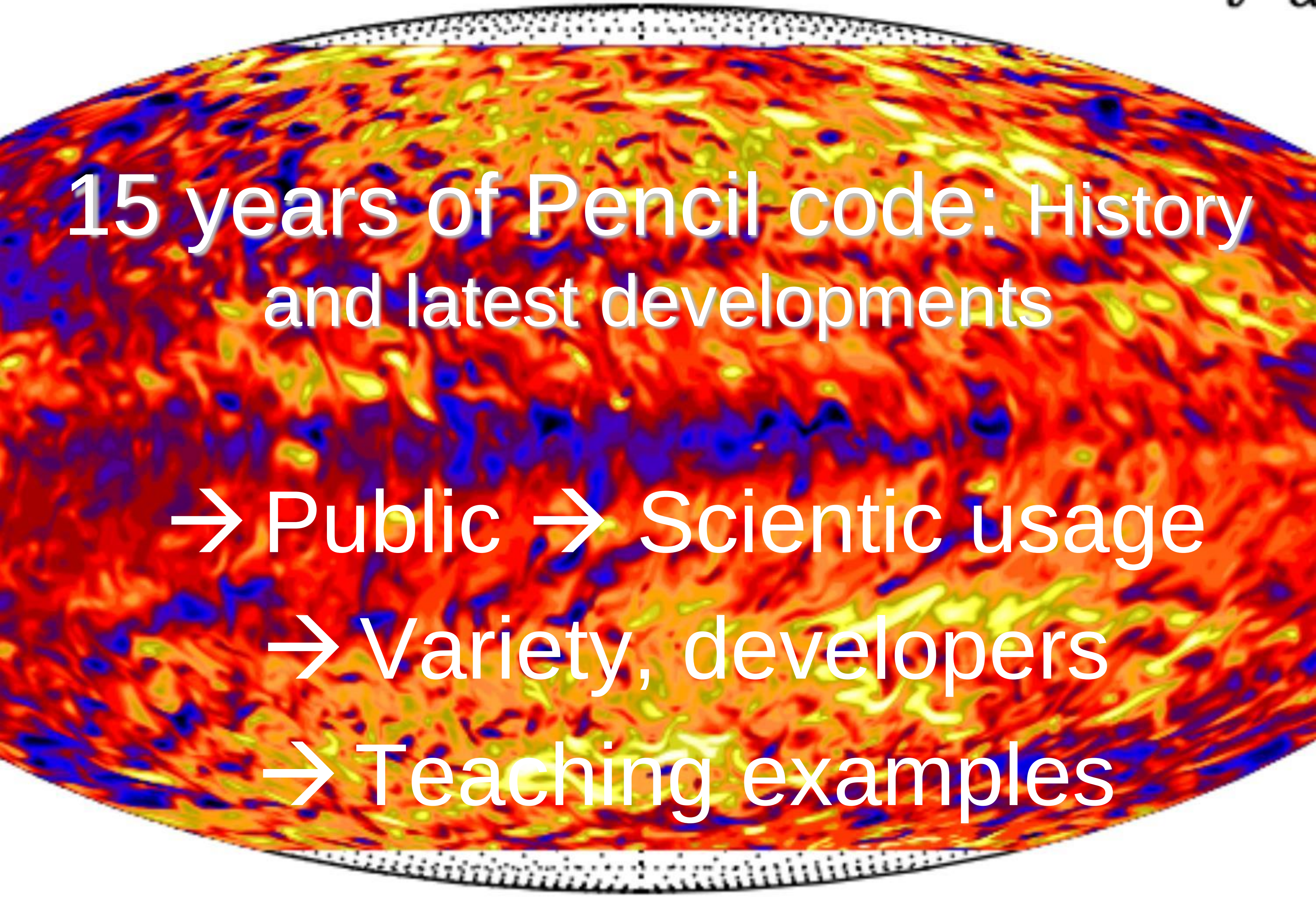
15 years of Pencil code: History and latest developments

→ Public → Scientic usage

→ Variety, developers

→ Teaching examples



Pencil Code Philosophy

- Maximum freedom
  - Alternatives possible
- Research-driven
  - Can’t expect service
- Minimum ties
  - Acknowledge development work of others
- Only one version
  - Minimal duplication

License agreement and giving credit

We, the PENCIL CODE community, ask that in publications and presentations the use of the code (or parts of it) be acknowledged with reference to the web site `http://www.nordita.org/software/pencil-code/`. As a courtesy to the people involved in developing particularly important parts of the program (use `svn annotate src/*.f90` to find out who did what!) we suggest to give appropriate reference to one or several of the following papers (listed here in temporal order):

Dobler, W., Haugen, N. E. L., Yousef, T. A., & Brandenburg, A.: 2003, "Bottleneck effect in three-dimensional turbulence simulations," *Phys. Rev.* **E 68**, 026304, 1-8 (astro-ph/0303324)

Haugen, N. E. L., Brandenburg, A., & Dobler, W.: 2003, "Is nonhelical hydromagnetic turbulence peaked at small scales?" *Astrophys. J. Lett.* **597**, L141-L144 (astro-ph/0303372)

Brandenburg, A., Käpylä, P., & Mohammed, A.: 2004, "Non-Fickian diffusion and tau-approximation from numerical turbulence," *Phys. Fluids* **16**, 1020-1027 (astro-ph/0306521)

Johansen, A., Andersen, A. C., & Brandenburg, A.: 2004, "Simulations of dust-trapping vortices in protoplanetary discs," *Astron. Astrophys.* **417**, 361-371 (astro-ph/0310059)

Haugen, N. E. L., Brandenburg, A., & Mee, A. J.: 2004, "Mach number dependence of the onset of dynamo action," *Monthly Notices Roy. Astron. Soc.* **353**, 947-952 (astro-ph/0405453)

Free licence, but giving credit to research

Brandenburg, A., Rädler, K.-H., Rheinhardt, M., & Käpylä, P. J.: 2008, "Magnetic diffusivity tensor and dynamo effects in rotating and shearing turbulence," *Astrophys. J.* **676**, 740-751 (arXiv/0710.4059)

Lyra, W., Johansen, A., Klahr, H., & Piskunov, N.: 2008, "Embryos grown in the dead zone. Assembling the first protoplanetary cores in low-mass selfgravitating circumstellar disks of gas and solids," *Astron. Astrophys.* **491**, L41-L44

Lyra, W., Johansen, A., Klahr, H., & Piskunov, N.: 2009, "Standing on the shoulders of giants. Trojan Earths and vortex trapping in low-mass selfgravitating protoplanetary disks of gas and solids," *Astron. Astrophys.* **493**, 1125-1139

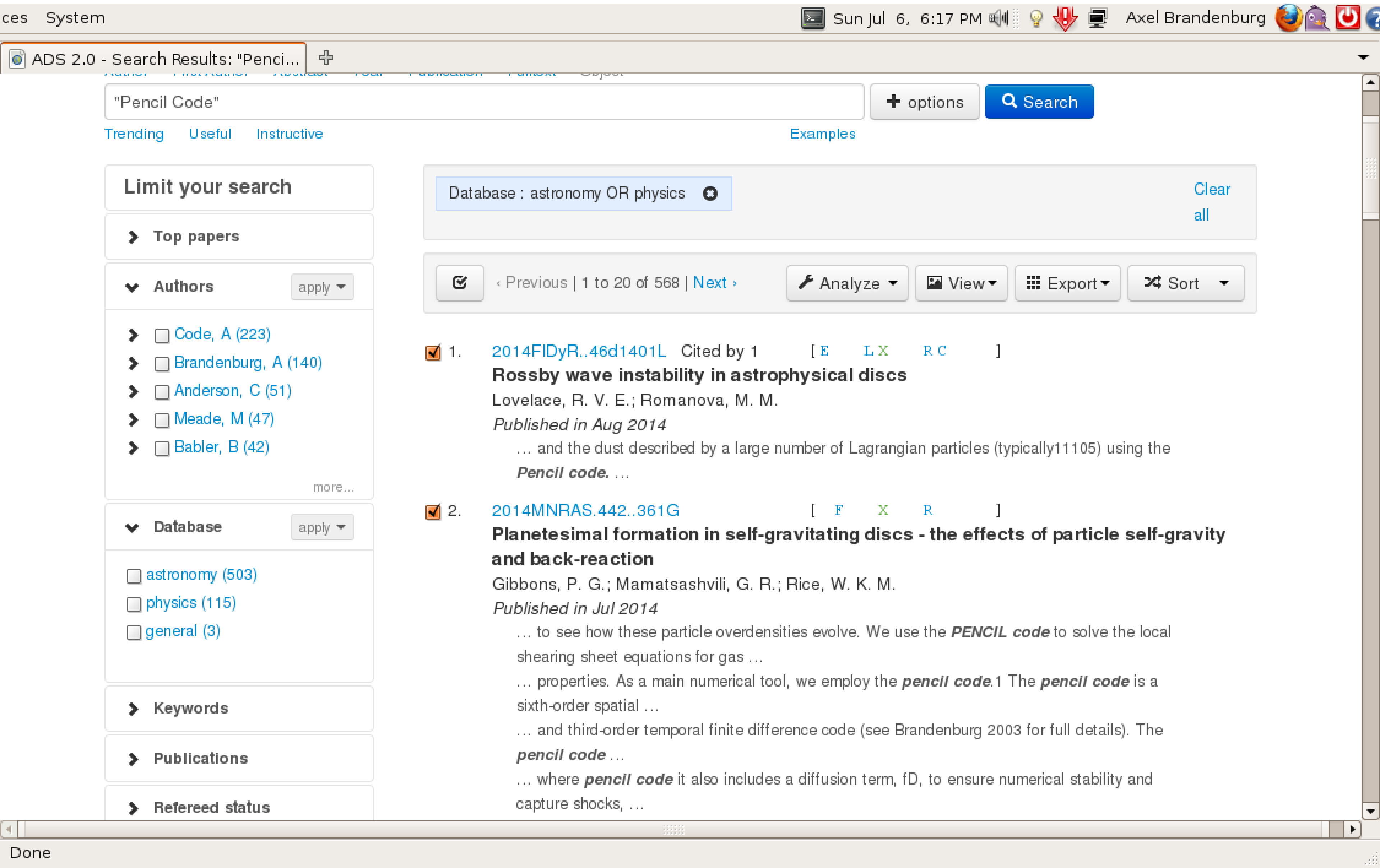
Lyra, W., Johansen, A., Zsom, A., Klahr, H., & Piskunov, N.: 2009, "Planet formation bursts at the borders of the dead zone in 2D numerical simulations of circumstellar disks," *Astron. Astrophys.* **497**, 869-888 (arXiv/0901.1638)

Mitra, D., Tavakol, R., Brandenburg, A., & Moss, D.: 2009, "Turbulent dynamos in spherical shell segments of varying geometrical extent," *Astrophys. J.* **697**, 923-933 (arXiv/0812.3106)

Haugen, N. E. L., & Kragset, S.: 2010, "Particle impaction on a cylinder in a crossflow as function of Stokes and Reynolds numbers," *J. Fluid Mech.* **661**, 239-261

Rheinhardt, M., & Brandenburg, A.: 2010, "Test-field method for mean-field coefficients with MHD background," *Astron. Astrophys.* **520**, A28 (arXiv/1004.0689)

Using ADSlab; search for "Pencil Code"



Pair dispersion of turbulent premixed flame elements

Swetaprovo Chaudhuri*

Department of Aerospace Engineering, National Center for Combustion Research and Development, Indian Institute of Science, Bangalore 560012, India

(Received 20 November 2014; published 26 February 2015)

Flame particles are mathematical points comoving with a reacting isoscalar surface in a premixed flame. In this Rapid Communication, we investigate mean square pair separation of flame particles as a function of time from their positions tracked in two sets of direct numerical simulation solutions of H_2-air turbulent premixed flames with detailed chemistry. We find that, despite flame particles and fluid particles being very different concepts, a modified Batchelor's scaling of the form $\langle|\mathbf{\Delta}^F(t) - \mathbf{\Delta}^F(0)|^2\rangle = C_F(\langle\varepsilon\rangle_0^F \Delta_0^F)^{2/3} t^2$ holds for flame particle pair dispersion. The proportionality constant, however, is not universal and depends on the isosurface temperature value on which the flame particles reside. Following this, we attempt to analytically investigate the rationale behind such an observation.

