

The PENCIL CODE Newsletter

Issue 2020/3

December 1, 2020, Revision: 1.56

Contents

1	PENCIL CODE office hours	1
2	19 years of PENCIL CODE	1
3	Election of the PCSC	1
4	JOSS paper	2
5	Code of Conduct	2
6	Letters to the editor	2
7	Training opportunities	3
8	Employment opportunities	3
9	Code developments	3
9.1	Infrastructure and postprocessing	4
9.2	Mail from the code	4
9.3	Automatic snapshot repair	4
9.4	Abandoning source linking	4
10	Papers since September	4

1 PENCIL CODE office hours

During the last PENCIL CODE User Meeting, it was suggested to have PENCIL CODE office hours, where users can discuss with experts on how to do this or that. This is not just a tutorial-type activity for the users, but also an opportunity for the experts to get together on a more frequent schedule. The PENCIL CODE Steering Committee (PCSC) decided in its last meeting of November 27, 2020 that our first PENCIL CODE office hours will be held on **Friday, 8 January, 2021 at 13:00 CET**. Please be sure to mark this date in your calendar. See you then on <https://stockholm.university.zoom.us/j/6415995185>.

2 19 years of PENCIL CODE

We now have 587,832 lines of f90 code. Every year, the code gains about 35,000 new lines; see Figure 1.

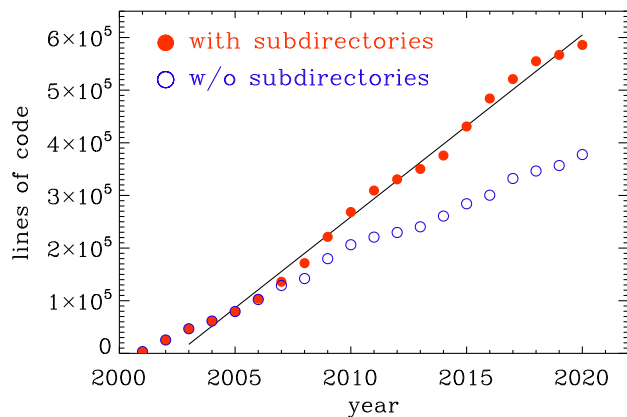


Figure 1: Lines of f90 code with and without subdirectories. The fit starts with 2003.

Next year, it will be 20 years of PENCIL CODE. Nothing special is planned yet, but the PENCIL CODE User Meeting is planned to take place again in Glasgow, where Simon is still holding some funds, which remained unused during this year's meeting. If you have ideas for recognizing the anniversary, please let us know.

3 Election of the PCSC

The PCSC started its work in 2016 and was now newly elected; see <https://www.nordita.org/~brandenb/pencil-code/PCSC/> for further information, including the minutes of past meetings.

Jennifer Schober (PENCIL CODE committer from the EPFL in Lausanne) acted as the impartial observer. As a novelty, we had to enter a runoff election as in the first round five candidates had the same number of votes. The runoff round then yielded a unambiguous result. Jennifer registered a voter turnout of 87% in the first round and of 65% in the runoff round. The result is

- Axel Brandenburg with 18 votes (= 90%),
- Nils Haugen with 17 votes (= 85%),
- Matthias Rheinhardt with 13 votes (= 65%),
- Piyali Chatterjee with 8 votes (= 53%), and
- Philippe Bourdin with 7 votes (= 47%).

We thank Jennifer for handling this task and the PCSC welcomes the two new members, Piyali and Philippe, to its circle. We recall that the Terms of Reference of

the PCSC are in: <https://www.nordita.org/~brandenb/pencil-code/PCSC/ToR/ToR.pdf>.

We also thank Wolfgang and Wlad for the work on the committee since 2016. Wolfgang is a core developer since the first hour back in April 2001, when he and Axel developed the first fragments of a code that, during the Helmholtz Summer School in Potsdam in August/September 2001, expanded into a public domain code with some rudimentary hydrodynamics at the time.

