxiv, figshare, OSF.io, or Zenodo systems.

Reproducible

Reproducible computations, however, demand more than open code and data. While access and liberal licensing are a prerequisite, on top of this minimum we require structured organization of the materials and detailed documentation of the processes. Barba uses the term "reproducibility package" for a file bundle uploaded to a data repository that contains the data, analysis code, plotting scripts, and figures corresponding to a particular result.[4] Using an archival service and obtaining a DOI for the pack assures that the materials are not only discoverable but also citable. She advocated for this approach among the pledges in her 2012 "Reproducibility PI Manifesto" (doi:10.6084/m9.figshare.104539.v1).

We'll require research articles in the RR track to adhere to the practice of preparing reproducibility packages for primary results. A designated "reproducibility reviewer" will be in charge of testing their completeness, correctness, and usability. During the review of digital objects, the nonanonymous reproducibility reviewer may interact with the authors as needed. And for their service, reproducibility reviewers will be recognized in the article's acknowledgments section.

Software Engineering

Scientific software differs in essential ways from industrial software, and many software engineering practices aren't popular or might not even transfer well to research situations.[5] Still, embracing software engineering can have many positive effects on a research project. In particular, source code version control, issue tracking, and documentation are vital for managing and sustaining complex software projects. Jeffrey Carver and George Thiruvathukal

point out that incorporating software engineering is best done early: late in a project, it will not only be difficult but more time-consuming and costly (doi:10.6084/m9.figshare.830442.v2).

For this reason, we discourage the inclination to "release the code after acceptance," and instead promote the practice of developing code in the open, under a version control system. Other practices to aspire to include specifying requirements, code testing, writing and maintaining documentation, tracking software metrics, conducting code review, and using continuous integration. Articles submitted to the RR track won't be required to follow any of these practices—given that one process doesn't fit all software projects—but they will be asked to include an appendix reporting on what software engineering practices are, indeed, followed in the project. The designated reproducibility reviewer will evaluate these materials, and we'll use the assessment to inform the decision to accept the article. We also encourage authors to submit their code base to the *Journal of Open Source Software* (http://joss.theoj.org), which conducts peer review of software artifacts, for an extra badge of quality on the work.

Case Studies and Briefs

The RR track also invites brief submissions reporting case studies in reproducing or replicating published computational research (successful or not). We differentiate the terms of art as follows: reproducibility refers to executing the author-provided code with the data and getting the same results; replication refers to an independent study, using different methods, collecting new data, and arriving at the same findings.[6] Either type of work has trouble getting published; journals as a rule look for novelty when deciding on publication. (A notable exception is the *ReScience Journal*, http://rescience.github.io, which publishes full replications that are also newly reproducible.) We wish to provide a venue for short reports describing the following: what went wrong when attempting to use author-provided code and data; what information was missing that was needed to reproduce the computations; what skills were required; what effort was required and how one can measure it; what unexpected challenges arise when computing the same problem with different methods; what the limits of reproducible research are; and what we can do to improve the level of reproducibility in the published literature.

Technical briefs on libraries, tools, and techniques will serve the goal of supporting computational scientists and engineers in their pursuit of reproducible research. The RR track invites submissions describing tools designed to improve workflows, automate protocols, and facilitate documentation. Strategies that have worked to reduce complexity or improve the skills of the research team are also of interest. These abridged features will experience a fast-track review process by the department editors.

Open Questions

We're determined to make the *CiSE* RR track serve as a prototype for new norms in scholarly publishing of computational research. From our viewpoint, all steps to increase transparency have the potential to promote trust and contribute to reproducible research. Thus, we'll study how the track might implement double-open peer review, for example. We'll also look for ways to recognize the contributions of reviewers, and whether their critiques could be published alongside the article (assigned a DOI and made citable). Many technical and social questions remain open in this regard: subsidiary issues such as limitations of the manuscript submission system, and overriding issues such as protecting early career researchers from bias. The editors of the RR track will explore these questions with the rest of the editorial board, and share their ideas, assessments, and intentions via op-ed columns. Granted, many of the ideas of this department will be a challenge for authors, readers, and the editors (ourselves) alike. Yet, by acknowledging the history of *CiSE* being an early home for works on reproducible research, we hope to extend this tradition and make an open call to challenge the state of the art by including reproducibility in everything we do. ■

References

1. M. Schwab, M. Karrenbach, and J. Claerbout, "Making Scientific Computations Reproducible," *Computing in Science & Eng.*, vol. 2, no. 6, 2000, pp. 61–67; doi:10.1109/5992.881708.
2. S. Fomel and J.F. Claerbout, "Guest Editors' Introduction: Reproducible Research," *Computing in Science & Eng.*, vol. 11, no. 1, 2009, pp. 5–7; doi:10.1109/MCSE.2009.14.
3. "Reproducible Research," *Computing in Science & Eng.*, vol. 12, no. 5, 2010, pp. 8–13; doi:10.1109/mcse.2010.113.
4. L.A. Barba, "The Hard Road to Reproducibility," *Science*, vol. 354, no. 6308, 2016, p. 142; doi:10.1126/science.354.6308.142.
5. J.C. Carver et al., eds., *Software Engineering for Science*, Chapman and Hall/CRC Press, 2016.
6. R.D. Peng, "Reproducible Research in Computational Science," *Science*, vol. 334, no. 6060, 2011, pp. 1226–1227; doi:10.1126/science.1213847.