II. COMPUTATIONS

Two DNS cases (cases A and B) were performed with the PENCIL code for lean H_2-air premixed flames with detailed chemical reaction mechanism of [17]. PENCIL code is an open source code designed for compressible turbulent flows using sixth order finite difference and third order Runge-Kutta schemes for spatial and temporal discretization, respectively. Combustion chemistry was implemented in [18]. The well-

[17] J. Li, Z. W. Zhao, A. Kazakov, and F. L. Dryer, Int. J. Chem. Kinet. **36**, 566 (2004).

[18] N. Babkovskaia, N. E. L. Haugen, and A. Brandenburg, J. Comput. Phys. **230**, 1 (2011).

Assemble bibtex file

Scientific usage of the PENCIL CODE

Search results using http://adslabs.org and Bumblebee https://ui.adsabs.harvard.edu/

http://pencil-code.nordita.org/highlights/

A search using http://adslabs.org or Bumblebee https://ui.adsabs.harvard.edu/ lists the papers in which the PENCIL CODE is being quoted. In the following we present the papers that are making use of the code either for their own scientific work of those authors, or for code comparison purposes. We include conference proceedings, which make up 15–20% of all papers. We classify the references by year and by topic, although the topics are often overlapping. The primary application of the PENCIL CODE lies in astrophysics, in which case we classify mostly by the field of research.

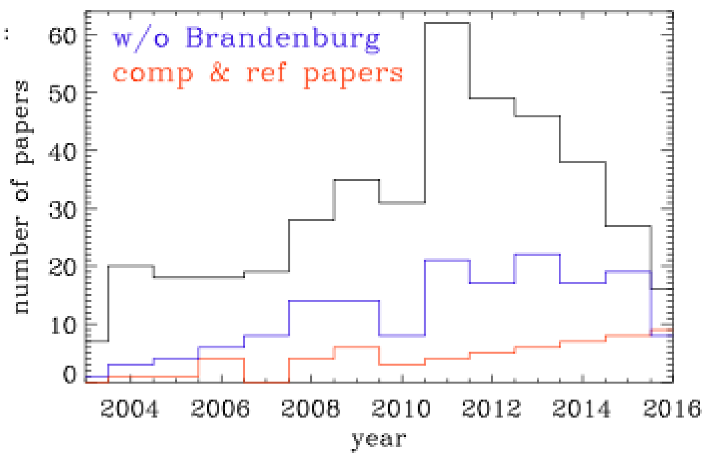


Figure 1: Number of papers since 2003 that make use of the PENCIL CODE. In red is shown the number of papers that reference it for code comparison or other purposes and in blue the papers that are not co-authored by Brandenburg. The enhanced number of papers during 2011–2013 results from publications related to his ERC Advanced Grant.

1 Papers by year

As of May 2016, the PENCIL CODE has been used for a total of 416 research papers; see Figure 1. In addition, 58 papers reference it for code comparison or other purposes (see the red line).

17 times in 2016 (Bhat et al., 2016; Osano and Adams, 2016; Yang and Johansen, 2016; Cole et al., 2016; Warnecke et al., 2016a,b; Li et al., 2016; Mitra et al., 2016; Lambrechts et al.,

and 7 times in 2003 (Yousef et al., 2003; Yousef and Brandenburg, 2003; McMillan and Sarson, 2003; Haugen et al., 2003; Brandenburg, 2003; Brandenburg et al., 2003; Dobler et al., 2003).

2 Papers by topic

The PENCIL CODE has been used for the following research topics

1. **Interstellar and intercluster medium as well as early Universe**

   (a) *Interstellar and intercluster medium* (Rodrigues et al., 2016; Gent, 2014; Chamandy et al., 2013; Gent et al., 2013a,b; Bykov et al., 2013; Yang and Krumholz, 2012; Mantere and Cole, 2012; Rogachevskii et al., 2012; Ruoskanen et al., 2011; Ruszkowski et al., 2007, 2008; Brandenburg et al., 2007b; Gustafsson et al., 2006, 2007; Brandenburg et al., 2005a; Haugen et al., 2004b; Brandenburg et al., 2003).

   (b) *Small-scale dynamos and reconnection* (Bhat and Subramanian, 2013; Brandenburg, 2011c; Baggaley et al., 2009, 2010; Schekochihin et al., 2005, 2007; Haugen and Brandenburg, 2004b; Haugen et al., 2003, 2004a,c; Dobler et al., 2003).

   (c) *Primordial magnetic fields and decaying turbulence* (Osano and Adams, 2016; Kahniashvili et al., 2015; Brandenburg et al., 2015; Adams and Osano, 2014; Kahniashvili et al., 2012, 2013; Tevzadze et al., 2012; Candelaresi and Brandenburg, 2011a; Kahniashvili et al., 2010; Del Sordo et al., 2010; Christensson et al., 2005; Yousef et al., 2004).

2. **Planet formation and inertial particles**

   (a) *Planet formation* (Lyra et al., 2016; Yang and Johansen, 2016; Lambrechts et al., 2016; Johansen et al., 2015; Richert et al., 2015a; Gibbons et al., 2015; Baehr and Klahr, 2015; Carrera et al., 2015, 2014; Yang and Johansen, 2014a,b; McNally et al., 2014; Turner et al., 2014; Gibbons et al., 2014; Dittrich et al., 2014, 2013; Hubbard, 2013; Lyra and Kuchner, 2013; Gibbons et al., 2012; Hubbard, 2012; Horn et al., 2012; Lyra and Kuchner, 2012; Yang et al., 2012; Lambrechts and Johansen, 2012; Johansen et al., 2012; Fromang et al., 2011; Johansen et al., 2011c; Lambrechts, 2011; Johansen et al.,

   2008; Oishi et al., 2007; Johansen et al., 2007a,b; Johansen and Youdin, 2007; Youdin and Johansen, 2007; Johansen et al., 2006a,b,c; Johansen and Klahr, 2005; Johansen et al., 2004, 2005).

   (b) *Inertial particles* (Li et al., 2016; Mitra et al., 2016; Raettig et al., 2015; Pan and Padoan, 2014, 2013; Pan et al., 2014b,a; Mitra et al., 2013; Haugen et al., 2012; Hydle Rivedal et al., 2011; Haugen et al., 2010).

3. **Accretion discs and shear flows**

   (a) *Accretion discs and shear flows* (Tian and Chen, 2016; Richert et al., 2015b; Lyra, 2014; Lyra et al., 2015; Väisälä et al., 2014; Lyra, 2013; Raettig et al., 2013; Di Bernardo and Torkelsson, 2013; Latter and Papaloizou, 2012; Gaburov et al., 2012; Lyra and Mac Low, 2012; Rice et al., 2011, 2012; Oishi and Mac Low, 2011; Flock et al., 2011; Käpylä et al., 2010a; Käpylä and Korpi, 2011; Fromang et al., 2010; Korpi et al., 2010; Johansen et al., 2009a; Heinemann and Papaloizou, 2009; Fromang et al., 2009; Johansen and Levin, 2008; Workman and Armitage, 2008; Fromang et al., 2007; Fromang and Papaloizou, 2007; Ouyed et al., 2006; Brandenburg, 2005d).

   (b) *Shear flows* (Singh and Jingade, 2015; Modestov et al., 2014; Vermersch and Brandenburg, 2009; Käpylä et al., 2009c; Green et al., 2008; Yousef et al., 2008; Babkovskaia et al., 2008; Brandenburg et al., 2004a).

4. **Solar physics**

   (a) *Coronal heating and coronal mass ejections* (Bourdin et al., 2016; Threlfall et al., 2016; Chen et al., 2015; Smiet et al., 2015; Warnecke and Brandenburg, 2014; Bourdin, 2014; Bourdin et al., 2013a,b,c; van Wettum et al., 2013; Bingert and Peter, 2013; Peter and Bingert, 2012; Peter et al., 2012; Warnecke et al., 2012a,b; Warnecke and Brandenburg, 2011a; Zacharias et al., 2011a,b; Warnecke et al., 2011b; Bingert and Peter, 2011; Warnecke and Brandenburg, 2011b; Warnecke et al., 2011a; Warnecke and Brandenburg, 2010; Bingert et al., 2010; Zacharias et al., 2009b,a).

   (b) *Helical dynamos, helical turbulence, and catastrophic quenching* (Cole et al., 2016; Karak and Brandenburg, 2016; Karak et al., 2015b; Subramanian and Brandenburg, 2014; Brandenburg and Stepanov, 2014; Brandenburg, 2014; Brandenburg and Hubbard, 2015; Chian et al., 2014; Park, 2014b; Park et al., 2013; Brandenburg and Lazarian, 2013; Park, 2013b,a, 2014a; Candelaresi and Brandenburg, 2013a; Park, 2013a; Del Sordo et al., 2013; Brandenburg, 2013; Rempel et al., 2013; Candelaresi and Bran-

Variety of usage

- Interstellar medium
  - Galaxy clusters, Early Universe
- Planet formation
  - Inertial particles, raindrop formation
- Accretion disks
  - Shear flows
- Solar Physics
  - Coronal heating, dynamos, spot formation
  - Convection, global convective dynamos
- Miscellanea
  - Test-field method, Hydro turb, turb combustion

interactions, galactic dynamics, star formation and planet formation and given the implemented physics, other applications are possible as well.

[ascl:1010.060] Pencil: Finite-difference Code for Compressible Hydrodynamic Flows

Brandenburg, Axel; Dobler, Wolfgang

The Pencil code is a high-order finite-difference code for compressible hydrodynamic flows with magnetic fields. It is highly modular and can easily be adapted to different types of problems. The code runs efficiently under MPI on massively parallel shared- or distributed-memory computers, like e.g. large Beowulf clusters. The Pencil code is primarily designed to deal with weakly compressible turbulent flows. To achieve good parallelization, explicit (as opposed to compact) finite differences are used. Typical scientific targets include driven MHD turbulence in a periodic box, convection in a slab with non-periodic upper and lower boundaries, a convective star embedded in a fully nonperiodic box, accretion disc turbulence in the shearing sheet approximation, self-gravity, non-local radiation transfer, dust particle evolution with feedback on the gas, etc. A range of artificial viscosity and diffusion schemes can be invoked to deal with supersonic flows. For direct simulations regular viscosity and diffusion is being used. The code is written in well-commented Fortran90.

