Fresh Outlook in Numerical Methods for Geodynamics

Preprint · January 2021

4 authors, including:

Gabriele Morra
University of Louisiana at Lafayette

Matthew Knepley
University at Buffalo, The State University of New York

Fresh Outlook in Numerical Methods for Geodynamics—Part 2: Big Data, HPC, Education

Gabriele Morra, Department of Physics, University of Louisiana at Lafayette, Lafayette, LA, United States

David A Yuen, Department of Applied Physics and Applied Mathematics, Columbia University, New York, NY, United States

Henry M Tufo, Department of Computer Science, University of Colorado at Boulder Engineering Center, Boulder, CO, United States

Matthew G Knepley, State University of New York, Buffalo, NY, United States

Email addresses: gabrielemorra@gmail.com (G. Morra); daveyuen@gmail.com (D.A. Yuen); henry.tufo@colorado.edu (H.M. Tufo)

Big Data

Introduction

Geodynamic simulation has required high performance computing from its beginning (**Glatzmaier et al., 1990**; **Bercovici et al., 1988**; **Malevsky and Yuen, 1992**). Greater computer power has often meant new discoveries. With the same spirit for HPC, we should introduce Machine Learning in our scientific research, combining the creation of hybrid models between modeling and big data. In fact, the volume of the output of numerical modeling is already beyond the capacity of human brain power and human memory. To perform data analysis successfully in the 21st century, how can a young graduate student remember the dynamics of 500 time-dependent runs without the proper tool?

Data Assimilation Methods

Attempts at inverting the equations controlling mantle flow have involved the assimilation of data into numerical models of mantle convection. This strategy has been used in a variety of settings such as solar flares (**Bélanger et al., 2007**) and shallow water sedimentation (**Bélanger and Vincent, 2005**). **Bunge et al. (2003)** and **Hier-Majumder et al. (2005)** were the pioneers in using the adjoint methods to invert the momentum and energy equation for mantle convection, such as slabs and plumes starting from a reasonable guess of present mantle conditions based on seismic velocity and thermal model as initial conditions.

Liu and Gurnis (2008) have shown that a similar result can be obtained using a simple backward integration (SBI), first-order time-stepping besides assimilating present-day seismic structure as a constraint and assuming dynamic topography and its rate of change to simultaneously inverting, lessening the computational expense compared to adjoint methods. Using this technique, **Liu et al. (2008)** showed that seismic anomalies in the Farallon slab coincide paleo-geographically with the restored positions of the Shatsky and Hess conjugate plateaux as they subducted and are responsible for the Laramide phase of mountain building in North America (**Liu and Gurnis, 2010**). The same group has introduced adjoint-based inversion of time-dependent mantle convection with nonlinear viscosity (**Li et al., 2017**; **Ratnaswamy et al., 2015**).

This data-assimilation approach was pioneered in geophysics by seismologists, who perform data assimilation models routinely used for inverting seismic data, e.g., (**Fichtner et al., 2006**). A standard approach makes use of the backward seismic wave propagation using the Spectral Element (SE) Method (**Komatitsch et al., 2005**) using the adjoint method (**Luo et al., 2009**). Following this method, synthetic seismograms have been calculated based upon spectral elements and compared to data from the Global Seismographic Network. This approach allows the use of the remaining differences between the measured and synthetic seismograms to constrain the seismic source and compare the validity of alternative Earth models (**Stein and Wysession, 2003**). The adjoint Methods naturally offer a computational approach for solving complex inverse problems (**Tenorio, 2017**), but accurate higher-order time-stepping is recommended.

Application to geomagnetism

Probably the biggest grand challenge problem in geophysics involves the geodynamo simulations of the outer core since the set of geodynamo equations involves the Maxwell equations, variable electrical conductivity (due to its temperature dependence) as well as the finite Prandtl number momentum equation with the contribution of the Coriolis force, and the energy equation which contains ohmic dissipation with a temperature-dependent electrical conductivity (**Glatzmaier, 2013**). These partial differential equations are more numerous and more nonlinear than those associated with mantle convection.

The increase in data quality and quantity during the last decade of magnetic observations of the Earth from space, and the improvement of the numerical description of core dynamics have generated a new type of data assimilation methods applied to the dynamics of the core (**Fournier et al., 2010**).

The state of the art in developing stable, parallel, fast 3D convection for modeling the Magnetic Hydro-dynamics equations of the geodynamo and of star interiors is the code Rayleigh, a community project, with an active Github page. Physically, it is based on the solution of incompressible and inelastic MHD equations in spherical geometry using spherical harmonics in the horizontal direction and Chebyshev polynomials in the radial direction. Rayleigh is developed on MPI and has shown highly efficient strong scaling on 131,000 cores (Mira Blue Gene/Q supercomputer), with 2048^3 grid points (**Matsui et al. 2016**).

Machine Learning

On May 11, 1997, the world chess champion Garry Kasparov lost, under tournament conditions, against Deep Blue, an intelligent machine built by IBM (**DeCoste, 1998**). While before the match, the public did not have the perception of the importance of that event (**Levy, 1990**), in retrospect, that match was the beginning of a new epoch. It took exactly 20 years before computers could crack the last table game in which humans were superior, when AlphaGo beat the world reigning Go champion (**Kasparov, 2018**). Go is a game so complex that its theoretical organization was never formalized and Go top players are driven by intuition and have been trained by sitting in front of a master and learning by practicing the game.

While Deep Blue could be programmed using very powerful linear programming (developing an idea, implementing smart algorithms, debugging, testing), AlphaGo was conceived and programmed to efficiently mimic the human cognitive system to be able to reach hyper-human performance. It is possible that AlphaGo has invented a new Go playing style, or that it has learned to "feel" the next good move. Instead of being directly programmed, AlphaGo self-trained for only a few months, mostly playing against itself.

Today artificial intelligence tools can recognize which animal is in a picture, and even identify a dog's breed, which is a feature superior to many humans. Similarly, in geodynamics, and more in general in computational geosciences, we see the emergence of deep learning tools that interpret data, as well as generating artificial datasets (GANs).

The solution of PDEs using machine learning is equally reaching new frontiers. Already several tools exist for just-in-time compilation of code generated from abstractions. An example in geophysics is the Devito Project in which they are applied to industry seismic imaging (**Louboutin et al., 2019**) with the focus on seismic as well as on fluid flow. Recently, automatic differentiation is going through major innovations using Machine Learning, for example with Tensorflow and PyTorch (Witte et al., 2019a,b). We note here that Tensorflow and PyTorch are just general libraries, which serve as a starting point. To go further, for modeling modern approaches, especially in graph computing, one needs to write dedicated software, or use specialized matrix software such as SMASH in linear algebra (**Cai et al., 2018**).

Supervised machine learning

In Supervised Machine Learning, models are built based on labeled training data and then used to predict results on previously unseen data of the same class. In Geophysics, multiple ML algorithms have already been applied, starting from simple ones (e.g., Logistic Regression) up to Deep Learning tools such as Neural Networks. Although more complex models can perform exceptional tasks and need less parameter tuning, in general they require more training (**Bergen et al., 2019**).

Many supervised machine learning implementations need to be trained through a set for classification, which implies that the accuracy of the training set is essential in the entire process. The term "ground truthing" has been introduced to indicate the research of objectively well labeled data. In some fields, the "ground truth" has been put forward, e.g., in real-world networks (**Yang and Leskovec, 2012**). In geophysics, this is a well-known problem, for which pioneering solutions have been proposed for example for the Magellan SAR images of Venus (**Burl et al., 1998**), with the purpose of avoiding bias of expert labelers in detecting small volcanoes. This problem is very open and general in Machine Learning (**Sheng et al., 2008**; **Goodfellow et al., 2016**) and will require opening training data to the community and discussions on labeling.

Logistic regression

Logistic regression, the first type of regression analysis to be studied rigorously, and to be used extensively in practical applications, is likely the simplest of the classifying algorithms. It returns the probability that data belongs to one among two classes (**Cox, 1959**; **Puntanen, 2010**).

Logistic Regression has been applied often in Geophysics, before the recent growth of the ML method. For example it has been used to test the relationship between hydraulic conductivity and geophysical properties **(Chen et al., 2001)**.

More recently, in seismology, it has been used to distinguish automatically between earthquake signals and explosions, and to classify time segments as either earthquakes or noise (**Reynen and Audet, 2017**). Similarly, it has been used to classify induced seismicity in injection wells, separating aseismic from seismogenic injection wells (**Pawley et al., 2018**).

Support vector machine

One of the main problems in using techniques belonging to Computational Intelligence to detect events such as volcanic eruptions and earthquakes, or predict them in the future, lies in the many false detection/classifications, in particular when using very complex methods implemented from partially understood deep learning algorithms. Support Vector Machine (SVM) is instead a traditional and well tested methodology that has been extensively tested in many diverse scientific fields. **Ruano et al. (2014)** for example compare multi-layer perceptrons (MLPs) and support vector machines (SVMs) for the detection of seismic events, finding superior results with SVMs. In geophysics, SVMs have been applied to volcanic tremors (**Masotti et al., 2006**), SAR image classification (**Fukuda et al., 2002**), and global geomagnetic disturbance (**Ji et al., 2013**, where NN was found to be superior to SVM).

In order to accelerate the learning phase of SVM, new strategies have appeared recently, among them the "algebraic multi-grid support vector machines" (**Sadrfaridpour et al., 2016**). We suggest that the use of SVM to categorize and classify geodynamic models outcome is a simple way in which geodynamicists could start merging their computational tools with machine learning.

Random forests

Random forests is yet another method for classification, regression and similar tasks. The conceptual idea is to construct a multitude of decision trees during training and to associate a tree with the target to classify, or predict, if regression is searched. The method was introduced as "Random Decision Forests" by Tin Kam Ho of the AT&T Laboratory Labs for classifying handwritten digits.

Random Forest has been used with great success in geophysics to analyze laboratory quakes, both obtained with sliding rocks under controlled conditions (Rouet-Leduc et al., 2017, 2018; Hulbert et al., 2019) and with the study of analog laboratory quakes using gel (**Corbi et al., 2019**). The application of RF to natural seismic data of Random Forest has not yet shown an equivalent success, possibly due to the seismic noise constantly present, and also to the impossibility of collecting the seismic signal right on the fault, as possible under controlled laboratory conditions. Up till now success has been obtained in the analysis of slow slip events, by connecting seismic tremors with GPS data (**Rouet-Leduc et al., 2018**).

Other applications

Supervised Machine Learning has been applied to numerous problems at the boundary between seismology and geodynamics. This natural time/space scale emerges from the extensive dataset available in seismology and the great variety of geodynamic models that realistic physical parameters offer. For example **Schäfer and Wenzel (2019)** applied a variety of machine learning techniques to analyze the relationship between 76 subduction zone segments worldwide and maximum magnitudes and return periods for these zones, finding that almost all major subduction zones have the potential to produce earthquakes of Mw above 8.5 and a correlation between maximum magnitude and geometry of the subduction zone.