During its constitutive meeting on November, 27th, the committee appointed Nils Haugen (SINTEF, Trondheim, Norway) to be its chairperson. Congratulations, Nils, and good luck!

4 JOSS paper

The submission of a short publication for the Journal of Open Source Software (JOSS) <https://joss.theoj.org/about> is still pending; see the pre-review in <https://github.com/openjournals/joss-reviews/issues/2684> and also Figure 2 for the arXiv look. Some issues were already raised by one of the two referees, who has so far mainly been concerned with our web documentation and software installation. We have fixed those; see the end of the report <https://github.com/openjournals/joss-reviews/issues/2807> as it is being assembled.



The Pencil Code, a modular MPI code for partial differential equations and particles: multipurpose and multiuser-maintained

The Pencil Code Collaboration¹, Axel Brandenburg^{1,2,3}, Anders Johansen⁴, Philippe A. Bourdin⁵, Wolfgang Dobler⁶, Wladimir Lyra⁷, Matthias Rheinhardt⁸, Sven Bingert⁹, Nils Erland L. Haugen^{10,11}, Antony Mee¹², Frederick Gent^{8,13}, Natalia Babkovskaia¹⁴, Chao-Chin Yang¹⁵, Tobias Heinemann¹⁶, Boris Dintrans¹⁷, Dhruvadya Mitra¹, Simon Candelaresi¹⁸, Jörn Warnecke¹⁹, Petri J. Käpylä²⁰, Andreas Schreiber¹⁴, Piyali Chatterjee²¹, Maarit J. Käpylä^{8,19}, Xiang-Yu Li¹, Jonas Krüger^{10,11}, Jörgen R. Aarnes¹¹, Graeme R. Sarson¹³, Jeffrey S. Oishi²², Jennifer Schober²³, Raphaël Plason²⁴, Christer Sandin¹, Ewa Karchniwy^{11,25}, Luiz Felipe S. Rodrigues^{13,26}, Alexander Hubbard²⁷, Gustavo Guerrero²⁸, Andrew Snodin¹³, Illa R. Losada¹, Johannes Pekkilä⁸, and Chengeng Qian²⁹

Figure 2: Our JOSS paper on <https://arxiv.org/abs/2009.08231>.

If you have any comments or suggestions on this, let us know.

5 Code of Conduct

While working on the JOSS submission, we noticed that many community projects adopt the <https://www.contributor-covenant.org/> code of conduct. We do this now too; see https://github.com/pencil-code/pencil-code/blob/master/license/CODE_OF_CONDUCT.md.

6 Letters to the editor

Sahel Dey and Piyali Chatterjee from Indian Institute of Astrophysics in Bengaluru wrote to us announcing their first results on plasma jet (aka spicule) formation in a three dimensional radiative MHD simulation of the solar atmosphere driven by sub-surface convection, using the code. A snapshot from their impressive PENCIL CODE simulation is shown in Fig. 3.

Xiang-Yu Li of Pacific Northwest National Lab wrote us about HDF5 IO being superior to IO with PENCIL-specific binary files (scattered) for particle modules in two ways: First, it creates monolithic files, significantly reducing the number of files on a super-computer when one performs massive DNS. Second, it can be read by the intrinsic `h5py` module of Python directly. This applies of course also to non-particle runs.

Illa R. Losada (Canary Islands) advocated a list for python-related problems, such as where to import modules, a general coding style, documentation, or functionality checking. In response, Fred Gent proposed a Python PENCIL CODE meeting, which took place on November 23. The main topics were:

- Currently, the imports are placed inside the function, but this practice goes against the PEP-8 Python rules. During the meeting, some developers were in favor of the current rule, whereas others proposed to change to PEP-8. This issue remains open.
- Currently we cannot reload specific functions while using `ipython`. The problem seems to be that the file names are the same as the functions and classes. The solution can be either changing names, or finding other reload options.
- Illa suggested to use `epydoc` or `sphinx` to automatically generate documentation of the code, and will present the better option, also in terms of minimal code changes, in the next meeting.