[ascl:1010.074] StarCrash: 3-d Evolution of Self-gravitating Fluid Systems

Faber, Joshua; Lombardi, Jamie; Rasio, Fred

StarCrash is a parallel fortran code based on Smoothed Particle Hydrodynamics (SPH) techniques to calculate the 3-d evolution of self-gravitating fluid systems. The code in particularly suited to the study of stellar interactions, such as mergers of binary star systems and stellar collisions. The StarCrash code comes with several important features, including:

- Several routines which construct the initial conditions appropriate to a wide variety of physical systems
- An efficient parallel neighbor-finding algorithm for calculating hydrodynamic quantities
- A parallel gravitational field solver based on FFT convolution techniques, which uses the FFTW software libraries
- Relaxation Techniques for single stars and synchronized binaries
- Three different artificial viscosity treatments to calculate the thermodynamic evolution of the matter
- An optional gravitational radiation back-reaction treatment, which calculates the damping force from gravity wave losses to lowest relativistic order in a spatially accurate way

[ascl:1010.082] FLASH: Adaptive Mesh Hydrodynamics Code for Modeling Astrophysical Thermonuclear Flashes

Fryxell, B.; Olson, K.; Ricker, P.; Timmes, F. X.; Zingale, M.; Lamb, D. Q.; MacNeice, P.; Rosner, R.; Truran, J. W.; Tufo, H.

The FLASH code, currently in its 4th version, is a publicly available high performance application code which has evolved into a modular, extensible software system from a collection of unconnected legacy codes. FLASH consists of inter-operable modules that can be combined to generate

Searching for 'pencil'

[ascl:1010.060] Pencil: Finite-difference Code for Compressible Hydrodynamic Flows

Brandenburg, Axel; Dobler, Wolfgang

The Pencil code is a high-order finite-difference code for compressible hydrodynamic flows with magnetic fields. It is highly modular and can easily be adapted to different types of problems. The code runs efficiently under MPI on massively parallel shared- or distributed-memory computers, like e.g. large Beowulf clusters. The Pencil code is primarily designed to deal with weakly compressible turbulent flows. To achieve good parallelization, explicit (as opposed to compact) finite differences are used. Typical scientific targets include driven MHD turbulence in a periodic box, convection in a slab with non-periodic upper and lower boundaries, a convective star embedded in a fully nonperiodic box, accretion disc turbulence in the shearing sheet approximation, self-gravity, non-local radiation transfer, dust particle evolution with feedback on the gas, etc. A range of artificial viscosity and diffusion schemes can be invoked to deal with supersonic flows. For direct simulations regular viscosity and diffusion is being used. The code is written in well-commented Fortran90.

SAO/NASA ADS Astronomy Abstract Service

- Find Similar Abstracts (with default settings below)
- Electronic On-line Article (HTML)
- Citations to the Article (1) (Citation History)
- Refereed Citations to the Article
- Associated Articles
- Reads History
-
- Translate This Page

Title:	Pencil: Finite-difference Code for Compressible Hydrodynamic Flows
Authors:	Brandenburg, Axel; Dobler, Wolfgang
Publication:	Astrophysics Source Code Library, record ascl:1010.060
Publication Date:	10/2010
Origin:	ASCL; ATEL
Bibliographic Code:	2010ascl.soft10060B

Usage of “Pencil” rather common…

A pencil distributed finite difference code for strongly turbulent wall-bounded flows

Erwin P. van der Poel[1,a], Rodolfo Ostilla-Mónico[1], John Donners[2] and Roberto Verzicco[3,1]

[1] *Department of Physics, Mesa+ Institute, and J. M. Burgers Centre for Fluid Dynamics, University of Twente, 7500 AE Enschede, The Netherlands*

[2] *SURFsara, Science Park, 1098 XG Amsterdam, The Netherlands*

[3] *Dipartimento di Ingegneria Industriale, University of Rome “Tor Vergata”, Via del Politecnico 1, Roma 00133, Italy*

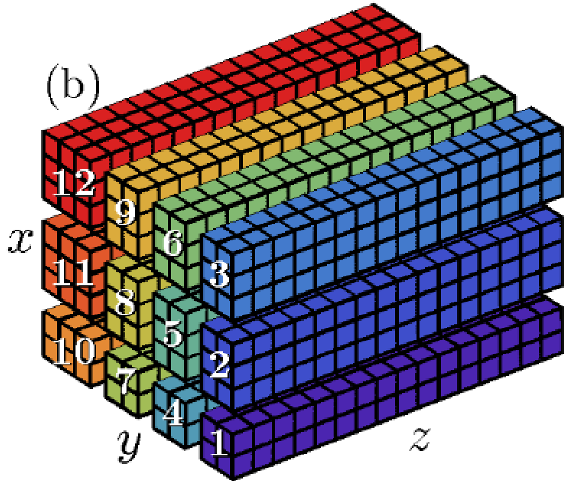
(Dated: January 7, 2015)

We present a numerical scheme geared for high performance computation of wall-bounded turbulent flows. The number of all-to-all communications is decreased to only six instances by using a two-dimensional (pencil) domain decomposition and utilizing the favourable scaling of the CFL time-step constraint as compared to the diffusive time-step constraint. As the CFL condition is more restrictive at high driving, implicit time integration of the viscous terms in the wall-parallel directions is no longer required. This avoids the communication of non-local information to a process for the computation of implicit derivatives in these directions. We explain in detail the numerical scheme used for the integration of the equations, and the underlying parallelization. The code is shown to have very good strong and weak scaling to at least 64K cores.

Usage of "Pencil" rather common…

the current density and particle density profiles. In Ref. 4, ITER simulations were carried out using the mixed Bohm/gyro-Bohm transport model in the JETTO code with heating profiles computed with the PENCIL code.[17] Furthermore, as a

[17] C. D. Challis, J. G. Cordey, H. Hamnen, P. M. Stubberfield, J. P. Christiansen, E. Lazzaro, D. G. Muir, D. Stork, and E. Thompson, Nucl. Fusion **29**, 563 (1989).



Pencil formulation

- In CRAY days: worked with full chunks *f(nx,ny,nz,nvar)*
  - Now, on SGI, nearly 100% cache misses
- Instead work with *f(nx,nvar)*, i.e. one *nx*-pencil
- No cache misses, negligible work space, just *2N*
  - Can keep all components of derivative tensors
- Communication before sub-timestep
- Then evaluate all derivatives, e.g. *call curl(f,iA,B)*
  - Vector potential *A=f(:,:,:,iAx:iAz), B=B(nx,3)*

Switch modules

- magnetic or nomagnetic (e.g. just hydro)
- hydro or nohydro (e.g. kinematic dynamo)
- density or nodensity (burgulence)
- entropy or noentropy (e.g. isothermal)
- radiation or noradiation (solar convection, discs)
- dustvelocity or nodustvelocity (planetesimals)
- Coagulation, reaction equations
- Chemistry (reaction-diffusion-advection equations)

Other physics modules: MHD, radiation, partial ionization, chemical reactions, selfgravity

High-order schemes

- Alternative to spectral or compact schemes
  - Efficiently parallelized, no transpose necessary
  - No restriction on boundary conditions
  - Curvilinear coordinates possible (except for singularities)
- 6th (or other) order central differences in space
- Non-conservative scheme
  - Allows use of logarithmic density and entropy
  - Copes well with strong stratification and temperature contrasts

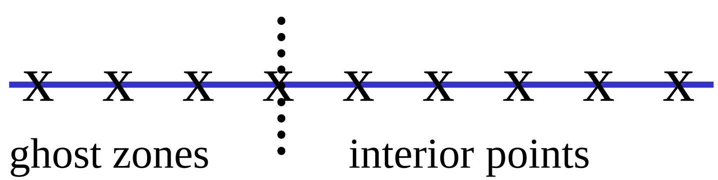
(i) High-order spatial schemes

Main advantage: low *phase* errors

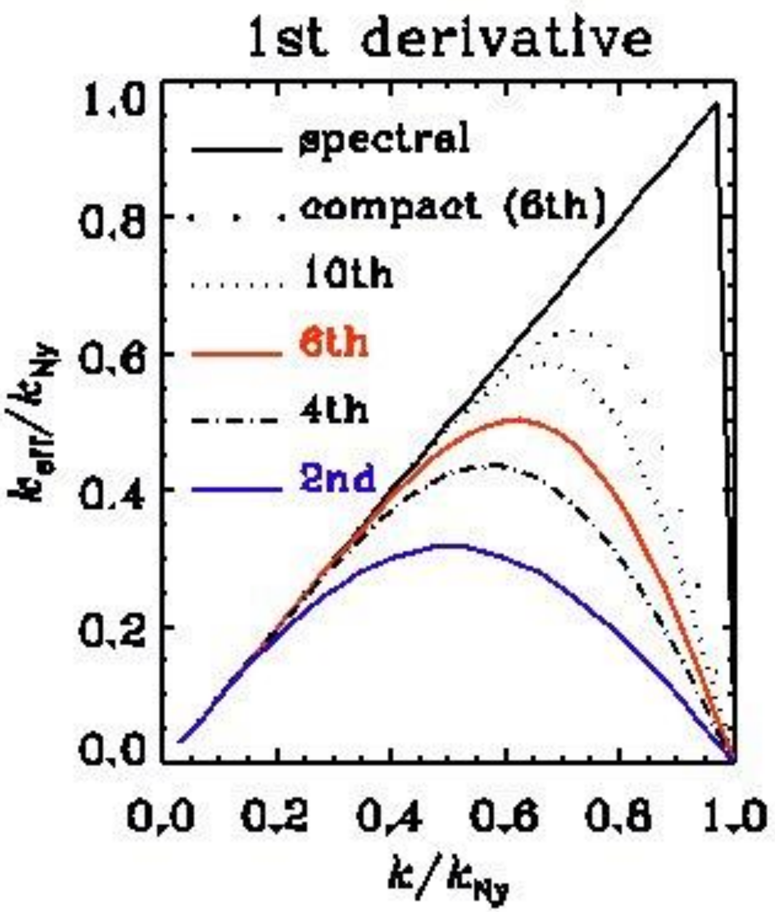
$$f_i' = \frac{-f_{i-3} + 9f_{i-2} - 45f_{i-1} + 45f_{i+1} - 9f_{i+2} + f_{i+3}}{60\delta x}$$