Deep Learning and Neural Networks

A standard neural network (NN) consists of layers of many connected processors called neurons, each producing a sequence of real-valued signals (**Schmidhuber, 2015**). Input neurons are activated by either sensors connected to the environment or by previously recorded data (e.g., images, time series). The neurons are activated by weighted connections to the active neurons of the previous layer.

The development of deep learning (DL) through Neural Network has a long history that cannot be summarized here. In the last years NNs have become the mainstream ML tool due to the development of very advanced back propagation (BP) techniques. In fact, early NN architectures did not learn (**McCulloch and Pitts, 1943**). Until the 1970s, NNs had only a single layer (SL) or were used for simple Unsupervised Learning tasks. The perceptron algorithm (**Rosenblatt, 1957**) was used for the first time to obtain deep learning (DL) at the end of the 1960s (**Ivakhnenko and Lapa, 1965**) with a Feed-forward Multilayer approach. The introduction of Convolutional NNs (CNNs, also called Convnets) arrived much later with the introduction of the Neocognitron (**Fukushima, 1979**), which is the first to resemble the modern CNNs that have won so many contests and now dominate the ML scene. They are supervised, feedforward, gradient-based Deep Learners (**Schmidhuber, 2015**).

Fundamental for this development, BP was introduced and refined in the 1960s and 1980s. In the 1960s already efficient BP algorithms were introduced, however backpropagation was based on the calculation of the Jacobian matrix from one layer to the previous one only. Famously, a book (**Minsky and Papert, 1969**) on the limitation of the single layer perceptron discouraged many researchers from investigating the potentials of NNs. The first efficient error BP algorithm applicable to NN was described in a Master's thesis (**Linnainmaa, 1970**), together with its Fortran implementation. Its application to NN emerged only in the 1980s (**Werbos, 1982**; **Parker, 1982**), since when new powerful BP algorithms have allowed using NN at unprecedented depth, up to tens of layers, starting a revolution in the field. As soon as multilayer NNs began to be adopted due to increased computer power in the 1980s, extraordinary results in image processing, feature detection and many other tasks have emerged.

In seismology, in the past decade, many published studies have reported the similarity of waveforms between nearby sources for detecting seismic events. Many small earthquakes, or events far from installed stations, are undetected, therefore automatic tools can substantially increase the list of known earthquakes. This has been done using template matching (**Ross et al., 2017**; **Beaucé et al., 2018**; **Chamberlain et al., 2018**), convolutional neural network (e.g., GPD, generalized phase detection, **Ross et al., 2018**), dictionaries (e.g., FAST, Fingerprinting and Similarity Thresholding, **Yoon et al., 2015**; **Bergen and Beroza, 2019**). Of all these methods, CNN and Sparse Representation (dictionaries) presently compete as the most powerful tools.

For picking single phases in seismology, deep learning has been the most successful tool up to now. For example, **Zhu and Beroza (2018)** used CNN to pick P- and S-wave arrival times starting from a > 10^6 manually picked seismograms, even providing a mechanism to estimate the quality of the picks. **Ross et al. (2018)** also used CNN as a regressor to predict the time index of the P-wave phase onset and another to detect first-motion polarities of P waves. Both these works have achieved super-human performance, above professional seismic analysts. **Zhu et al. (2019)** used 30,146 labeled phases to analyze the aftershock sequences of the 2008 Mw7.9 Wenchuan Earthquake in Sichuan, obtaining a five-times improvement over the ObsPy AR picker; most importantly, when tested on a small dataset from a different region (Oklahoma, United States), their algorithm achieved 97% accuracy after fine tuning only the fully connected layer of the model. **Wang et al. (2018)** employed deep-learning methods for analyzing marine seismic data.

Geodynamicists have only started their adventure in the world of Machine Learning. For example, **Shahnas et al. (2018)** employed support vector machine (SVM) trained with snapshots taken from 2000 time-dependent runs of numerical convection models to estimate the magnitude of the spin transition-induced density anomalies in the deep mantle. In order to apply it to exoplanet research, **Baumeister et al. (2019)** use mixture density neural networks (MDNs) to infer the distribution of possible thicknesses of a planetary layer (mantle, core) from mass and radius of the planet. In general, deep NN can be used to solve general high-dimensional stochastic elliptic partial differential equations, therefore the technical possibility to develop simulator-free solutions for standard geodynamics problems is in sight (**Karumuri et al., 2019**).

Deep NNs suffer vanishing gradient and degradation problems, which has prompted attempts at developing "shorter" architectures. For example, recently introduced architectures are residual NN (ResNet, WideResNet), dense NN (DensNet), fractal NN. The research in the field in Deep Learning is rapid and this review can only partially describe its development.

Hybrid traditional and neural network methods

To combine traditional numerical approaches and new Machine Learning tools is one of the main challenges, if not the principal one, that computational geophysicists and geodynamicists have to solve today. One successful example is in climate supermodeling, where a new ML driven assimilation scheme has been applied to better synchronize a large climate model (atmosphere, ocean) to improve climate predictions, in which ML is used to find coupling factors between the submodels (**Selten et al., 2017**).

Another important application that has just emerged for NNs is automatic differentiation. This feature exploits the extraordinary ability of NNs to minimize highly non-linear cost-functions through the process of gradient descent. NNs do so by computing partial derivatives accurately and quickly. Recent applications have shown how this process can be used to find the leading equations from a set of noisy spatio-temporal data (**Both et al., 2019**).

Radial basis functions and neural networks

Radial basis functions (RBFs) is a powerful numerical methodology originally developed by geophysicists in the early 1970s to fit rugged topographical data in rugged terrains. RBFs have been developed since the late 1990s for solving PDEs to high accuracy in any number of dimensions (**Fornberg and Flyer, 2015**). They have been employed also in 3D mantle convection (**Wright et al., 2010**) and for weather forecasting problems in geophysical fluid dynamics (**Flyer and Wright, 2009**).

RBFs are also used in neural networks. One implementation of RBF NNs is a three-layer neural network with only one hidden layer, including a nonlinear relationship between the input layer and the hidden layer, and a linear relationship between the hidden layer and the output layer. Because of the few number of layers, RBF neural networks have a faster learning speed and stronger nonlinear approximation ability (**Martin et al., 2017**). This feature provides a good nonlinear forecasting model. Within the framework of

RBF neural networks, the input layer node transmits signal to the hidden layer. Hidden layer nodes are composed of radial action functions such as the Gaussian kernel functions, performing a nonlinear transformation through the basis function to map the input space to a new space (**Gan et al., 2010**). The output layer node realizes the linear weighting combination in the new space. RBF neural networks are a local approximation that can determine the corresponding network topology according to each individual problem. The advantages of RBF neural networks over other networks such as CNNs and RNNs are: fewer layers, smaller memory requirements, and faster computational speed (**Steve et al., 2002**). Only a few weights need to be adjusted for each input and output data. Therefore, RBF has the advantages of deep learning, global approximation and optimal approximation performance. However, RBF neural networks need to be tested on supercomputers to demonstrate their true worth.

Deep generative networks

One of the most interesting trends in the field of Deep Neural Networks is the reverse use of NN. Generative adversarial networks (GANs) are obtained by combining two neural networks with different objectives. The first network learns from training data and produces synthetic data, while a second NN learns to distinguish between the real data and the synthetic data produced by the first. The two networks are combined through back-propagation signals in order that the first network must improve at creating more realistic synthetics and the second at its discriminatory abilities.

Numerous applications have already appeared, for example involving (synthetic) images of people's faces and animals (**Creswell et al., 2018**). Applications in geosciences are still limited, with several applications in seismology but also in generating samples for modeling. For example 3D synthetic porous media have been generated using GANs (**Feng et al., 2019**; **Liu et al., 2019**; **Mosser et al., 2017**) and in particular for limestone (**Mosser et al., 2018**).

Unsupervised Learning

Clustering: Nearest neighbor

Clustering has been used extensively in seismology. For example, for hypocenter relocation. Several techniques can improve the accuracy of initially located earthquakes (e.g., **Jones and Stewart, 1997**; **Karasözen et al., 2016**). Cross-correlation of waves for similar clustered events has been used for relocation (e.g., **Poupinet et al., 1984**; **Shearer, 1997**). Recently, Hierarchical Clustering was used in GrowClust (**Trugman and Shearer, 2017**) for earthquake relocation. Clustering has also been applied to various earthquake features to investigate earthquake processes and precursors (**Dzwinel et al., 2005**; **Yuen et al., 2009**).

Several other applications of clustering exist. For example, **Leśniak and Isakow (2009)** clustered hypocenters using several algorithms based on location and time of occurrence for assessing seismic safety in mines. **Rietbrock et al. (1996)** used waveform clustering to find about 20 different clusters of earthquakes in the Gulf of Corinth. Moment tensor based clustering has also been used (**Cesca, 2020**). **Peng and Zhao (2009)** used the *match pattern* of the waveform of thousands of relocated earthquakes at Parkfield to identify 11 times more aftershocks than listed in the previous standard catalogue of the Northern California Seismic Network.

Fuzzy clustering (**Chen, 2018**) was used to pick seismic-wave arrival times, by using as sets the amplitude of the seismic data before and after the arrival. Clustering was then used to develop a decision boundary.

Standard regression

A recent attempt at combining regression with deep learning was attempted by **Iturrarán-Viveros (2012)** who used the regression results to aid in the construction of Neural Network models to predict porosity from seismic data.

Quite surprisingly, recent applications of a sparse regression algorithm have shown an ability to discover unknown partial differential equations from a set of data. By using traditional differentiation from well refined data, regression allowed the PDE of equations such as Navier Stokes, Korteweg De Vries, Reaction Diffusion to be found (**Rudy et al., 2017**).

Bayesian approaches

In geosciences, the holy grail of numerical models is to develop the ability to predict future behavior from previous data. Due to the complexity of the Earth Systems, many geophysical models can only describe past observations but not predict future geophysical events. Predictions are effectively normally obtained through a variety of sophisticated statistical tools, part today of the large umbrella called Machine Learning. For example, hidden Markov models (HMMs), Dynamic Bayesian networks, Bayesian hierarchical modeling, Random Forest, and others have all been used for probabilistic prediction.

A Bayesian Network (BN), a generalization of Kalman filters and Hidden Markov Models, is a high level representation of a probability distribution over a set of variables that are used to build models of a specific problem domain (Mittal and Ankush, 2007). The network can be represented by a graph where branches are associated with defined probabilities. In contrast to deep learning technologies, the details of graphical models are intuitive and easier to understand by humans. Their range of applications is very large, due to their flexible structure.