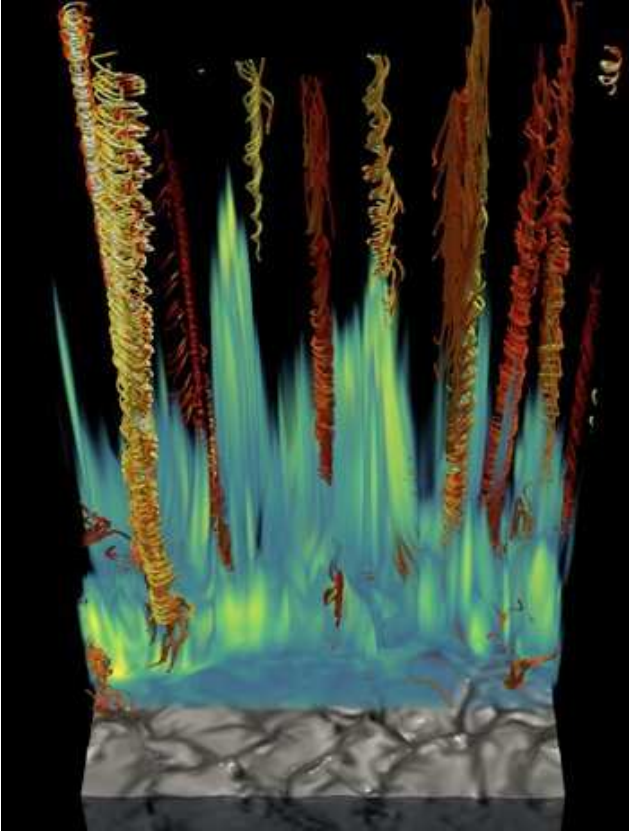


Figure 3: Volume rendering of synthetic plasma intensity (green-yellow) at a temperature of 10,000 K to show solar spicules. The helical streamlines correspond to highly dynamic coronal swirls (shaded yellow for anti-clockwise and red for clockwise) formed alongside the spicules. Courtesy of S. Dey and P. Chatterjee.

- We updated the PENCIL CODE Python wiki, changing out-of-date functions to the new python routines. Also, we planned on adding more tutorials there.
- We agreed on duplicating the current `jupyter` tutorials on the Python part to include Python scripts and `ipython` examples. Simon added the file `README_DEVELOPER` to the tutorials with this information and with the preferred tutorial format. Simon also changed the structure of the tutorials directory.
- Wolfgang suggested the use of `Black` for formatting and standardizing the code. He will give more information about this tool in the next meeting.
- For the PENCIL CODE postprocessing functions that are available exclusively in IDL, we discussed whether they are still in use, and if so, whether to

rewrite them in Python.

The meeting was really successful and gave us an idea of the current situation. We decided to have short periodic meetings to monitor and continue the development of the Python branch.

7 Training opportunities

The Numerical Algorithm Group (NAG), representing several HPC centers of excellence in Europe (<https://pop-coe.eu/partners>), is acting in the framework of the EU H2020 – Performance Optimization and Productivity (POP) project. They offer support on optimization of computational codes developed and used by researchers. The following kinds of support are dedicated for research groups developing their own codes, particularly those who perform parallel computations.

- participation of a dedicated POP person in code performance analysis,
- access to profiling tools (<https://pop-coe.eu/partners/tools>),
- organization of on-line training on selected profiling tools,
- access to learning material.

The team of Federico Panichi at NAG decided to share a free-of-charge offer for an on-line training course which might be shared with our community. POP offers on-line training at any time, it can be descriptive/virtual training. If you are interested in this on-line training, please email [<federico.panichi@nag.co.uk>](mailto:federico.panichi@nag.co.uk).