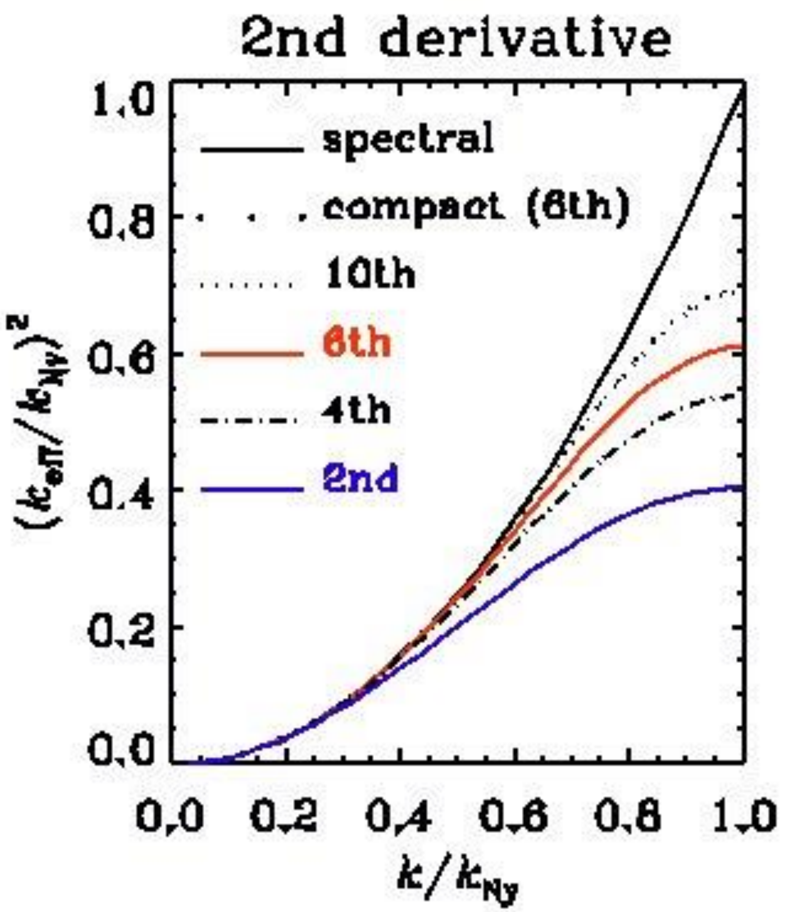
$$f_i'' = \frac{2f_{i-3} - 27f_{i-2} + 270f_{i-1} - 490f_i + 270f_{i+1} - 27f_{i+2} + 2f_{i+3}}{180\delta x^2}$$

Near boundaries:



Wavenumber characteristics

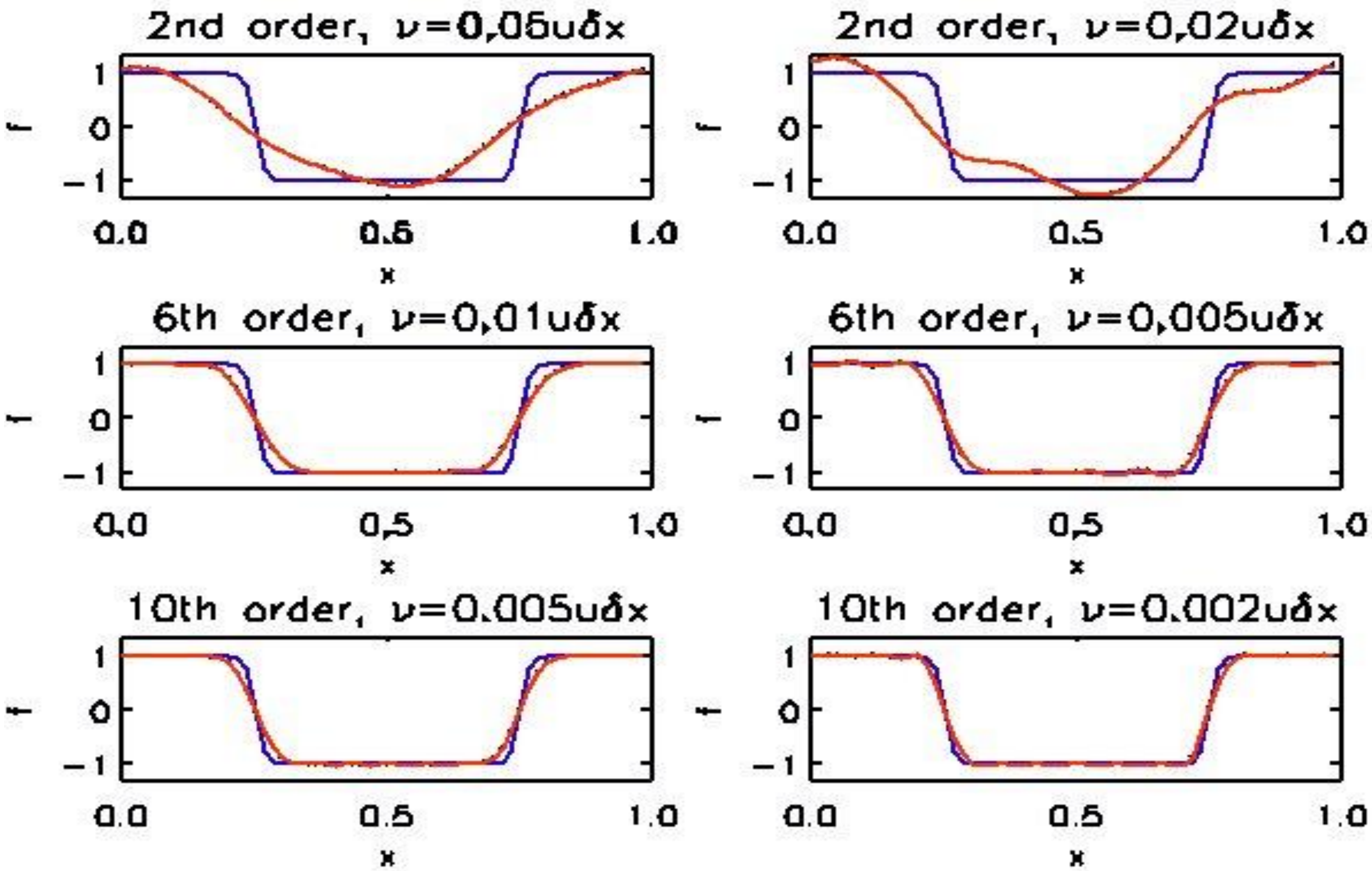




$$k_{eff} = \frac{d(\cos kx)/dx}{-\sin kx}$$

$$k_{eff}^2 = \frac{d^2(\cos kx)/dx^2}{-\cos kx}, \qquad k_{Ny} = \pi/\delta x$$

Higher order – less viscosity



(ii) High-order temporal schemes

Main advantage: low *amplitude* errors

2*N*-RK3 scheme (Williamson 1980)

$$w_i = \alpha_i w_{i-1} + \delta t\, F(t_{i-1}, u_{i-1})$$

$$u_i = u_{i-1} + \beta_i w_i$$

$$u_0 = u^{(n)}, \quad u^{(n+1)} = u_3$$

2nd order

$$\alpha_1 = 0, \quad \alpha_2 = -1/2$$

$$\beta_1 = 1/2, \quad \beta_2 = 1$$

3rd order

$$\alpha_1 = 0, \quad \alpha_2 = -2/3, \quad \alpha_3 = -1$$

$$\beta_1 = 1/3, \quad \beta_2 = 1, \quad \beta_3 = 1/2$$

1st order

$$\alpha_1 = 0$$

$$\beta_1 = 1$$

Transfer equation & parallelization

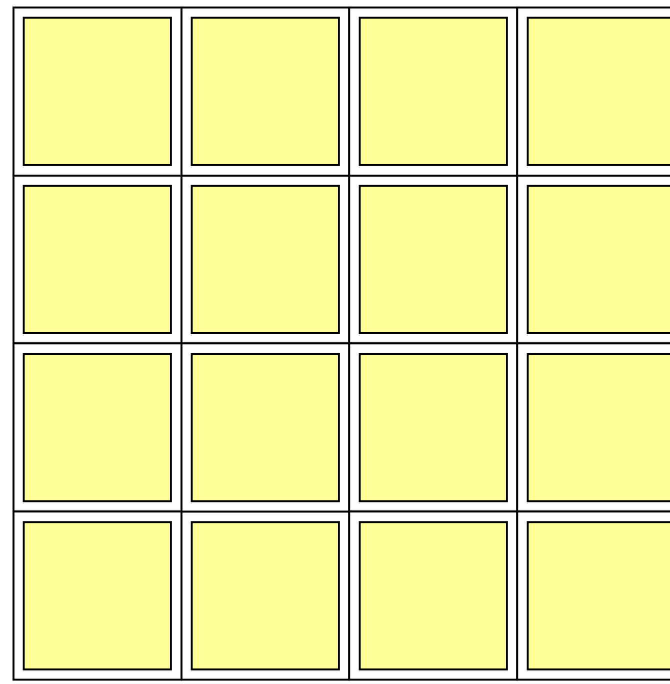
$$\frac{\mathrm{d}I}{\mathrm{d}\tau} = I - S$$

Analytic Solution:

$$I(\tau) = I_0 e^{\tau_0 - \tau} + \int_{\tau_0}^{\tau} e^{\tau' - \tau} S(\tau') d\tau'$$

Intrinsic Calculation

Processors



Ray direction ↗

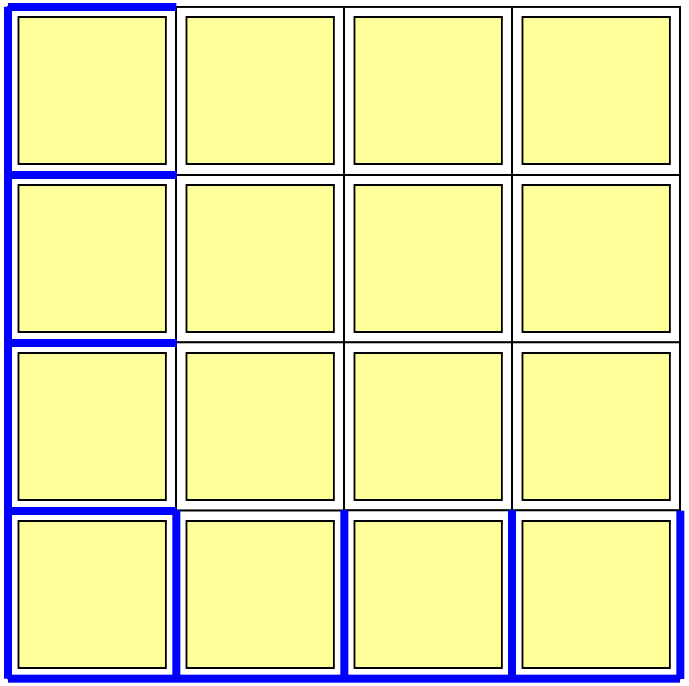
The Transfer Equation & Parallelization

Analytic Solution:

$$I(\tau) = I_0 e^{\tau_0 - \tau} + \int_{\tau_0}^{\tau} e^{\tau' - \tau} S(\tau') d\tau'$$