Bayesian hierarchical modeling (BHM) is a statistical model written in multiple levels (hierarchical form) that estimates the parameters of the posterior distribution using the Bayesian method. In exploration geophysics, BHM has been applied to determine the connectivity from time-series measurements of subsurface reservoirs controlled by convection-diffusion physics (**Denli and Subrahmanya, 2014**). An important application of hierarchical graphs is the alleviation of computational resources requirement by reducing the problem size (see sections on multi-grid for more examples).

A Dynamic Bayesian Network (DBN) is a Bayesian network which relates variables to each other over adjacent time steps. It has been successfully used to construct 3D maps of buried underworld infrastructure from multiple sensors, using vibro-acoustic sensors and Ground Penetrating Radar (**Dutta et al., 2013**). In Seismology, it has been applied to the enhancement of earthquake detection, based on a time-frequency decomposition framework (**Riggelsen and Ohrnberger, 2014**).

Markov Chains Monte Carlo (MCMC) is a class of algorithms of which the Metropolis is the most famous; they are widely used in sciences as different as statistics, econometrics, physics and computing science (**Andrieu et al., 2003**). In mathematics, the fastest way to compute the volume of a complex convex shape in a high number of dimensions D is MCMC, with a time that increases as a polynomial in D (**Jerrum & Sinclair, 1996**). The most popular MCMC method is the Metropolis-Hastings (MH) algorithm (**Hastings, 1970**; **Metropolis et al., 1953**). **Malinverno (2002)** used MCMC to establish the number of layers that a 1D Earth interior model of subsurface properties needs to best describe a set of available data. A modern approach that solves some of the issues of tractability of the MCMC is in **Oware et al. (2019)** who apply it to hydrogeological problems.

Hidden Markov Models (HMMs) are the simplest dynamic Bayesian networks. They are applied to sequential data and used in several scientific fields such as computing perception (**Brand et al., 1997**) and with great success to visual and speech recognition (**Rabiner and Juang, 1992**). In geophysics, HMMs have been employed to classify a variety of signals in volcano-seismology (**Beyreuther et al., 2008**) and induced earthquakes (**Beyreuther et al., 2012**). **Hammer et al. (2013)** use the advantage of HMMs to start classifying immediately after the first interesting event is identified, contrary to Neural Networks and Support Vector Machine, for which a large training set is necessary before the network is trained. This enables, at the same time, a reduction of required workload and the potential detection of very rare events (**Dammeier et al., 2016**).

In fact HMMs are a subclass of Graphical Models (**Andrieu et al., 2003**; **Wainwright and Jordan, 2007**), which are deployed in large-scale data analysis problems, such as in bioinformatics. Graphical models form a large area in which statistics and computer science meet. Although they are commonly applied to problems where a Bayesian paradigm is chosen, the graphical model formalism is agnostic between frequentist and Bayesian statistics. Graphical models are “inference engines” (**Cowell, 1998**) in which the hierarchical latent variable models are easy to represent and manipulate. Applications to geophysics are for example in **Srinivasan et al. (2018)**, in which a three-dimensional discrete fracture network (DFN) model is developed by selecting only the primary flow sub-network and only simulate flow and transport thereon, based on a priori graph network analysis. This approach has been used in environmental research as well, to map the subsurface (**Wainwright et al., 2014**). A very powerful technique to apply graphical models to very large problems is stochastic variational inference (**Hoffman et al., 2013**).

Dictionary methods

Since the pioneering paper of **Wright et al. (2009)** which focused on face recognition, there have been many thousands of works applying sparse representation to science and technology. Recent attempts have appeared in seismic tomography (**Soltani et al., 2016**; **Venkatakrishnan and Wohlberg, 2019**). This technique, combined with deep learning, might have great potential for time-dependent situations in geochemistry and porous media flow.

Holtzman et al. (2018) used over 50,000 micro-earthquake events and employed a dictionary method for detecting time-dependent cyclic behavior for over 4 years in geothermal fields from time-dependent behavior of seismic properties. In detail, they combined three sequential unsupervised steps: non-negative matrix factorization, HMM and then a clustering method.

Bianco et al. (2019) obtained a high-resolution tomography under Long Beach, CA, from seismic noise recorded on 5204 geophones. Dense sampling was applied in order to learn a dictionary from locally sparse travel time tomography using Autoencoders, a method based on fully connected NN. Their work is based on a previous, theoretically developed approach presented in **Bianco and Gerstoft (2018)**.

How HPC Has to Evolve to Accommodate the Big Data Revolution

How to deal with a variety of data

In the earliest days of geophysical computer simulations, it was only possible to construct simplified box geometries and model boundary conditions, and the simulations themselves were intended only to elucidate mechanisms, rather than to predict natural behavior. With a great increase in both data volume and processing power, combined with the introduction of many novel data streams, it has become possible to model in detail precise locations on Earth using realistic initial and boundary conditions. Overall, augmenting computing resources has historically meant a possibility to increase model resolution, capturing new physics, assimilate data into models, and be able to increase the duration of the run and ensemble size of the simulations.

While simulations of deep mantle convection still focus largely on pattern formation (**Tackley, 2000**; **Piromallo et al., 2006**; **Puckett et al., 2018**) and large chemical and phase transitions (**Kimura et al., 2017**; **Shahnas et al., 2018**), regional mantle convection coupled to lithospheric deformation can take advantage of many more observations. As an example, consider the simulation of the Alaskan subduction zone (**Jadamec and Billen, 2010**) that uses earthquake hypocenters, sea floor age, surface heat flows, and seismic tomography to assemble the initial model geometry and temperature distribution, further uses GPS surface measurements and seismic anisotropy for model validation, and could possibly make use of gravitational measurements for model density variations.

The use of so many varied data streams of differing coverage and accuracy, greatly impacts the process of data assimilation. Since some data may only impact a small subset of the model, using a uniform norm over the model domain as an optimization objective is inappropriate and weighted norms must be explored. Furthermore, since uncertainties in the different data streams vary greatly, calculating a single descent direction from the Jacobian is often inappropriate, and schemes which separately update portions of the model or of the objective should be investigated. Solving these optimization problems, which are themselves both multiscale and multiphysics, require the same solver advances as the forward problem, but with different considerations for reduced accuracy solvers and preconditioning. **Haber and Oldenburg (1997)** developed a methodology to invert two different data sets with the assumption that the underlying models have a common structure.

Performances of software at a large number of processors heavily depend on the type of computational problem (**Stoller et al., 2019**). At one extreme, a problem like Monte Carlo, which is almost embarrassingly parallel, can be virtually scaled up to any number of processors. On the contrary, problems requiring intense node to node communication are limited by memory throughput.

In Earth Sciences, HPC has been applied to coupling mantle convection with plate tectonics (**Stadler et al., 2010**). In the past, a sudden progress in modeling Geodynamics meant a rapid scientific change in paradigm, as for the pioneering papers on the flushing event in mantle convection, e.g., (**Tackley, 1993**; **Honda et al., 1993**). An excellent introduction to HPC for scientists is the recent book of **Chopp (2019)**.

Types of supercomputers, servers, workstations available

Supercomputers at any era lie in the forefront and cost the most, in excess of $100 million, or the price of a top-notch warplane, like the B-2 Stealth bomber. Supercomputers can perform calculations 100–500 times faster than computers at smaller centers like at universities. This is the reason why researchers at universities must submit proposals to request excessive amounts of core-hours, like a few tens of million to hundred millions CPU hours at national centers. Servers cost in the range between $200,000 and $2,000,000 and serve a community or a small powerful department. It is like a powerful chemical instrument, a mass spectrometer or a magnetic-resonance instrument with strength of around 1 Tesla. Workstations cost varies between $10,000 and $35,000, in the past 30 years, and are useful for an individual power user or a group of enthusiastic students in one group or a small department at a junior college. One needs to emphasize that a user has only partial access to large computing systems, and sometimes has to share its access with thousands of other users.

Today one can classify supercomputers into those which are dedicated to numerical simulations and those which are focused on data, called data-centric. Servers can be classified in the same way as are workstations. The reader is urged to look at information about data-centric supercomputing focus under Steve Quenette of E-Research Center of Monash University in Melbourne, Australia (**Abramson et al., 2010**). Cloud computing, where one uses computing resources over the internet, is an inexpensive way of trying out new computational techniques, when one is short for time and enough funds to purchase the equipment outright for examining an interesting new problem. Cloud computing provides flexibility for small groups or start-up companies to engage in big data and large-scale simulations for a few case studies. One has to remind the reader, however, that cloud computing is presently still limited versus HPC supercomputers because the connection between nodes in the cloud is not designed to sustain high-demand, which is instead essential in parallel computational geophysics.

Exascale computing is a national imperative for many countries, especially China and the United States and is being pursued aggressively by both nations, with other countries like Japan, South Korea, India, Russia, Israel and the European Union behind. HPC touches on a country's critical challenges in national security, energy assurance, economic competitiveness, health care and scientific progress. It is therefore important for geodynamicists to recognize this opportunity and jump on this wagon. The reader can read more about America's relentless drive toward exascale computing by consulting the website exascaleproject.org, where the latest information is released.

New Chinese supercomputers will be located in both Qingdao and Changsha. A recent announcement has revealed the prowess of a single Tianhe-3 rack which can reach 4.7 Petaflops. Therefore 300–500 of them will enable Tianhe-3 to reach around two exaflops. They can rival the American exaflop machines produced by CRAY at Oak Ridge National Laboratory and Argonne National Laboratory. The one at Qingdao, rated currently at 1250 Petaflops, will be devoted to geosciences in modeling ocean-driven waves, tsunami waves and geodynamics. With the current silicon-based chip technology American and Chinese exascale supercomputers will all be in the range around 2500 Petaflops for the next 3 years. The machine at Qingdao will be a big boon to Chinese geosciences, as was the Earth Simulator at 40 Teraflops, when it was introduced in Japanese earth sciences back in 2004. At the Argonne National Laboratory, Intel will develop Aurora, an Exascale system built upon the Intel Xeon Scalable platform and Xe architecture-based GPUs. The new computer will require 200 racks, will support over 10 petabytes of memory and over 230 petabytes of storage.

A remarkable recent example of application of HPC to seismic simulations was obtained using 27,648 cores on Amazon Web Services achieving 1.09 Petaflops of performance (**Breuer et al., 2019**). In machine learning, a remarkable recent advancement has been a big deep learning system implemented by Google engineers called GPipe, in which a novel batch-splitting pipelining algorithm is developed, resulting in almost linear speedup when a model is partitioned across multiple accelerators (**Huang et al., 2018**), and applied to both image recognition and language translation, reaching unprecedented performance.

Among the most recent innovations, the introduction of three-dimensional memory by both Intel and AMD Inc. (a Taiwan based company) in 2017 has made GPU obsolete, by increasing the bandwidth rate tremendously, over two orders of magnitude. This development has caused a devaluation of Nvidia stock prices. However, more research is needed to convince the many agnostics that GPUs no longer held sway in HPC and data science, as 3D memory is going to upend this paradigm.