8 Employment opportunities

Wladimir Lyra [<wladimir.lyra@gmail.com>](mailto:wladimir.lyra@gmail.com) at New Mexico State University will be hiring a postdoc in computational planet formation; see <https://jobregister.aas.org/ad/0b720bba>. The deadline is Saturday, December 15.

9 Code developments

The code is changing as we speak; see <https://github.com/pencil-code/pencil-code/commits/master>. Keep an eye on this. Some changes can be important.

9.1 Infrastructure and postprocessing

GCC10 On Oct 8, Chao-Chin Yang noted problems with the GCC10 compiler due to overeager type checking. Chao-Chin implemented in the Makefile that whenever the gfortran compiler with version 10 or higher is detected, the compiler will avoid type checking on several source files; see commit d15dad8f <https://github.com/pencil-code/pencil-code/commit/d15dad8f987878795396fe61487ec7ea798d2e3f>. This workaround has worked for him.

cvsci_run This script was used to check in relevant minimal information into CVS, but it can now also be used just for producing a tarball. Check with Matthias if there are problems.

pc_varcontent.pro This routine used to trouble us. Matthias has now done some modifications.

9.2 Mail from the code

Given that code warnings are often overlooked by users, it seems desirable to send an e-mail in case of a severe warning, which is likely to require action. To enable this, one can provide one's address either by exporting the environment variable `PENCIL_USER_MAILADDR` properly or by setting `mailaddress` in `run.pars`. If the mail command can't be simply `mail`, one would also need to set `PENCIL_USER_MAILCMD` appropriately.

9.3 Automatic snapshot repair

Running huge setups on many CPUs with `IO = io_dist` is increasingly endangered due to job canceling by the queuing system or node failure, both potentially resulting in corrupted snapshots. As this until now fatally stopped the runs, connected with the necessity to return to a (hopefully not too distant in time) earlier snapshot and get back to the queue again, the code is now by default enabled to explore whether one of the up to 26 neighboring CPUs has written a healthy snapshot chunk. The first one found is used to overwrite the corrupted one, which allows the code to continue. Of course, the solution is incorrect in the respective subdomain, but the error can be expected to "heal out" while the integration is ongoing, if the subdomain volume is not too big. Given the delicate character of this operation, the user is notified by e-mail, if enabled.

9.4 Abandoning source linking

The source code subdirectory `src` now contains on all levels more than 600 files with a growing tendency. Although a concrete setup is normally only employing a subset of them, at least all files of the top level of `src` are linked to the `src` of the working directory. After compilation this would easily contain several 100 additional `*.o` and `*.mod` files. On the other hand, some clusters set quotas for the number of files, and when considering that the number of run directories needed for a single paper may well reach a hundred, such quotas become relevant. A way out could be to avoid the linking completely, also making the `pc_setupsrc` step dispensable. The only disadvantage would be to lose a convenient way to access the source files from the working directory. Convenience can be restored by a "smart" editor releasing one from the duty always to type `$PENCIL_HOME/src/...` when accessing a source file. A starting point for such a device is `pc_edit`. It takes as its first argument the name of an editor and as the second the name of the file. As it searches all subdirectories of `$PENCIL_HOME` one can even use only the file name (if unique). To adapt to personal preferences, one would define an alias, e.g., `alias vip='pc_edit vim'` if your favored editor is `vi`. File name completion is partly implemented. To activate it, one has to source the file `bin/pc_complete` in `.bashrc` and after that to add `complete -F _pc_complete <user-defined alias>`. For some other functionality of file handling like, e.g., `ls`, analogous tools are needed. Let us know your opinions.

10 Papers since September

Since the last newsletter of September 10, some new papers have appeared on the arXiv, and others that were just preprints, have now been published. We list both, but not intermediate updates. We also list two Zenodo references that are now also on ADS.