Communication

Processors



Ray direction

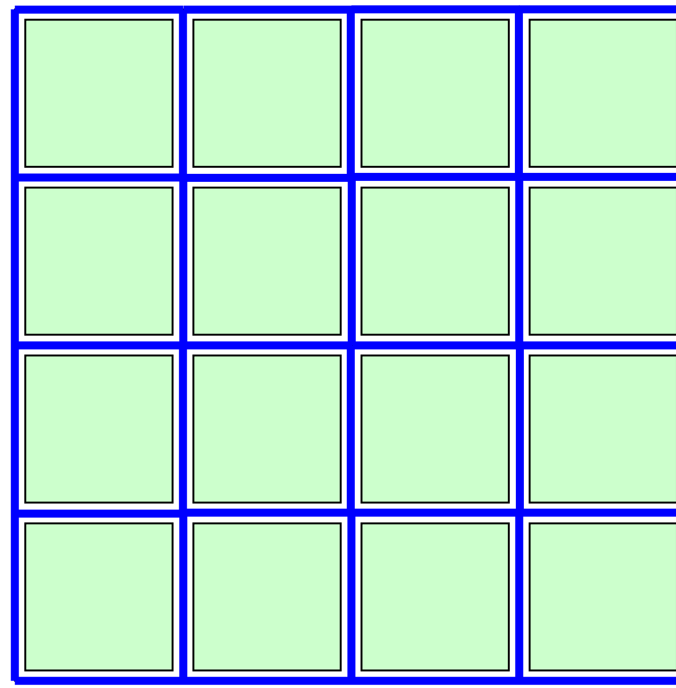
The Transfer Equation & Parallelization

Analytic Solution:

$$I(\tau) = \boxed{I_0 e^{\tau_0 - \tau} + \int_{\tau_0}^{\tau} e^{\tau' - \tau} S(\tau') d\tau'}$$

Intrinsic Calculation

Processors



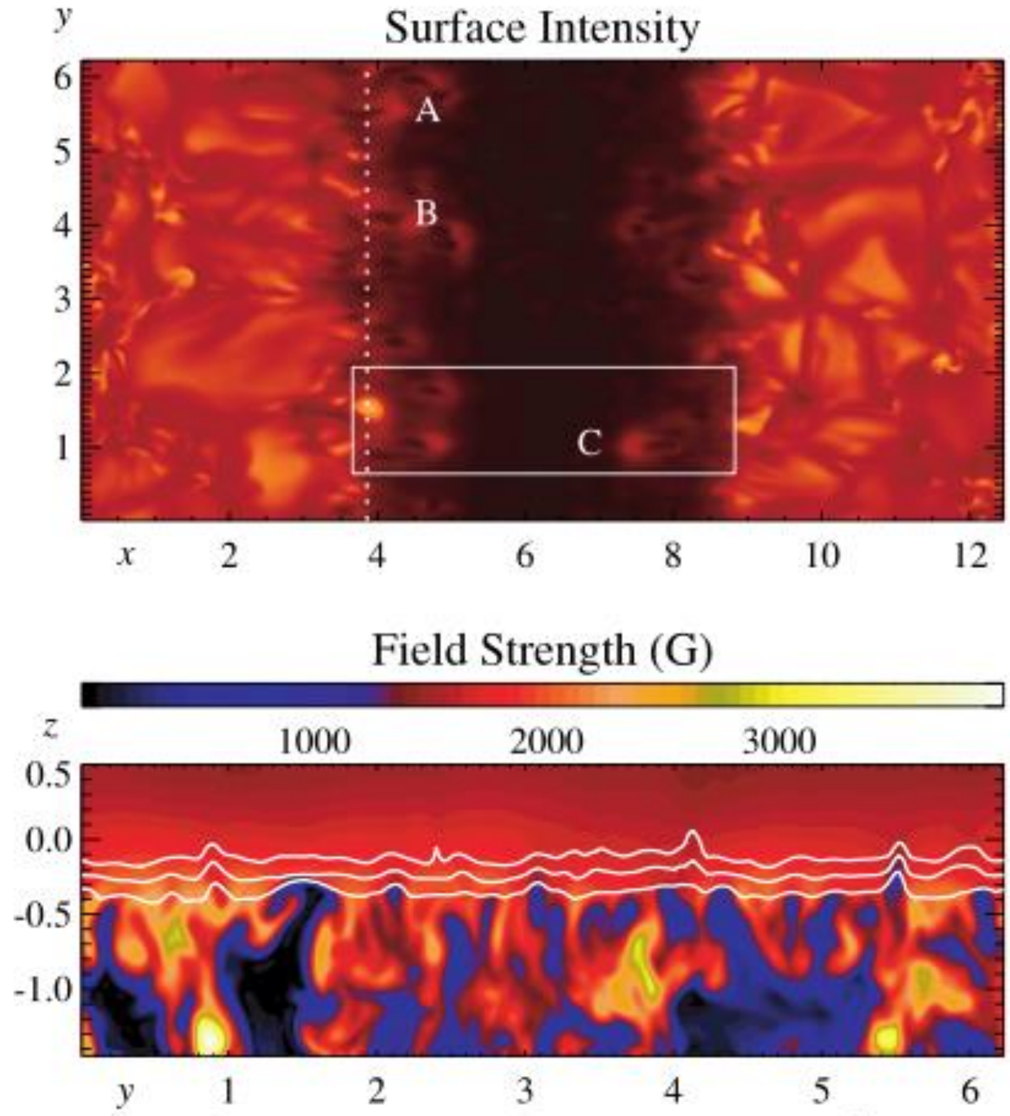
Ray direction

Realistic Solar MHD

2. MHD SIMULATIONS

The simulations were carried out using the PENCIL code,[2] modified to handle energy transfer by radiation in a gray atmosphere (Heinemann 2006; Heinemann et al. 2006). We used a rectangular computational box 12448 × 6212 km in the horizontal (x and y) directions, extending over a depth (z) range of 3094 km and with a grid separation of 24.36 km in both horizontal and vertical directions (512 × 256 × 128 grid points). The quiet Sun photosphere

[2] See http://www.nordita.org/software/pencil-code/.



Title: MHD Simulations of Penumbra Fine Structure

Authors: Heinemann, T.; Nordlund, Å.; Scharmer, G. B.; Spruit, H. C.

Affiliation: AA(Department of Applied Mathematics and Theoretical Physics, Centre for Mathematical Sciences, Wilberforce Road, Cambridge CB3 0WA, UK), AB(Niels Bohr Institute, University of Copenhagen, Juliane Maries Vej 30, 2100 Copenhagen, Denmark), AC(Institute for Solar Physics, Royal Swedish Academy of Sciences, AlbaNova University Center, SE-106 91 Stockholm, Sweden), AD(Max Planck Institute for Astrophysics, Box 1317, 85741 Garching bei München, Germany)

Publication: The Astrophysical Journal, Volume 669, Issue 2, pp. 1390-1394. (ApJ Homepage)

Publication 11/2007

Use temperature

```
###                                      -*-Makefile-*-
### Makefile for modular pencil code -- local part
### Included by `Makefile'
###

MPICOMM    =   mpicomm
HYDRO      =   hydro
DENSITY    =   density
ENTROPY    =   temperature_ionization
MAGNETIC   =   magnetic
RADIATION  =   radiation_ray
EOS        =   eos_temperature_ionization
SHOCK      =   shock
FORCING    =   forcing
GRAVITY    =   gravity_simple
REAL_PRECISION = double
```

Useful units, etc

```
! -*-f90-*-  (for Emacs)   vim:set filetype=fortran:  (f
!
! Convection in vertically stratified atmosphere/solar co
! Initialisation parameters
!
&init_pars
  cvsid='$Id: start.in,v 1.1 2014/07/03 07:34:12 palvi Ex
  unit_length=1e8, unit_velocity=1e5, unit_density=1e0, u
  xyz0=-2.,-2., 0.0
  xyz1= 2., 2., 9.0
  lperi= T, T, F
  lwrite_aux=T
  lwrite_ic=F
/
&eos_init_pars
  lss_as_aux=T, lpp_as_aux=T, lcp_as_aux=T, lcv_as_aux=T,
  xHe=0.1, yMetals=1e-4
  !lHminus_opacity_correction=T
  !lconst_yH=T, yH_const=.99
/
&hydro_init_pars
/
&density_init_pars
  initlnrho='exp_zbot', rho_left=2e-3, Hrho=4.
/
&grav_init_pars
  gravz_profile='const', gravz=-274.
/
&entropy_init_pars
/
&magnetic_init_pars
/
&radiation_init_pars
  bc_rad='p:p','p:p','S:0'
  radx=0, rady=0, radz=1, rad2max=1
  opacity_type='Hminus', scalefactor_kappa=1e-5
/
```

```
! -*-f90-*-  (for Emacs)   vim:set filetype=fortran:  (for vim)
!
! Convection in vertically stratified atmosphere/solar convection zone
! Run parameters
!
&run_pars
  cvsid='$Id: run.in,v 1.1 2014/07/03 07:34:12 palvi Exp $'
  nt=5000000, it1=20, isave=10, itorder=3, cdt=.7
  dsnap=50., dvid=100.
  iz=100
  bcz = 's','s','a','a2','f:a2','s','s','a','s'
  lwrite_aux=T
/
&eos_run_pars
  lss_as_aux=T, lpp_as_aux=T, lcp_as_aux=T, lcv_as_aux=T, lgamma_as_aux=T
  lHminus_opacity_correction=T
/
&hydro_run_pars
  tdamp=1.0
  dampu=10.0
  ldamp_fade=T
/
&density_run_pars
  lreinitialize_lnrho=F, initlnrho='rescale', rescale_rho=1.03
  lupw_lnrho=T
/
&forcing_run_pars
/
&grav_run_pars
/
&entropy_run_pars
  chi=1e-6
  lupw_lnTT=T
/
&magnetic_run_pars
  iresistivity='eta-const','eta-shock'
  lweyl_gauge=F
  eta=1e-2, eta_shock=5.
  va2max_jxb=2500., va2power_jxb=4
/
&radiation_run_pars
/
&viscosity_run_pars
  ivisc='nu-const','nu-shock'
  nu=1e-2, nu_shock=5.
/
&shock_run_pars
  lshock_first=F
/
                                                          1,1          All
```

Basic equations

$$
\begin{aligned}
\frac{\mathrm{D}\ln\rho}{\mathrm{D}t} &= -\boldsymbol{\nabla}\cdot\boldsymbol{u}, \\
\rho\frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t} &= -\boldsymbol{\nabla}(p+\phi)+\rho\boldsymbol{g}+\boldsymbol{J}\times\boldsymbol{B}+\boldsymbol{\nabla}\cdot(2\rho\nu\mathsf{S}), \\
\rho T\frac{\mathrm{D}s}{\mathrm{D}t} &= -\boldsymbol{\nabla}\cdot\boldsymbol{F}_{\mathrm{rad}}+2\rho\nu\mathsf{S}^2, \\
\frac{\partial\boldsymbol{A}}{\partial t} &= \boldsymbol{u}\times\boldsymbol{B}+\eta\nabla^2\boldsymbol{A},
\end{aligned}
$$

Working with temperature

$$p = \frac{\mathcal{R}}{\mu} T \rho, \quad \mathcal{R} = k_{\rm B}/m_{\rm u}$$

Y=0.24: mass fract
$x_{\rm He}$=0.08: vol fract

$$\mu(\rho, T) = \mu_Y / (1 + y_{\rm H} + x_{\rm He})$$

$$\mu_Y = 1/(1 - Y), \quad Y \approx 1/(1 + 1/4 x_{\rm He})$$

$$\rho T \frac{{\rm D}s}{{\rm D}t} = \rho \frac{{\rm D}e}{{\rm D}t} + p \boldsymbol{\nabla} \cdot \boldsymbol{u} = \rho c_v T \left(\frac{{\rm D} \ln T}{{\rm D}t} + \frac{\gamma - 1}{\delta} \boldsymbol{\nabla} \cdot \boldsymbol{u} \right)$$