We note that quantum computing (e.g., **Bernhardt, 2019**), which is based on a fusion of quantum mechanics and computer science, can hold surprising promises especially in networking and nano-material properties in the near future. Currently, many efforts are being put into its fundamental research by many countries and companies, such as IBM and Google. Finally, it has been recently announced that Google has achieved "quantum supremacy," i.e., it has built a quantum computer that performs far above the greatest and fastest supercomputer in the world (**Arute et al., 2019**). This news is being scrutinized now by scientists in the world, but if confirmed would clearly change computational physics, geophysics, and computer sciences in general, since quantum computers need to be programmed in a completely different way than traditional ones (**National Academies of Sciences, Engineering, and Medicine, 2019**).

Educational Aspects

In the past 30 years geoscientists were divided between the ones who programmed in standard languages like Fortran and C, and the ones that used commercial tools such as Mathematica and Matlab. Now Python and Julia are coming to the fore. For example, Julia has recently been shown to perform as well as compiled code for differentiable Programming as well as for multiple Machine Learning algorithms (**Innes et al., 2019**).

Innovating the Geosciences

Progresses in open source tools have put at disposition easy to use open source options that make commercial tools obsolete, when Linux made Unix obsolete. Examples are the Python ecosystem and possibly recently Julia, specifically for HPC.

Students and new professionals approaching this field should take note of this feature of differentiable programming because it is an important novel advantage of Julia not available on Python and Matlab for HPC. Two recent books on Julia (**Kaminski and Szufel, 2018**; **McNicholas and Tait, 2019**) are recommended to students as well as “Scientific Machine Learning: How Julia Employs Differentiable Programming” (**Bezanson et al., 2019**).

We want to emphasize here how stepping from the proprietary Matlab software to the open Python ecosystem, a flurry of novel tools available for scientists have appeared. Among them, Jupyter Notebooks allow a clear interface to communicate science, which allows the users to reproduce results immediately. Several attempts have been put forward in seismology, to propose efficient practical programming exercises, such as “seismo-live” (**Krischer et al., 2018**) and parallel Jupyter efforts (**Aiken et al., 2018**). Notebooks are a setting in which students are allowed to learn by playing with numerical tools, making mistakes and immediately looking at the consequences. Python language and the Jupyter Notebook are for example used to design Underworld thermomechanical models without any pre-existing knowledge of programming (**Beucher et al., 2019**). A geophysical community platform of tools for geodata is offered by Pangeo, a project that serves as a coordination point between scientists, software, and computing infrastructure (pangeo.io).

Books for computational geodynamics have been mostly written with examples written in Matlab **(Gerya, 2019)** and Python (**Morra, 2018**). Several attempts at producing modern, interactive, hands-on books have been recently put forward. One example is the effort by David I. Ketcheson, Randall J. LeVeque, and Mauricio J. del Razo to write a book that teaches Riemann solutions for hyperbolic systems of partial differential equations and approximate Riemann solvers in Jupyter notebooks; the textbook, titled “Riemann Problems and Jupyter Solutions, Theory and Approximate Solvers for Hyperbolic PDEs,” is still in preparation but mostly available in Notebook format (see software table for the link). Another remarkable attempt is by the mathematician Ed Bueler from University of Alaska Fairbanks who is preparing a book entitled “PETSc for Partial Differential Equations: Numerical Solutions in C using the Portable, Extensible Toolkit for Scientific computing” and whose numerous Python implementations (using PETSc4py) are already available (see **Table 1** for link).

National Laboratories and other major institutions are also developing visualization interfaces or specific software for Jupyter Notebooks. HPC center security requirements pose challenges to develop software infrastructure to be directly used in Jupyter simulations. For example, **Ibrahim et al. (2019)** describe a system that enables executing the visualization software Ascent within Jupyter to visualize large scale HPC results.

Recruiting Teachers for the Transition

Because of the rapid growth of the job market in machine learning (ML) applied to geosciences, university education is not sufficient and too slow to adapt to market conditions. In parallel to restructuring university education with the addition of courses covering ML at the undergraduate level, the present workforce needs to be educated by focused ad-hoc training. However, such professionals are difficult to find because of economic market forces. In China a solution is to hire just freshly graduated PHDs in computer science. At the China University of Geosciences (CUG) in Wuhan, this strategy is adopted because of the increasing demand for training undergraduates in Big Data for immediate employment. Eight assistant professors were hired in the past 15 months. At CUG there is no plan to promote a graduate program in Data Science. In the West, a solution is to hire part-time professionals from retired industrial personnel from Cray Inc. or from the United States or foreign national laboratories, like Los Alamos, Oak Ridge, Livermore, Jet Propulsion national laboratories, French C.N.R.S. and the German Max Planck Institutes. Many of these old professionals have fundamental computer skills, for example going back to CRAY machines in the 1980s. A good example of the experience from a veteran programmer from CRAY Research in the 1990s can be found in this lucid book on parallel programming based on co-array Fortran, an alternative to MPI (**Numrich, 2018**), which has proved its worth against Julia on a weather code *WRF* written at the National Center for Atmospheric Research Center at Boulder, Colorado and on over 3000 CPU cores.

Software Teams

Although historically most of the successful and widely used codes in geophysics and geodynamics have been developed by a single educated geoscientist, as the software that simulates physical systems becomes an integral part of scientific and engineering practice,

Table 1 List of some of the open source software among the ones reviewed in the text.

Software	*Numerical technique*	*Application*	*Link*
Geodynamics and seismology			
Aspect	AMR Finite element	Thermal convection	https://github.com/geodynamics/aspect
CITCOM	Multigrid Finite element	Thermochemical convection	https://github.com/geodynamics/citcoms https://github.com/geodynamics/citcomcu
DG Library	Discontinuous Galerkin Finite elements	Wave propagation and geodynamics	http://maartendehoop.rice.edu/software/
GrowClust	Hierarchical clustering	Relocating earthquakes	https://github.com/dttrugman/GrowClust
LaMeM	Marker-in-cell. Finite differences	Lithosphere and Mantle evolution	https://bitbucket.org/bkaus/lamem
Milamin	Multigrid Finite element	Geomechanics	http://milamin.org/
Pangeo	Python. Jupyter	Ocean, atmosphere, land, climate science	https://pangeo.io/
PyLith	Finite-element	Dynamic and quasistatic crust and lithosphere modeling	https://github.com/geodynamics/pylith
Rayleigh	Pseudo-spectral	Magneto hydro-dynamics (MHD) convection	https://github.com/geodynamics/Rayleigh
SEPRAN	Finite-element	Mantle convection	http://ta.twi.tudelft.nl/Ftp/sepran/
SPECFEM	Spectral elements	Seismic wave propagation	https://github.com/geodynamics/specfem3d https://github.com/geodynamics/axisem https://github.com/geodynamics/specfem3d_globe
Sympeg	Finite volume. Inversion utilities	Applied geophysics	https://simpeg.xyz
Tectosaur	Boundary element method	Crustal stress	https://github.com/tbenthompson/tectosaur https://github.com/brendanjmeade/Bem2d.jl
Thetis	Finite element method	Ocean circulation	https://thetisproject.org/
Underworld	Particles in cell	Mantle and Crustal dynamics	https://github.com/underworldcode/underworld2
Machine learning			
Caffe	Python	Deep learning with expression, speed, and modularity	https://github.com/BVLC/caffe/
GAN Lab	TensorFlow JavaScript	Generative Adversarial Networks	https://poloclub.github.io/ganlab/
Keras	Python	High-level neural networks	
Microsoft Cognitive Toolkit	Directed graph	Deep neural networks	https://github.com/Microsoft/cntk
Polaris	Random search. Bayesian optimization	A library for hyperparameter optimization	https://github.com/rioyokotalab/polaris
PyMC	Python	Bayesian statistical modeling	https://github.com/pymc-devs
PyTorch	Python package	GPU accelerated Tensor computation. Deep NN	https://pytorch.org/
SciKit-learn	Python	Large machine learning library	https://scikit-learn.org
TensorFlow	CoLab in Jupyter Notebooks	Deep neural network library	https://www.tensorflow.org/
Theano	Fast multi-dimensional arrays Python Library	Define, optimize, and evaluate mathematical expressions	http://deeplearning.net/software/theano/
General computational libraries			
BEMLIB	Boundary elements	Fluid-dynamics	http://dehesa.freeshell.org/BEMLIB/
Clawpack	Finite volume	Linear and nonlinear hyperbolic systems of conservation laws	http://www.clawpack.org/
DeVito	Code generation and symbolic finite differences	Inversion methods	https://www.devitoproject.org/

Table 1 (Continued)

Software	*Numerical technique*	*Application*	*Link*
EVSL	Polynomial methods. Spectrum slicing	Eigenvalues extraction	https://github.com/eigs/EVSL
FEniCS	Finite elements. Parallel solvers	Generic solver of partial differential equations	https://fenicsproject.org/
HyTeg	Geometric Multigrid Finite element	Large scale high performance simulations	https://i10git.cs.fau.de/hyteg/hyteg
LB3D	Lattice Boltzmann method. Shan-Chen model	Binary fluid interactions	http://mtp.phys.tue.nl/lb3d/
Numerical Tours	Python, Julia, Matlab	Imaging, machine learning, vision and graphics	http://www.numerical-tours.com/
OpenLB	Lattice Boltzmann method	Transport problems. Fluid-dynamics	https://www.openlb.net/
Palabos	Lattice Boltzmann method. C ++	Fluid-dynamics	https://palabos.unige.ch/
PETSc	Parallel Library for Data Structures and Routines	Generic solver for scientific applications	https://www.mcs.anl.gov/petsc/
PyCurvelab	Fast Discrete Curvelet Transform	Digital image transformation	http://www.curvelet.org/
waLBerla	Lattice Boltzmann method	Multiphysics applications	https://www.walberla.net/
dolfin-adjoint	Automatic Differentiation. Python	Discrete adjoint and tangent linear models	http://www.dolfin-adjoint.org/en/latest/
Educational tools			
p4pdes	PETSc	Solve partial differential equations	https://github.com/bueler/p4pdes
Riemann problems	Clawpack and Jupyter	Hyperbolic systems of partial differential equations	https://github.com/clawpack/riemann_book
Seismo-live	Jupyter	Seismology	https://github.com/krischer/seismo_live

we must reconsider the relationships that produce and maintain these packages. Just as the low-temperature laboratory work of Kamerlingh Onnes in early part of the 20th century precipitated a move toward large academic groups under a single investigator to maintain and develop large experimental apparatus, support and development of software are forcing a change in the organization of scientific collaboration in the 2020s.

Developing open source scientific software requires many different competencies, from subject matter expertise, to version control, automated testing, bug tracking and maintenance, which is broader than what can be supported by the typical scientific research group. Moreover, there can be significant leverage in carrying out research using software already developed and maintained by another group. Thus groups developing scientific software are evolving in the direction of the broader open source software community, namely to larger, geographically separated groups from different scientific backgrounds.