References

Brandenburg, A., Johansen, A., Bourdin, P.A., Dobler, W., Lyra, W., Rheinhardt, M., Bingert, S., Haugen, N.E.L., Mee, A., Gent, F., Babkovskaia, N., Yang, C.C., Heinemann, T., Dintrans, B., Mitra, D., Candelaresi, S., Warnecke, J., Käpylä, P.J., Schreiber, A., Chatterjee, P., Käpylä, M.J., Li, X.Y., Krüger, J., Aarnes, J.R., Sarson, G.R., Oishi, J.S., Schober, J., Plasson, R., Sandin, C., Karchniwy, E., Ro-

- drigues, L.F.S., Hubbard, A., Guerrero, G., Snodin, A., Losada, I.R., Pekkilä, J. and Qian, C., The Pencil Code, a modular MPI code for partial differential equations and particles: multipurpose and multiuser-maintained, 2020, arXiv:2009.08231.
- Brandenburg, A., Hall cascade with fractional magnetic helicity in neutron star crusts. *Astrophys. J.*, 2020a, **901**, 18.
- Brandenburg, A., Piecewise quadratic growth during the 2019 novel coronavirus epidemic. *Infectious Disease Modelling*, 2020b, **5**, 681–690.
- Brandenburg, A., Scientific usage of the Pencil Code., 2020c, DOI:10.5281/zenodo.3466444.
- Brandenburg, A. and Das, U., Turbulent radiative diffusion and turbulent Newtonian cooling., 2020, arXiv:2010.07046.
- Gent, F.A., Mac Low, M.M., Kopyla, M.J. and Singh, N.K., Small-Scale Dynamo in Supernova-Driven Interstellar Turbulence, 2020, arXiv:2010.01833.
- Hyder, A., Lyra, W., Chanover, N., Morales-Juberias, R. and Jackiewicz, J., Vortex Dynamics in the Polar Atmosphere of Jupiter; in *AAS/Division for Planetary Sciences Meeting Abstracts*, Vol. 52 of *AAS/Division for Planetary Sciences Meeting Abstracts*, Oct., 2020, p. 103.01.
- Kahniashvili, T., Brandenburg, A., Gogoberidze, G., Mandal, S. and Roper Pol, A., Circular polarization of gravitational waves from early-universe helical turbulence, 2020, arXiv:2011.05556.
- Klahr, H. and Schreiber, A., Testing the Jeans, Toomre and Bonnor-Ebert concepts for planetesimal formation: 3D streaming instability simulations of diffusion regulated formation of planetesimals., 2020a, arXiv:2011.07849.
- Klahr, H. and Schreiber, A., Turbulence Sets the Length Scale for Planetesimal Formation: Local 2D Simulations of Streaming Instability and Planetesimal Formation. *Astrophys. J.*, 2020b, **901**, 54.
- Li, X.Y. and Mattsson, L., Dust Growth by Accretion of Molecules in Supersonic Interstellar Turbulence. *Astrophys. J.*, 2020, **903**, 148.
- Pencil Code Collaboration, The Pencil Code., 2020, DOI:10.5281/zenodo.3961647.
- Roper Pol, A., Mandal, S., Brandenburg, A., Kahniashvili, T. and Kosowsky, A., Numerical simulations of gravitational waves from early-universe turbulence. *Phys. Rev. D*, 2020, **102**, 083512.

This PENCIL CODE Newsletter was edited by Axel Brandenburg <brandenb@nordita.org>, Nordita, KTH Royal Institute of Technology and Stockholm University, Roslagstullsbacken 23, SE-10691 Stockholm, Sweden; and Matthias Rheinhardt <matthias.rheinhardt@aalto.fi>, Department of Computer Science, Aalto University, PO Box 15400, FI-00076 Aalto, Finland. See <http://www.nordita.org/~brandenb/pencil-code/newsletter> for the online version. **Back issues of the newsletter can also be found here.**