Pressure gradient

$$\frac{1}{\rho}\nabla p = \frac{c_s^2}{\gamma}(\nabla \ln \rho + \delta \nabla \ln T),$$

$$c_s^2 = \gamma p/\rho\alpha$$

$$\alpha = (\partial \ln \rho / \partial \ln p)_T$$

$$\delta = (\partial \ln \rho / \partial \ln T)_p$$

$$c_p = (\partial e / \partial T)_p$$

$$c_v = (\partial e / \partial T)_v$$

$$\gamma = c_p / c_v$$

Saha equation

$$\frac{y_{\rm H}^2}{1-y_{\rm H}} = \frac{\rho_{\rm e}}{\rho}\left(\frac{\chi_{\rm H}}{k_{\rm B}T}\right)^{-3/2} \exp\left(-\frac{\chi_{\rm H}}{k_{\rm B}T}\right)$$

$$\chi_{\rm H} = 13.6\,{\rm eV}$$

$$\rho_{\rm e} = \mu_Y m_{\rm u}(m_{\rm e}\chi_{\rm H}/2\pi\hbar^2)^{3/2}$$

Opacity: either Kramers or Hminus

$$\kappa = \kappa_0 \rho^a T^b$$

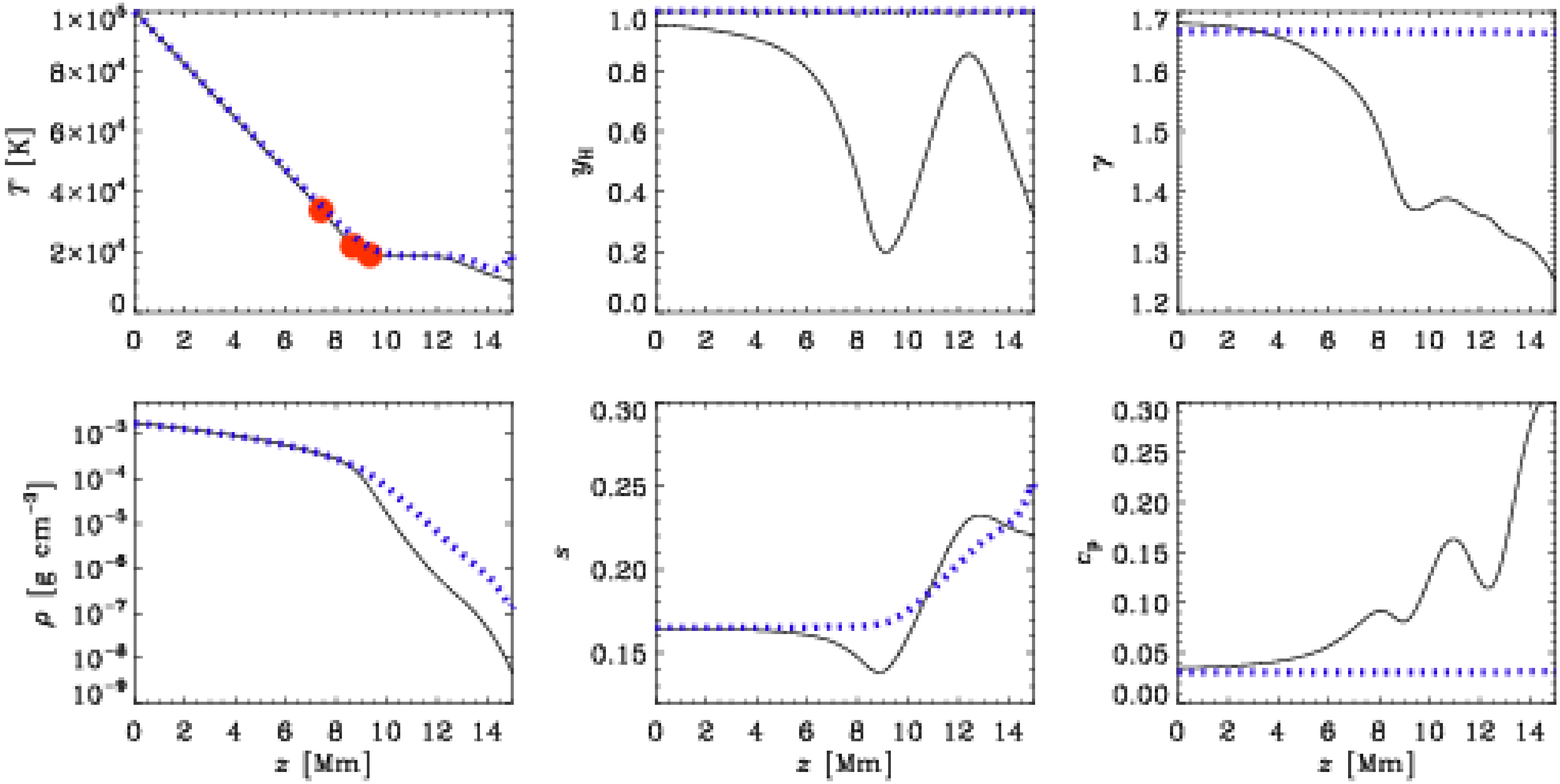
$$\kappa = \kappa_0 (y_{\rm H} + x_Z)(1-y_{\rm H})\frac{\rho}{\rho_{e^-}}\left(\frac{\chi_{\rm H^-}}{k_{\rm B}T}\right)^{3/2} \exp\left(\frac{\chi_{\rm H^-}}{k_{\rm B}T}\right)$$

$$\chi_{\rm H^-} = 0.754\,{\rm eV}, \quad \kappa_0 = \sigma_{\rm H^-}/4m_{\rm u}\mu_Y,$$

$$\sigma_{\rm H^-} = 4\times 10^{-17}\,{\rm cm}^2, \quad x_Z = 10^{-4}$$

$$\rho_{e^-} = \mu_Y m_{\rm u}(m_{\rm e}\chi_{\rm H^-}/2\pi\hbar^2)^{3/2}$$

With/without ionization



Move to Github

The Pencil Code

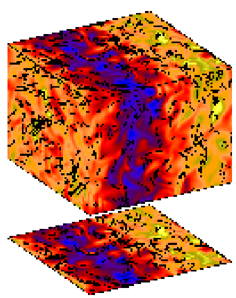
a high-order finite-difference code for compressible MHD

- Home
- News
- Documentation
- Highlights
- Samples overview
- Autotests
- Download
- Meetings
- References
- Contact
- Latest changes ...

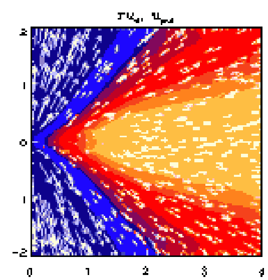
The **Pencil Code** is a high-order finite-difference code for compressible hydrodynamic flows with magnetic fields. It is highly modular and can easily be adapted to different types of problems. The code runs efficiently under MPI on massively parallel shared- or distributed-memory computers.

Attention: Pencil Code has moved to Github! [more details...]

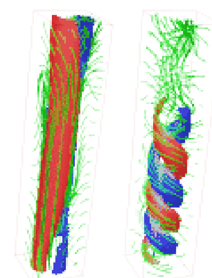
The Pencil Code or equivalent codes have been used for many different applications in a (more or less) astrophysical context. Examples are



Turbulence simulations



Outflows from accretion discs



Dynamo experiments

Available as open source: https://github.com/pencil-code (formerly, before 18 April 2015, under Pencil-Code.googlecode.com)

Pencil News

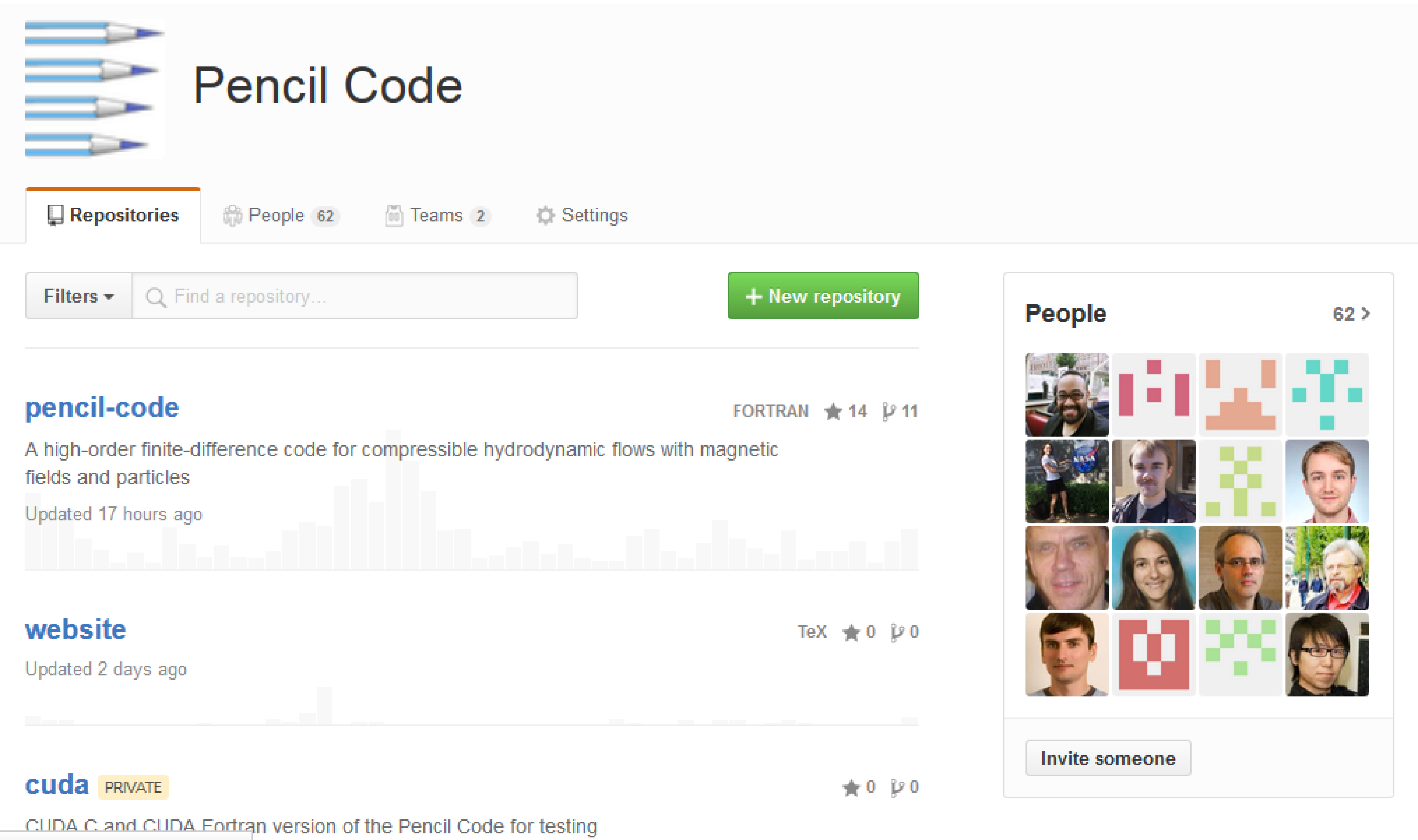
The next Pencil Code User Meeting will be in spring 2015. [more...]