For example, the PETSc numerical libraries, a widely used package for PDE simulations, now have active developers in 20 research groups across the spectrum of mathematics and science, and receive contributions from more than 100 developers per release (for comparison, the Linux operating system might receive 10,000 contributions). The necessity of forming such an expansive community and its benefits for both development and maintenance of scientific software infrastructure are now recognized as integral to computational science (**Turk, 2013**). This poses new challenges, such as the need of maintaining the right flexibility in addressing new questions for large teams, which are traditionally designed to follow the initial goal.

Conclusions and Future Perspectives

Traditional geodynamic modeling, Machine Learning and Deep Learning will all merge in one greater set of techniques (**Fig. 1**). Seismology and other branches in geosciences under the AGU and AOGS umbrellas, such as atmospheric sciences and ocean sciences, have been more prepared than geodynamicists to embrace the new available data science techniques. However, other branches in earth sciences, such as geochemistry, petrology and structural geology, are even further behind. Data will become more and more part

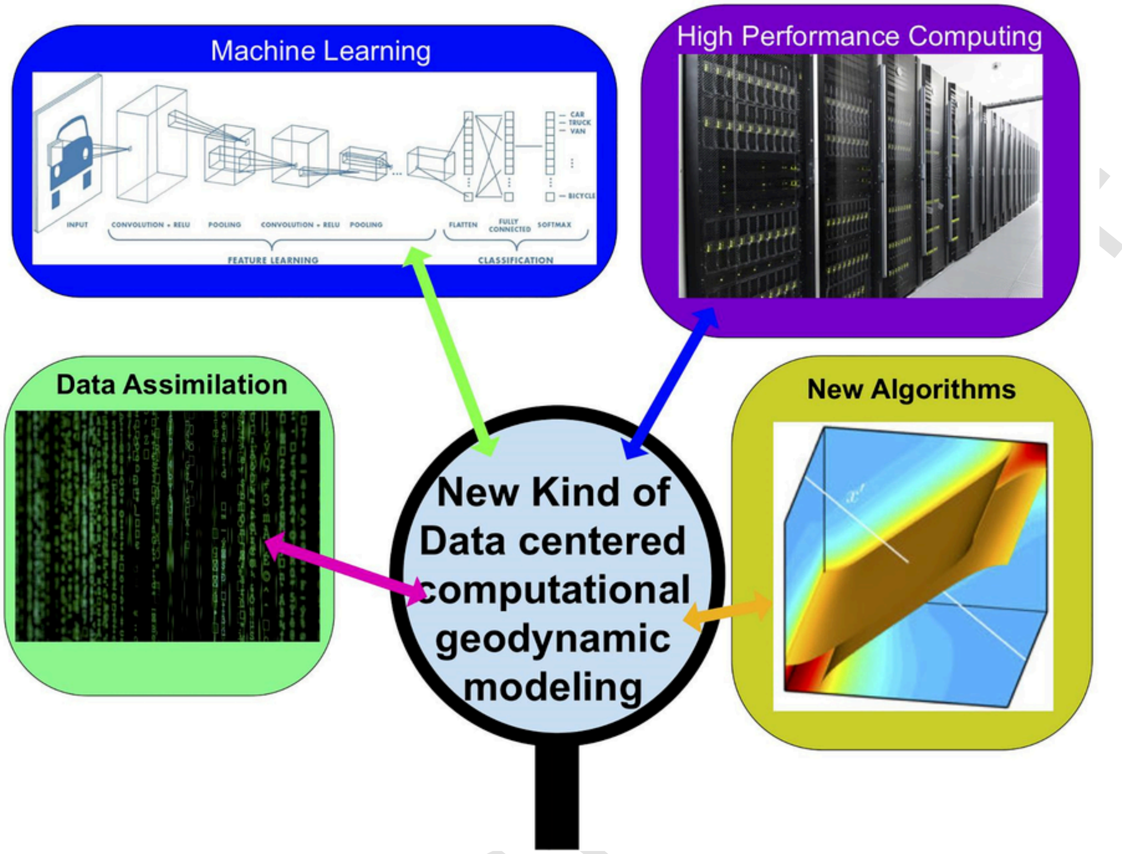

Fig. 1 Future geodynamic modeling will be based on integration of high performance computing, data assimilation, machine learning and new computational algorithms. The figure on the top right is from the Tryton supercomputer of Gdansk University of Technology. The image on the right is from Kiss et al. (2019).

of geodynamic modeling. To model large phase diagrams with numerous models to explore the entire parameter range will be less common. Instead, ML and DL will allow one to explore more parameters by interrogating fewer numerical tests.

Many techniques for modeling non-linear systems in geodynamics, such as development of shear-zones from complex feed-backs, and propagation of porosity waves in the deep crust, will become more common but they will require access to supercomputers. The same has already been mentioned for three-dimensional porosity waves, which are computationally more challenging than 3D convection with variable viscosity.

Numerical tools will have to become simpler in order to allow students and researchers to be able to develop algorithms for these complex tools. For this reason, Python, NumPy, Julia, and other emerging languages will become more common. Communication of science will also be done using more auxiliary tools such as Jupyter Notebooks. Procedures will need to be explained in a detailed manner in papers. Some journals (e.g., Journal of Open Source Software, JOSS) are already embracing this new approach.

Education of students and of workforce both in academia and in the private sector will play a pivotal role in the future of computational geosciences. Many courses will have to be added to undergraduate curricula and at Master's level, where more tuition can be charged to fuel a vigorous postdoc program at that university for data science research. Private companies will need educators for retooling some of the workers to the new ML and DL powered tools that will soon be ubiquitous in every sector of work.

Overall, we predict that data assimilation, machine learning, high performance computing, new data-driven algorithms and a new wave of educational tools using interactive communication software like Zoom and specially written portals for 5G phones will open the door for young researchers to a novel type of data centric geodynamic modeling.

Acknowledgments

We thank many people for making comments which helped to improve this article. Among them are Jiajun Chong, Witek Dzwinel, Mike Gurnis, Felix J. Herrmann, Ben Holtzman, Maarten V. De Hoop, Yuri Podladchikhov, Michele Rabinowicz, Ludovic Raess, Hosein Shahnas, Wim Spakman and Bei (Bobby) Zhang. DAY was supported by both DOE and NSF grants. HT was supported by the NSF grant "Stampede 2: Operations and Maintenance for the Next Generation of Petascale Computing," 1663578.

Further Reading

Abramson, D., Dinh, M.N., Kurniawan, D., Moench, B., DeRose, L., 2010. Data centric highly parallel debugging. In: Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, ACM. pp. 119–129.

Aiken, J.M., Aiken, C., Cotton, F., 2018. A Python library for teaching computation to seismology students. Seismological Research Letters 89 (3), 1165–1171.

Andrieu, C., de Freitas, N., Doucet, A., Jordan, M.I., 2003. An introduction to MCMC for machine learning. Machine Learning https://doi.org/10.1023/a:1020281327116.

Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J.C., Barends, R., Biswas, R., et al., 2019. Quantum supremacy using a programmable superconducting processor. Nature 574 (7779), 505–510.

Baumeister, P., Padovan, S., Tosi, N., Montavon, G., Nettelmann, N., MacKenzie, J., Godolt, M., 2019. Machine learning inference of the interior structure of low-mass exoplanets. arXiv preprint arXiv1911.12745.

Beaucé, E., Frank, W.B., Romanenko, A., 2018. Fast matched filter (FMF): An efficient seismic matched-filter search for both CPU and GPU architectures. Seismological Research Letters 89 (1), 165–172.

Bélanger, E., Vincent, A., 2005. Data assimilation (4D-VAR) to forecast flood in shallow-waters with sediment erosion. Journal of Hydrology https://doi.org/10.1016/j.jhydrol.2004.06.009.

Bélanger, E., Vincent, A., Charbonneau, P., 2007. Predicting solar flares by data assimilation in avalanche models. Solar Physics https://doi.org/10.1007/s11207-007-9009-3.

Bercovici, D., Schubert, G., Zebib, A., 1988. Geoid and topography for infinite Prandtl number convection in a spherical shell. Journal of Geophysical Research—Solid Earth 93 (B6), 6430–6436.

Bergen, K.J., Beroza, G.C., 2019. Earthquake fingerprints: Extracting waveform features for similarity-based earthquake detection. Pure and Applied Geophysics 176 (3), 1037–1059.

Bergen, K.J., Johnson, P.A., de Hoop, M.V., Beroza, G.C., 2019. Machine learning for data-driven discovery in solid earth geoscience. Science 363 (6433)https://doi.org/10.1126/science.aau0323.

Bernhardt, C., 2019. Quantum Computing for Everyone. Mit Press.

Beucher, R., Moresi, L., Giordani, J., Mansour, J., Sandiford, D., Farrington, R., Mondy, L., et al., 2019. UWGeodynamics: A teaching and research tool for numerical geodynamic modelling. Journal of Open Source Software 4 (36), 1136.

Beyreuther, M., Carniel, R., Wassermann, J., 2008. Continuous hidden Markov models: Application to automatic earthquake detection and classification at Las Canãdas Caldera, Tenerife. Journal of Volcanology and Geothermal Research https://doi.org/10.1016/j.jvolgeores.2008.04.021.

Beyreuther, M., Hammer, C., Wassermann, J., Ohrnberger, M., Megies, T., 2012. Constructing a hidden Markov model based earthquake detector: Application to induced seismicity. Geophysical Journal International https://doi.org/10.1111/j.1365-246x.2012.05361.x.

Bezanson J, Edelman A, Karpinski S and Shah VB (2019) *Scientific Machine Learning: How Julia Employs Differentiable Programming to Do it Best*, Siam News, October 2019.

Bianco, M.J., Gerstoft, P., 2018. Travel time tomography with adaptive dictionaries. IEEE Transactions on Computational Imaging 4 (4), 499–511.

Bianco, M.J., Gerstoft, P., Olsen, K.B., Lin, F.-C., 2019. High-resolution seismic tomography of Long Beach, CA using machine learning. Scientific Reports 9 (1), 1–11.

Both, G.-J., Choudhury, S., Sens, P., Kusters, R., 2019. DeepMoD: Deep learning for Model Discovery in noisy data. arXiv preprint arXiv1904.09406.

Brand, M., Oliver, N., Pentland, A., 1997. Coupled hidden Markov models for complex action recognition. In cvpr 97, 994.

Breuer, A., Cui, Y., Heinecke, A., 2019. Petaflop seismic simulations in the public cloud. In: International Conference on High Performance Computing, Springer, Cham. pp. 167–185.

Bunge, H.-P., Hagelberg, C.R., Travis, B.J., 2003. Mantle circulation models with variational data assimilation: Inferring past mantle flow and structure from plate motion histories and seismic tomography. Geophysical Journal International https://doi.org/10.1046/j.1365-246x.2003.01823.x.

Burl, M.C., Asker, L., Smyth, P., Fayyad, U., Perona, P., Crumpler, L., Aubele, J., 1998. Learning to recognize volcanoes on Venus. Machine Learning https://doi.org/10.1023/a:1007400206189.

Cai, D., Chow, E., Erlandson, L., Saad, Y., Xi, Y., 2018. SMASH: Structured matrix approximation by separation and hierarchy. Numerical Linear Algebra with Applications https://doi.org/10.1002/nla.2204.

Cesca, S., 2020. Seiscloud, a tool for density-based seismicity clustering and visualization. Journal of Seismology 24, 443–457.

Chamberlain, C.J., Hopp, C.J., Boese, C.M., Warren-Smith, E., Chambers, D., Chu, S.X., Michailos, K., Townend, J., 2018. EQcorrscan: Repeating and near-repeating earthquake detection and analysis in python. Seismological Research Letters 89 (1), 173–181.

Chen, J., Hubbard, S., Rubin, Y., 2001. Estimating the hydraulic conductivity at the south oyster site from geophysical tomographic data using bayesian techniques based on the normal linear regression model. Water Resources Research https://doi.org/10.1029/2000wr900392.

Chen, Y., 2018. Automatic microseismic event picking via unsupervised machine learning. Geophysical Journal International 212 (1), 88–102.

Chopp, D.L., 2019. Introduction to High Performance Scientific Computing. SIAM Press.

Corbi, F., Sandri, L., Bedford, J., Funiciello, F., Brizzi, S., Rosenau, M., Lallemand, S., 2019. Machine learning can predict the timing and size of analog earthquakes. Geophysical Research Letters 46 (3), 1303–1311.

Cox, D.R., 1959. The regression analysis of binary sequences. Journal of the Royal Statistical Society: Series B: Methodological https://doi.org/10.1111/j.2517-6161.1959.tb00334.x.

Cowell, R., 1998. Introduction to inference for Bayesian networks. In: Learning in Graphical Models. Springer, Dordrecht, pp. 9–26.

Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., Bharath, A.A., 2018. Generative adversarial networks: An overview. IEEE Signal Processing Magazine https://doi.org/10.1109/msp.2017.2765202.

Dammeier, F., Moore, J.R., Hammer, C., Haslinger, F., Loew, S., 2016. Automatic detection of alpine rockslides in continuous seismic data using hidden Markov models. Journal of Geophysical Research—Earth Surface https://doi.org/10.1002/2015jf003647.

DeCoste, D., 1998. The significance of Kasparov versus DEEP BLUE and the future of computer chess. ICGA Journal https://doi.org/10.3233/icg-1998-21104.

Denli, H., Subrahmanya, N., 2014. Multi-scale graphical models for spatio-temporal processes. In: Advances in Neural Information Processing Systems. pp. 316–324.

van Dinther, Y., Gerya, T.V., Dalguer, L.A., Mai, P.M., Morra, G., Giardini, D., 2013. The seismic cycle at subduction thrusts: Insights from seismo-thermo-mechanical models. Journal of Geophysical Research—Solid Earth 118 (12), 6183–6202.

Dutta, R., Cohn, A.G., Muggleton, J.M., 2013. 3D mapping of buried underworld infrastructure using dynamic Bayesian Network based multi-sensory image data fusion. Journal of Applied Geophysics https://doi.org/10.1016/j.jappgeo.2013.02.005.

Dzwinel, W., Yuen, D.A., Boryczko, K., Ben-Zion, Y., Yoshioka, S., Ito, T., 2005. Cluster analysis, data-mining, multi-dimensional visualization of earthquakes over space, time and feature space. Nonlinear Processes in Geophysics 12, 117–128.

Feng, J., He, X., Teng, Q., Ren, C., Chen, H., Yang, L., 2019. Reconstruction of porous media from extremely limited information using conditional generative adversarial networks. Physical Review E https://doi.org/10.1103/physreve.100.033308.

Fichtner, A., Bunge, H.-P., Igel, H., 2006. The adjoint method in seismology. Physics of the Earth and Planetary Interiors https://doi.org/10.1016/j.pepi.2006.03.018.

Flyer, N., Wright, G.B., 2009. A radial basis function method for the shallow water equations on a sphere. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 465 (2106), 1949–1976.

Fornberg, B., Flyer, N., 2015. A Primer on Radial Basis Functions With Applications to the Geosciences. SIAM Publisher, ISBN: 978-1-611974-02-7.

Fournier, A., Hulot, G., Jault, D., Kuang, W., Tangborn, A., et al., 2010. An introduction to data assimilation and predictability in geomagnetism. Space Science Reviews 155, 247–291.

Fukuda, S., Katagiri, R., Hirosawa, H., 2002. Unsupervised approach for polarimetric SAR image classification using support vector machines. In: IEEE International Geoscience and Remote Sensing Symposium, IEEE. vol. 5, pp. 2599–2601.

Fukushima, K., 1979. Neural network model for a mechanism of pattern recognition unaffected by shift in position-Neocognitron. IEICE Technical Report, A 62 (10), 658–665.

Gan, M., Peng, H., Peng, X., Chen, X., Inoussa, G., 2010. A locally linear RBF network-based state-dependent AR model for nonlinear time series modeling. Information Sciences 180 (22), 4370–4383.

Gerya, T., 2019. Introduction to Numerical Geodynamic Modeling. Cambridge University Press.

Glatzmaier, G.A., Schubert, G., Bercovici, D., 1990. Chaotic, subduction-like downflows in a spherical model of convection in the Earth's mantle. Nature 347 (6290), 274.

Glatzmaier, G.A., 2013. Introduction to Modeling Convection in Planets and Stars: Magnetic Field, Density Stratification, Rotation. vol. 24, Princeton University Press.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

Haber, E., Oldenburg, D., 1997. Joint inversion: A structural approach. Inverse Problems 13 (1), 63.

Hammer, C., Ohrnberger, M., Fäh, D., 2013. Classifying seismic waveforms from scratch: A case study in the alpine environment. Geophysical Journal International https://doi.org/10.1093/gji/ggs036.

Hastings, W.K., 1970. Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57, 97–109.

Hoffman, M.D., Blei, D.M., Wang, C., Paisley, J., 2013. Stochastic variational inference. The Journal of Machine Learning Research 14 (1), 1303–1347.

Holtzman, B.K., Pate, A., Paisley, J., Waldhauser, F., Repetto, D., 2018. Machine learning reveals cyclic changes in seismic source spectra in Geysers geothermal field. Science Advances 4 (5), eaao2929https://doi.org/10.1126/sciadv.aao2929.

Honda, S., Yuen, D.A., Balachandar, S., Reuteler, D., 1993. Three-dimensional instabilities of mantle convection with multiple phase transitions. Science 259 (5099), 1308–1311.

Huang, Y., Cheng, Y., Chen, D., Lee, H.J., Ngiam, J., Le, Q.V., Chen, Z., 2018. Gpipe: Efficient training of giant neural networks using pipeline parallelism. arXiv preprint arXiv1811.06965.

Ibrahim, S., Stitt, T., Larsen, M., Harrison, C., 2019. Interactive In Situ Visualization and Analysis Using Ascent and Jupyter No. LLNL-CONF-792241. Lawrence Livermore National Lab (LLNL), Livermore, CA.

Innes, M., Edelman, A., Fischer, K., Rackauckus, C., Saba, E., Shah, V.B., Tebbutt, W., 2019. Zygote: A differentiable programming system to bridge machine learning and scientific computing. arXiv preprint arXiv1907.07587.

Iturrarán-Viveros, U., 2012. Smooth regression to estimate effective porosity using seismic attributes. Journal of Applied Geophysics https://doi.org/10.1016/j.jappgeo.2011.10.012.

Ivakhnenko, A.G., Lapa, V.G., 1965. Cybernetic Predictive Devices. CCM Information Corp, New York.

Jerrum, M., Sinclair, A., 1996. The Markov Chain Monte Carlo Method: An Approach to Approximate Counting and Integration. Approximation Algorithms for NP-hard problems, PWS Publishing.

Ji, E.-Y., Moon, Y.-J., Park, J., Lee, J.-Y., Lee, D.-H., 2013. Comparison of neural network and support vector machine methods for Kp forecasting. Journal of Geophysical Research: Space Physics 118 (8), 5109–5117.

Jadamec, M.A., Billen, M.I., 2010. Reconciling surface plate motions with rapid three-dimensional mantle flow around a slab edge. Nature 465 (7296), 338–341.

Jones, R.H., Stewart, R.C., 1997. A method for determining significant structures in a cloud of earthquakes. Journal of Geophysical Research—Solid Earth 102 (B4), 8245–8254.

Kaminski, B., Szufel, P., 2018. Julia 1.0 Programming Cookbook: Over 100 Numerical and Distributed Computing Recipes for Your Daily Data Science Workflow. Packt Publishing.

Karasözen, E., Nissen, E., Bergman, E.A., Johnson, K.L., Walters, R.J., 2016. Normal faulting in the Simav graben of western Turkey reassessed with calibrated earthquake relocations. Journal of Geophysical Research—Solid Earth 121 (6), 4553–4574.

Karumuri, S., Tripathy, R., Bilionis, I., Panchal, J., 2019. Simulator-free solution of high-dimensional stochastic elliptic partial differential equations using deep neural networks. arXiv preprint arXiv1902.05200.

Kasparov, G., 2018. Chess, a of reasoning. Science 362 (6419), 1087.

Kimura, J.-i., Gill, J.B., van Keken, P.E., Kawabata, H., Skora, S., 2017. Origin of geochemical mantle components: Role of spreading ridges and thermal evolution of mantle. Geochemistry, Geophysics, Geosystems https://doi.org/10.1002/2016gc006696.

Komatitsch, D., Tsuboi, S., Tromp, J., 2005. The spectral-element method in seismology. Seismic Earth: Array Analysis of Broadband Seismograms https://doi.org/10.1029/157gm13.

Krischer, L., Aiman, Y.A., Bartholomaus, T., Donner, S., van Driel, M., Duru, K., Garina, K., et al., 2018. Seismo-live: An educational online library of Jupyter notebooks for seismology. Seismological Research Letters 89 (6), 2413–2419.

Leśniak, A., Isakow, Z., 2009. Space–time clustering of seismic events and hazard assessment in the Zabrze-Bielszowice coal mine, Poland. International Journal of Rock Mechanics and Mining Sciences 46 (5), 918–928.

Levy, D.N.L., 1990. How will chess programs beat Kasparov?. Computers, Chess, and Cognition https://doi.org/10.1007/978-1-4613-9080-0_4.

Li, D., Gurnis, M., Stadler, G., 2017. Towards adjoint-based inversion of time-dependent mantle convection with nonlinear viscosity. Geophysical Journal International 209 (1), 86–105.