Get Pencil

There are several ways how to get the code. [more...]

Learn Pencil

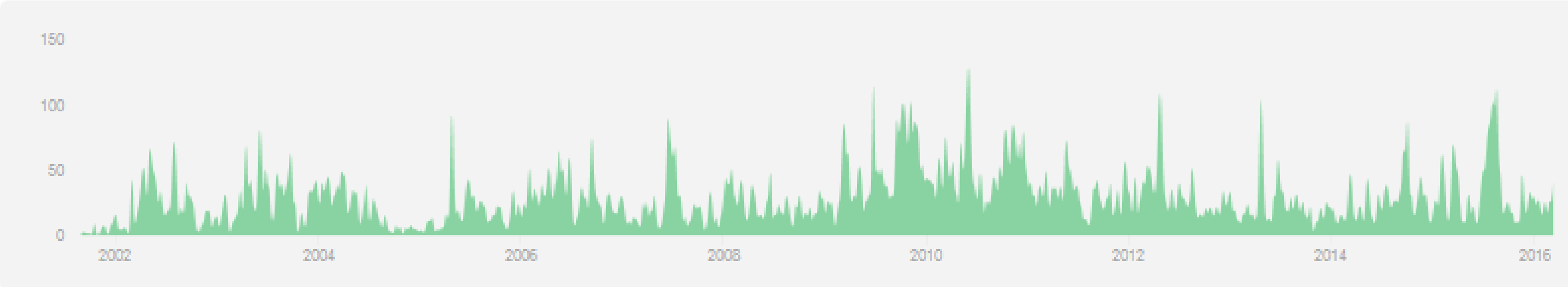
Quick start guide for beginners, samples, manual & [more...]


Pencil Code
Repositories
People 62
Teams 2
Settings
Filters
Find a repository...
+ New repository
pencil-code
FORTRAN 14 11
A high-order finite-difference code for compressible hydrodynamic flows with magnetic fields and particles
Updated 17 hours ago
website
TeX 0 0
Updated 2 days ago
cuda PRIVATE
0 0
CUDA C and CUDA Fortran version of the Pencil Code for testing
People
62
Invite someone

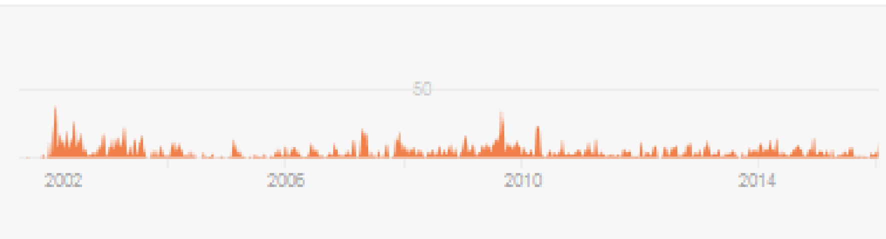
Oct 28, 2001 – May 11, 2016

Contributions to master, excluding merge commits

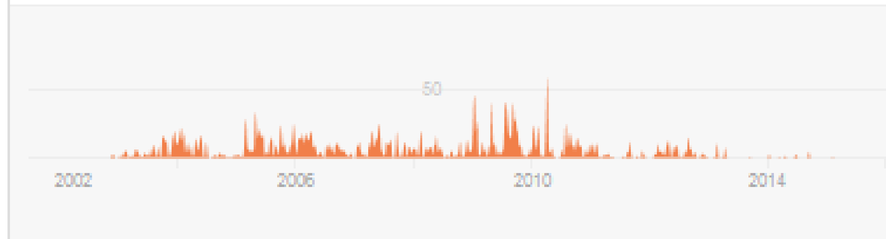
Contributions: **Commits**



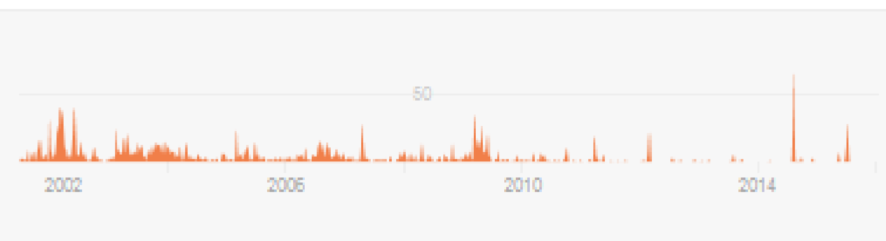
AxelBrandenburg #1
3,828 commits / 281,072 ++ / 132,718 --



ajohan #2
3,632 commits / 181,003 ++ / 135,001 --

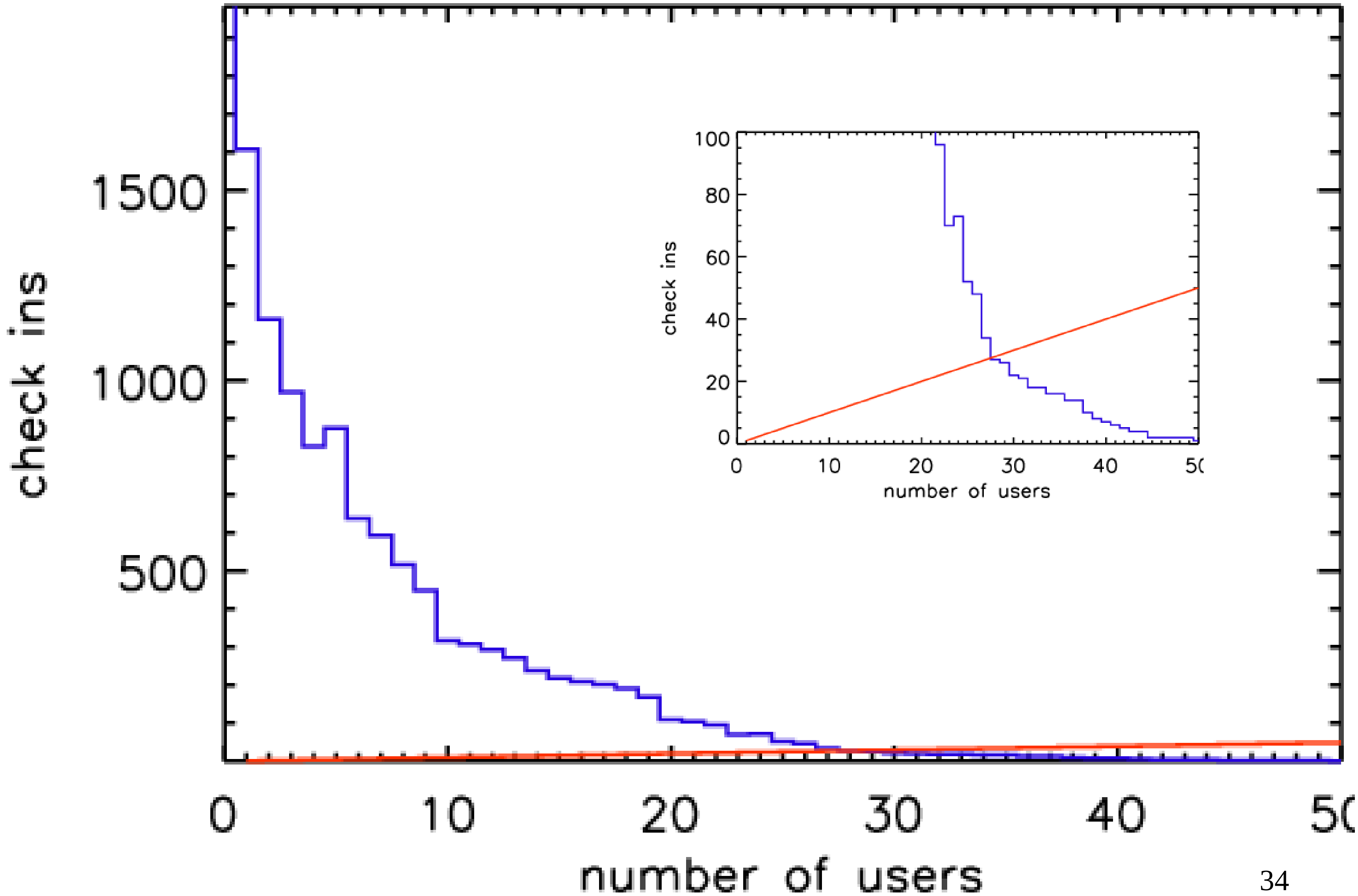


wdobler #3
2,705 commits / 150,741 ++ / 70,329 --



PABourdin #4
2,244 commits / 123,557 ++ / 87,236 --

H-index of check-ins



Automatic validation tests

Pencil Code -- Tests

Automatic test results

To ensure reproducability, the Pencil Code is tested daily for a number of sample applications. This is important for us in order to make sure certain improvements in some parts of the code do not affect the functionality of other parts. For other users who suspect that a new problem has emerged it could be useful to first see whether this problem also shows up in our own tests. The latest test results for a can be seen online:

- opto3 (Linux on 4 x Opteron 2.2GB, ifort 9.1 compiler with MPICH, by Anders Johansen)
- GNU Fortran (Ubuntu 4.4.1-4ubuntu9) 4.4.1 (by Philippe Bourdin)
- Shal (Linux on 2 x Quadcore Intel Xeon E5320@1.86GHz, ifort 64 bits v11.1.064, by Boris Dintrans, regular level 2 test)
- Shal (Linux on 2 x Quadcore Intel Xeon E5320@1.86GHz, ifort 64 bits v11.1.064, by Boris Dintrans, 16 separate tests)
- Linux/Ubuntu10.4 on Intel Core 2 Quad Q9000@2.00GHz, ifort 64bit v11.1 (Sven Bingert, standard + personal tests)
- Nordita Big Test (norlx51, gfortran, openmpi, by Wolfgang/Axel)
- Nordita Hourly Test (norlx51, gfortran, openmpi, by Wolfgang/Axel)
- Nordita PowerMac (os10, g95, ompi, by Axel) [previous]

Note: before checking in your own changes, you should at least do the very minimal auto-test:

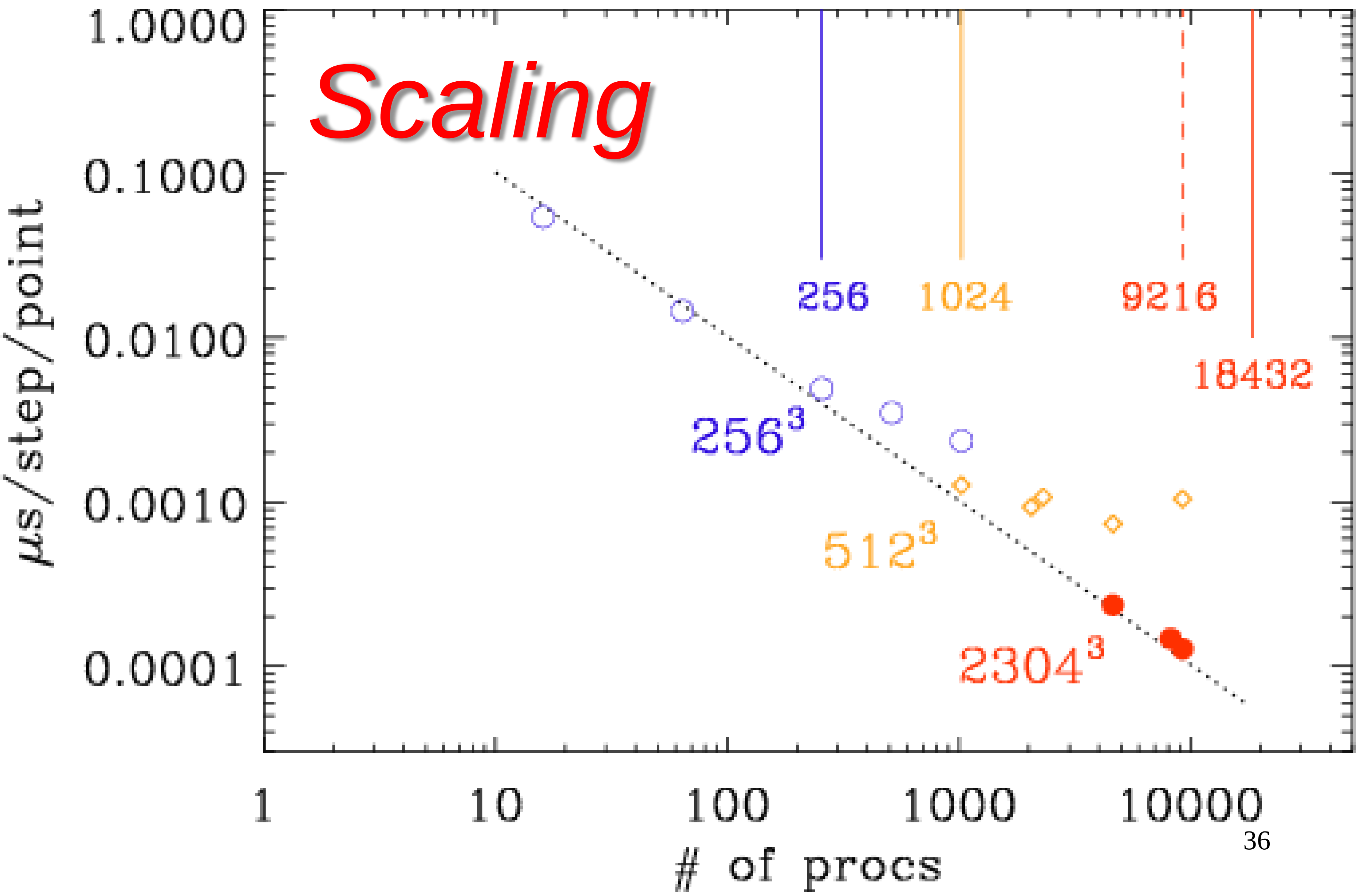
```
pc_auto-test --level=0 --no-pencil-check -C
```

```
/data/bourdin/Korona/Recent/pencil-code/samples/2d-tests/field-loop-fargo: (34/34)
    Compiling..                         ok
        No data directory; generating data -> /tmp/pencil-tmp-philippe-10473
    Starting..                          ok
    Running..                           ok
    Validating results..                ok

----------------------------------------------------------------------
All 34 tests succeeded.
```

```
------------------------------------------------------------------------
### auto-test failed ###
Failed 2 test(s) out of 48:
  /home/brandenb/pencil-weekly-tests/samples/corona (compilation)
  /home/brandenb/pencil-weekly-tests/samples/most-modules (running)

CPU time (including compilation): 02:29:20u 30:35s
Total wall-clock time:            02:56:19 = 01:45:10 + 01:09:12
Maintainers of failed tests: anders/astro:lu:se,wlyra/amnh:org,nbabkovska
```

Wed Apr 6 14:46:39 2011

Scaling



1.0000
0.1000
0.0100
0.0010
0.0001
μs/step/point
256
1024
9216
18432
256³
512³
2304³
1
10
100
1000
10000
# of procs

For this afteroon

Numerical simulations of Kelvin-Helmholtz instability: a two-dimensional parametric study

Chunlin Tian, Yao Chen

(Submitted on 6 Apr 2016)

Using two-dimensional simulations, we numerically explore the dependences of Kelvin-Helmholtz instability upon various physical parameters, including viscosity, width of sheared layer, flow speed, and magnetic field strength. In most cases, a multi-vortex phase exists between the initial growth phase and final single-vortex phase. The parametric study shows that the evolutionary properties, such as phase duration and vortex dynamics, are generally sensitive to these parameters except in certain regimes. An interesting result is that for supersonic flows, the phase durations and saturation of velocity growth approach constant values asymptotically as the sonic Mach number increases. We confirm that the linear coupling between magnetic field and Kelvin-Helmholtz modes is negligible if the magnetic field is weak enough. The morphological behaviour suggests that the multi-vortex coalescence might be driven by the underlying wave-wave interaction. Based on these results, we make a preliminary discussion about several events observed in the solar corona. The numerical models need to be further improved to make a practical diagnostic of the coronal plasma properties.

Comments: 11 pages, 5 figures, 3 tables. Accepted in ApJ
Subjects: **Solar and Stellar Astrophysics (astro-ph.SR)**
Cite as: **arXiv:1604.01546 [astro-ph.SR]**

We numerically solve the following resistive MHD equations using the PENCIL CODE[1], which is an open source, modular high-order finite difference code. It is of sixth-order accuracy in space and third-order in time by default.

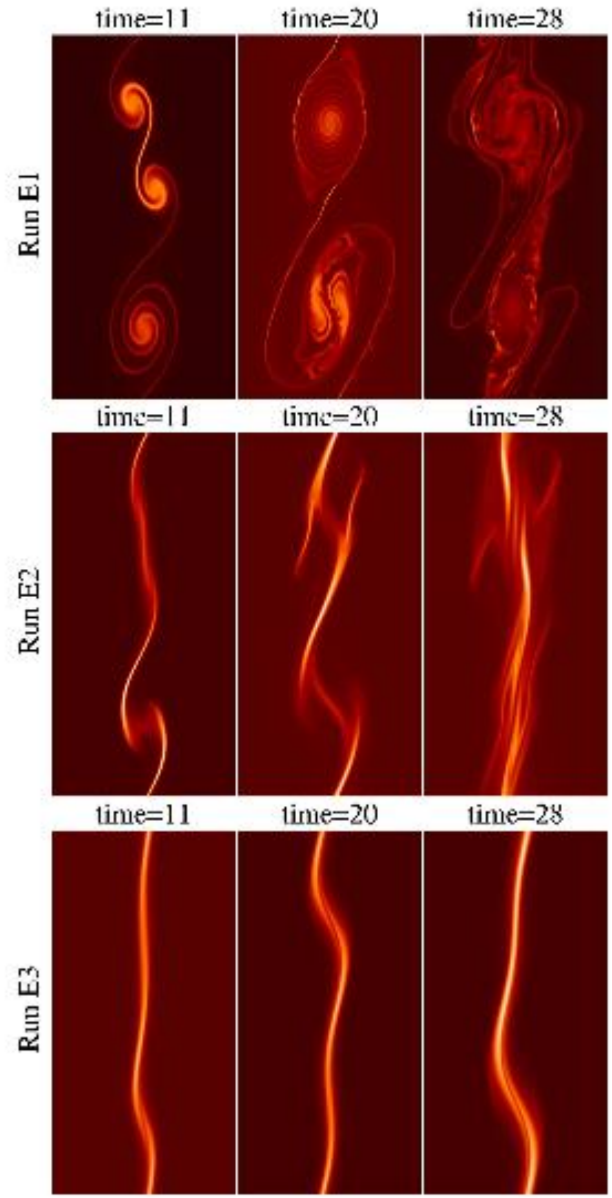


Figure 4. Snapshots of specific entropy taken during multi-vortex evolving phase for selected cases with different effective Alfvénic Mach number. Run E1: $M_{A,y} = 50$; E2: $M_{A,y} = 3.75$; E3: $M_{A,y} = 2.5$.

Roadmap

The Yin-Yang mesh

```
!
! MODULE_DOC: This module contains Yin-Yang related types and functions
! MODULE_DOC: which are incompatible with FORTRAN 95.
!
!***************************************************************
!
module Yinyang
!
  use Cdata, only: iproc_world

  include 'yinyang.h'

  type ind_coeffs
    integer, dimension(:,:,:), allocatable :: inds
    real, dimension(:,:,:), allocatable :: coeffs
  end type

  contains

!***************************************************************************
   subroutine bilin_interp(indcoeffs, ith, iph, f, buffer, i2buf, i3buf)
!
!  Performs bilinear interpolation for a pencil at position (ith, iph) of th
!  original strip from values in f-array using the precalculated weights
!  in indcoeffs%coeffs. Result is returned in buffer(i2buf,i3buf)=(ith,iph)
!  or buffer(i2buf,i3buf)=(iph,ith), the latter if transposition is required
!
!  20-dec-15/MR: coded
!
      use General, only: transform_spher_cart_yy, notanumber
```

```
    endsubroutine bilin_interp
!***********************************************************************
    function prep_bilin_interp(thphprime,indcoeffs,th_range) result (nok)
!
!  For each of the points in the strip thphprime (with shape 2 x thprime-extent x
!  phprime-extent), arbitrarily positioned in the yz-plane, determine in
!  which cell of the grid y(ma:me) x z(na:ne) it lies, store indices of the
!  cells upper right corner in indcoeffs%inds and the weights of bilinear
!  interpolation for the four corners in indcoeffs%coeffs. If no cell is found
!  for a point, indcoeffs%inds and indcoeffs%coeffs are set zero.
!  If present, return in th_range the interval in thprime-extent in which
!  interpolation cells could be assigned to any points.
!  Returns number of points in thphprime for which interpolation cell could be
!  found.
!
!  20-dec-15/MR: coded
!
      use General, only: find_index_range_hill
      use Cdata,   only: y, z, lfirst_proc_y, ipy, lfirst_proc_z, ipz
      use Cparam,  only: m1,m2,n1,n2

      real, dimension(:,:,:),          intent(IN) :: thphprime
      type(ind_coeffs),                intent(OUT):: indcoeffs
      integer, dimension(2), optional, intent(OUT):: th_range

      integer :: nok

      integer :: ip, jp, indth, indph, sz1, sz2, ma, na, me, ne, i1, i2
      real :: dth, dph, dthp, dphp, qth1, qth2, qph1, qph2
      logical :: okt, okp

      sz1=size(thphprime,2); sz2=size(thphprime,3)

      if (allocated(indcoeffs%inds)) deallocate(indcoeffs%inds,indcoeffs%coeffs)
      allocate(indcoeffs%inds(sz1,sz2,2))
      indcoeffs%inds=0
```