Linnainmaa S (1970) *The Representation of the Cumulative Rounding Error of An Algorithm As a Taylor Expansion of the Local Rounding Errors*, Master's Thesis (in Finnish), University of Helsinki, pp. 6–7.

Liu, L., Gurnis, M., 2008. Simultaneous inversion of mantle properties and initial conditions using an Adjoint of mantle convection. Journal of Geophysical Research—Solid Earth https://doi.org/10.1029/2008jb005594.

Liu, L., Gurnis, M., 2010. Dynamic subsidence and uplift of the Colorado plateau. Geology https://doi.org/10.1130/g30624.1.

Liu, L., Spasojevic, S., Gurnis, M., 2008. Reconstructing farallon plate subduction beneath North America back to the late cretaceous. Science 322 (5903), 934–938.
Liu, S., Zhong, Z., Takbiri-Borujeni, A., Kazemi, M., Qinwen, F., Yang, Y., 2019. A case study on homogeneous and heterogeneous reservoir porous media reconstruction by using generative adversarial networks. Energy Procedia https://doi.org/10.1016/j.egypro.2019.01.493.
Louboutin, M., Lange, M., Luporini, F., Kukreja, N., Witte, P.A., Herrmann, F.J., Velesko, P., Gorman, G.J., 2019. Devito (v3.1.0): An embedded domain-specific language for finite differences and geophysical exploration. Geoscientific Model Development https://doi.org/10.5194/gmd-12-1165-2019.
Luo, Y., Zhu, H., Nissen-Meyer, T., Morency, C., Tromp, J., 2009. Seismic modeling and imaging based upon spectral-element and adjoint methods. The Leading Edge https://doi.org/10.1190/1.3124932.
Majumder, C., Hier, A., Hier Majumder, C.A., Bélanger, E., DeRosier, S., Yuen, D.A., Vincent, A.P., 2005. Data assimilation for plume models. Nonlinear Processes in Geophysics https://doi.org/10.5194/npg-12-257-2005.
Malevsky, A.V., Yuen, D.A., 1992. Strongly chaotic non-Newtonian mantle convection. Geophysical and Astrophysical Fluid Dynamics 65 (1–4), 149–171.
Malinverno, A., 2002. Parsimonious Bayesian Markov chain Monte Carlo inversion in a nonlinear geophysical problem. Geophysical Journal International https://doi.org/10.1046/j.1365-246x.2002.01847.x.
Martin, B., Fornberg, B., Flyer, N., 2017. A primer on radial basis functions with applications to the geosciences. Jahresbericht der Deutschen Mathematiker-Vereinigung 119 (1), 53–58.
Masotti, M., Falsaperla, S., Langer, H., Spampinato, S., Campanini, R., 2006. Application of support vector machine to the classification of volcanic tremor at Etna, Italy. Geophysical Research Letters 33 (20).
Matsui, H., Heien, E., Aubert, J., Aurnou, J.M., Avery, M., Brown, B., Buffett, B.A., et al., 2016. Performance benchmarks for a next generation numerical dynamo model. Geochemistry, Geophysics, Geosystems 17 (5), 1586–1607.
McCulloch, W.S., Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics 5 (4), 115–133.
McNicholas, P.D., Tait, P., 2019. Data Science With Julia. CRC Press.
Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E., 1953. Equation of state calculations by fast computing machines. The Journal of Chemical Physics 21 (6), 1087–1092.
Minsky M and Papert S (1969) *An Introduction to Computational Geometry*. Cambridge TIASS HIT.
Mittal, A., Kassim, A., 2007. Bayesian Network Technologies: Applications and Graphical Models: Applications and Graphical Models. IGI Global.
Morra, G., 2018. Pythonic geodynamics. Lecture Notes in Earth System Sciences https://doi.org/10.1007/978-3-319-55682-6.
Mosser, L., Dubrule, O., Blunt, M.J., 2017. Reconstruction of three-dimensional porous media using generative adversarial neural networks. Physical Review E96 (4–1), , 043309.
Mosser, L., Dubrule, O., Blunt, M.J., 2018. Stochastic reconstruction of an oolitic limestone by generative adversarial networks. Transport in Porous Media https://doi.org/10.1007/s11242-018-1039-9.
National Academies of Sciences, Engineering, and Medicine, 2019. Quantum Computing: Progress and Prospects. National Academies Press.
Numrich, R.W., 2018. Parallel Programming With Co-Arrays: Parallel Programming in Fortran. Chapman and Hall/CRC.
Oware, E.K., Irving, J., Hermans, T., 2019. Basis-constrained bayesian Markov-chain Monte carlo difference inversion for geoelectrical monitoring of hydrogeologic processes. Geophysics https://doi.org/10.1190/geo2018-0643.1.
Parker DB (1982) *Learning Logic*, Invention report S81-64, File 1, Office of Technology Licensing, October, Stanford University.
Pawley, S., Schultz, R., Playter, T., Corlett, H., Shipman, T., Lyster, S., Hauck, T., 2018. The geological susceptibility of induced earthquakes in the Duvernay play. Geophysical Research Letters https://doi.org/10.1002/2017gl076100.
Peng, Z., Zhao, P., 2009. Migration of early aftershocks following the 2004 Parkfield earthquake. Nature Geoscience 2 (12), 877.
Piromallo, C., Becker, T.W., Funiciello, F., Faccenna, C., 2006. Three-dimensional instantaneous mantle flow induced by subduction. Geophysical Research Letters https://doi.org/10.1029/2005gl025390.
Poupinet, G., Ellsworth, W.L., Frechet, J., 1984. Monitoring velocity variations in the crust using earthquake doublets: An application to the Calaveras Fault, California. Journal of Geophysical Research—Solid Earth 89 (B7), 5719–5731.
Puckett, E.G., Turcotte, D.L., He, Y., Lokavarapu, H., Robey, J.M., Kellogg, L.H., 2018. New numerical approaches for modeling thermochemical convection in a compositionally stratified fluid. Physics of the Earth and Planetary Interiors https://doi.org/10.1016/j.pepi.2017.10.004.
Puntanen, S., 2010. Linear regression analysis: Theory and computing by Xin Yan, Xiao Gang Su. International Statistical Review https://doi.org/10.1111/j.1751-5823.2010.00109_11.x.
Rabiner, L.R., Juang, B.H., 1992. Hidden Markov models for speech recognition—Strengths and limitations. Speech Recognition and Understanding https://doi.org/10.1007/978-3-642-76626-8_1.
Ratnaswamy, V., Stadler, G., Gurnis, M., 2015. Adjoint-based estimation of plate coupling in a non-linear mantle flow model: Theory and examples. Geophysical Journal International 202 (2), 768–786.
Reynen, A., Audet, P., 2017. Supervised machine learning on a network scale: Application to seismic event classification and detection. Geophysical Journal International https://doi.org/10.1093/gji/ggx238.
Richardson, J.P., Waite, G.P., 2013. Waveform inversion of shallow repetitive long period events at Villarrica Volcano, Chile. Journal of Geophysical Research—Solid Earth 118 (9), 4922–4936.
Rietbrock, A., Tiberi, C., Scherbaum, F., Lyon-Caen, H., 1996. Seismic slip on a low angle normal fault in the Gulf of Corinth: Evidence from high-resolution cluster analysis of microearthquakes. Geophysical Research Letters 23 (14), 1817–1820.
Riggelsen, C., Ohrnberger, M., 2014. A machine learning approach for improving the detection capabilities at 3C seismic stations. Pure and Applied Geophysics https://doi.org/10.1007/s00024-012-0592-3.
Rosenblatt, F., 1957. The Perceptron, A Perceiving and Recognizing Automaton Project Para. Cornell Aeronautical Laboratory.
Ross, Z.E., Hauksson, E., Ben-Zion, Y., 2017. Abundant off-fault seismicity and orthogonal structures in the San Jacinto fault zone. Science Advances 3 (3), e1601946.
Ross, Z.E., Meier, M.-A., Hauksson, E., Heaton, T.H., 2018. Generalized seismic phase detection with deep learning. Bulletin of the Seismological Society of America 108 (5A), 2894–2901.
Rouet-Leduc, B., Hulbert, C., Lubbers, N., Barros, K., Humphreys, C.J., Johnson, P.A., 2017. Machine learning predicts laboratory earthquakes. Geophysical Research Letters 44 (18), 9276–9282.
Rouet-Leduc, B., Hulbert, C., Johnson, P.A., 2018. Breaking Cascadia's silence: Machine learning reveals the constant chatter of the Megathrust. arXiv preprint arXiv1805.06689.
Ruano, A.E., Madureira, G., Barros, O., Khosravani, H.R., Ruano, M.G., Ferreira, P.M., 2014. Seismic detection using support vector machines. Neurocomputing https://doi.org/10.1016/j.neucom.2013.12.020.
Rudy, S.H., Brunton, S.L., Proctor, J.L., Kutz, J.N., 2017. Data-driven discovery of partial differential equations. Science Advances 3 (4), e1602614.
Sadrfaridpour, E., Jeereddy, S., Kennedy, K., Luckow, A., Razzaghi, T., Safro, I., 2016. Algebraic Multigrid support vector machines. arXiv preprint arXiv1611.05487.

Schäfer, A.M., Wenzel, F., 2019. Global Megathrust earthquake hazard—maximum magnitude assessment using multi-variate machine learning. Frontiers in Earth Science https://doi.org/10.3389/feart.2019.00136.

Schmidhuber, J., 2015. Deep learning in neural networks: An overview. Neural Networks 61, 85–117.

Selten, F.M., Schevenhoven, F.J., Duane, G.S., 2017. Simulating climate with a synchronization-based supermodel. Chaos 27 (12), 126903.

Shahnas, M.H., Yuen, D.A., Pysklywec, R.N., 2018. Inverse problems in geodynamics using machine learning algorithms. Journal of Geophysical Research—Solid Earth https://doi.org/10.1002/2017jb014846.

Shearer, P.M., 1997. Improving local earthquake locations using the L1 norm and waveform cross correlation: Application to the Whittier Narrows, California, aftershock sequence. Journal of Geophysical Research—Solid Earth 102 (B4), 8269–8283.

Sheng, V.S., Provost, F., Ipeirotis, P.G., 2008. Get another label? Improving data quality and data mining using multiple, noisy labelers. In: Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD 08. https://doi.org/10.1145/1401890.1401965.

Soltani, S., Kilmer, M.E., Hansen, P.C., 2016. A tensor-based dictionary learning approach to tomographic image reconstruction. BIT Numerical Mathematics 56 (4), 1425–1454.

Srinivasan, G., Hyman, J.D., Osthus, D.A., Moore, B.A., O'Malley, D., Karra, S., Rougier, E., Hagberg, A.A., Hunter, A., Viswanathan, H.S., 2018. Quantifying topological uncertainty in fractured systems using graph theory and machine learning. Scientific Reports 8 (1), 1–11.

Stadler, G., Gurnis, M., Burstedde, C., Wilcox, L.C., Alisic, L., Ghattas, O., 2010. The dynamics of plate tectonics and mantle flow: From local to global scales. Science 329 (5995), 1033–1038.

Stein, S., Wysession, M., 2003. An Introduction to Seismology, Earthquakes, and Earth Structure. John Wiley & Sons.

Steve, I., Ding, Y., Wei, C., 2002. Dynamic bus arrival time forecasting with artificial neural networks. Journal of Transportation Engineering 10, 429–438.

Stoller, S.D., Carbin, M., Adve, S., Agrawal, K., Blelloch, G., Stanzione, D., Yelick, K., Zaharia, M., 2019. Future directions for parallel and distributed computing: SPX 2019 workshop report. In: NSF Workshop Reports.

Tackley, P.J., 1993. Effects of strongly temperature-dependent viscosity on time-dependent, three-dimensional models of mantle convection. Geophysical Research Letters 20 (20), 2187–2190.

Tackley, P.J., 2000. Self-consistent generation of tectonic plates in time-dependent, three-dimensional mantle convection simulations. Geochemistry, Geophysics, Geosystems https://doi.org/10.1029/2000gc000036.

Tenorio, L., 2017. An Introduction to Data Analysis and Uncertainty Quantification for Inverse Problems. SIAMhttps://doi.org/10.1137/1.9781611974928.

Trugman, D.T., Shearer, P.M., 2017. GrowClust: A hierarchical clustering algorithm for relative earthquake relocation, with application to the Spanish Springs and Sheldon, Nevada, earthquake sequences. Seismological Research Letters 88 (2A), 379–391.

Turk, M.J., 2013. Scaling a code in the human dimension. In: Proceedings of the Conference on Extreme Science and Engineering Discovery Environment Gateway to Discovery—XSEDE '13. https://doi.org/10.1145/2484762.2484782.

Venkatakrishnan, S.V., Wohlberg, B., 2019. Convolutional dictionary regularizers for tomographic inversion. In: ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). https://doi.org/10.1109/icassp.2019.8682637.

Wainwright, M.J., Jordan, M.I., 2007. Graphical models, exponential families, and variational inference. Machine Learning https://doi.org/10.1561/2200000001.

Wainwright, H.M., Chen, J., Sassen, D.S., Hubbard, S.S., 2014. Bayesian hierarchical approach and geophysical data sets for estimation of reactive facies over plume scales. Water Resources Research 50 (6), 4564–4584.

Wang, B., Zhang, N., Lu, W., Wang, J., 2018. Deep-learning-based seismic data interpolation: A preliminary result. Geophysics 84 (1), V11–V20.

Werbos, P.J., 1982. Applications of advances in nonlinear sensitivity analysis. In: System Modeling and Optimization. Springer, Berlin, Heidelberg, pp. 762–770.

Witte, P.A., Louboutin, M., Kukreja, N., Luporini, F., Lange, M., Gorman, G.J., Herrmann, F.J., 2019. A large-scale framework for symbolic implementations of seismic inversion algorithms in Julia. Geophysicshttps://doi.org/10.1190/geo2018-0174.1.

Witte, P.A., Louboutin, M., Luporini, F., Gorman, G.J., Herrmann, F.J., 2019. Compressive least-squares migration with on-the-fly Fourier transforms. Geophysicshttps://doi.org/10.1190/geo2018-0490.1.

Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y., 2009. Robust face recognition via sparse representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2), 210–227.

Wright, G.B., Flyer, N., Yuen, D.A., 2010. A hybrid radial basis function–pseudospectral method for thermal convection in a 3-D spherical shell. 11, Q07003https://doi.org/10.1029/2009GC002985.

Yang, J., Leskovec, J., 2012. Defining and evaluating network communities based on ground-truth. In: 2012 IEEE 12th International Conference on Data Mining. https://doi.org/10.1109/icdm.2012.138.

Yoon, C.E., O'Reilly, O., Bergen, K.J., Beroza, G.C., 2015. Earthquake detection through computationally efficient similarity search. Science Advances 1 (11), e1501057.

Yuen, D.A., Dzwinel, W., Ben-Zion, Y., Kadlec, B., 2009. Visualization of earthquake clusters over multidimensional space. In: Encyclopedia of Complexity and Systems Science. Springer New York, pp. 2347–2371.

Zhu, W., Beroza, G.C., 2018. PhaseNet: A deep-neural-network-based seismic arrival-time picking method. Geophysical Journal International 216 (1), 261–273.

Zhu, L., Peng, Z., McClellan, J., Li, C., Yao, D.D., Li, Z., Fang, L., 2019. Deep learning for seismic phase detection and picking in the aftershock zone of 2008 Mw7. 9 Wenchuan Earthquake. Physics of the Earth and Planetary Interiors 293, 106261.

Biography

Gabriele Morra currently holds a joint position as Associate Professor at the department of Physics and at the School of Geosciences at the University of Louisiana at Lafayette. Since 2015, he has also been Hensarling/Chapman Endowed Professor in Geology. He obtained a PhD in geophysics from the ETH Zürich in 2005, with a dissertation on new numerical methods for subduction, which he applied to oceanic arcs. After the PhD he took up a 2-year postdoctoral fellowship at the Institute of Computational Sciences at the ETH acquiring new skills in fast algorithms, such as the fast multiple method (FMM) and published the first a parallel FMM implementation of the boundary element method applied to geodynamics. He then moved to Rome where he investigated megathrusts in subduction zones. In 2009, he won a Swiss National Foundation Advanced Researcher Fellowship to move to Australia where he worked at the School of Geosciences of the University of Sydney. There he worked on processing paleo-tectonics reconstructions and proposed a new mechanism behind global plate reorganizations. In 2011 he became Research Professor at Seoul National University for 2 years, where in 2012 he co-organized the first international geodynamics conference in S. Korea, in Jeju Island. Since 2013, he has been a faculty at UL at Lafayette and has been working on new topics such as machine learning applied to volcanology and seismology, Lattice Boltzmann Method applied to mantle convection, underwater landslides, porous media flow. In 2017 he wrote a textbook titled “Pythonic Geodynamics.” He has written over 40 peer reviewed papers, presented his work over 100 times, has edited two books and he is co-editor in chief of “Artificial Intelligence in Geosciences.”

David A. Yuen is currently a senior research fellow at the Department of Applied Physics and Applied Mathematics at Columbia University in New York City and also a Professor at Department of Computer Science and Geophysics and Oceanography at Ocean University of China in Qingdao, China. He is also a technical officer at a start-up Ultimate Vision specializing in artificial intelligence in Beijing. Before Columbia University he spent 33 years at the Department of Earth Sciences and a fellow at the Minnesota Supercomputing Institute. He also taught at Arizona State University in the Department of Geology and 9 months at NCAR and Department of Geology and Geophysics at University of Colorado in Boulder. He earned his bachelor’s degree in chemistry at Caltech in 1969, a master’s degree in geophysics and oceanography at Scripps Institution of Oceanography in 1973 and a PhD at Department of Earth and Space Sciences at UCLA in 1978. From 2011 he spent nearly a decade with the China University of Geosciences at Wuhan, working on geothermal and machine learning has helped the Department of Big Data to craft out a curriculum for a Bachelor’s degree in Big Data. He has worked in many fields, among them numerical modeling, visualization, numerical algorithms, bio-informatics, high-performance computing, molecular dynamics, mantle convection. Chemical physics, mineral physics, mantle rheology machine learning and artificial intelligence. He is a Fellow of the American Geophysical Union. He has been listed in the highly cited geoscientists three times, the first was in 2003.

Henry M. Tufo is a Full Professor of Computer Science at the University of Colorado at Boulder and founding Director of the Computational Science Center, whose mission is to explore the intersection of high-performance computing, emerging architectures, and high-productivity computing. He has made contributions to a broad array of areas including high-performance and high-productivity computing, scientific application development, parallel algorithms and architectures, high-order numerical methods, scalable solvers, cluster and cloud computing, and immersive visualization. His current research focuses on increasing IPC of future processor architectures through use of horizontal and vertical instruction fusion, the application of deep-learning and big-data techniques to weather/climate science, personalized medicine, patient cohort analysis, and genomics, and architecting computing systems for data-intensive and artificial intelligence applications.Since coming to Boulder, Tufo has managed over $25 million and spearheaded efforts to create a unified cyberinfrastructure for the universities and research laboratories in the Front Range. To this end, he has led several high-end computing efforts, including bringing one of the world's first IBM BlueGene systems to Boulder in early 2005, being project manager and principal architect of the Janus supercomputer (#31 on the June 2010 Top500 list) and its co-designed facility, and leading the effort to create a high-performance research network for the campus.Previously, Tufo served as Computer Science Section Head at the National Center for Atmospheric Research where he directed their computer science research, ran the technology tracking and experimental systems programs, and managed NSF TeraGrid (now XSEDE) operations. He was a member of the development team for HOMME, a high-order dynamical core development environment, team leader for its discontinuous Galerkin, non-hydrostatic, and multi-species transport development, and lead designer of a service oriented architecture for Grid-BGC, an end-to-end gird-based solution for terrestrial ecosystem modeling. Before coming to Boulder, he was a member of the DOE ASC Center for Astrophysical Thermonuclear Flashes at the University of Chicago and Argonne National Laboratory, where he co-developed NEK5000, a state-of-the-art code for simulating unsteady incompressible flows in complex geometries and architected FLASH 1.x, an adaptive mesh refinement reactive flow code.Tufo has an extensive track record of serving the computational science community. He is a member of the Science Advisory Board for the Emerging Analytics Center at the University of Arkansas at Little Rock, NSF XSEDE Resource Allocation Committee, and, previously, Microsoft Technical Computing Executive Advisory Council, the VCR's Interdisciplinary Computational Science and Engineering Initiative Steering Committee, and the Boulder Campus Cyberinfrastructure Board. He has held leadership positions with various conferences including IEEE Cluster, LCI, ICCS, ICS, TrustCom, IET International Conference on Frontier Computing, etc.Tufo received his M.S. and PhD degrees in Applied Mathematics from Brown University, his M.S. degree in Mathematics from the University of Vermont, and his bachelor's degree in Physics from Duke University. He is the author of over 90 publications and is a two-time recipient of the Gordon Bell Prize for demonstrated excellence in high-performance and large-scale parallel computing (1999 and 2000) and the IBM Faculty Award for research excellence (2005 and 2006).

Matthew G. Knepley is an Associate Professor in the Computer Science and Engineering Department of the University at Buffalo. He received his B.S. in Physics from Case Western Reserve University in 1994, an M.S. in Computer Science from the University of Minnesota in 1996, and a PhD in Computer Science from Purdue University in 2000. He was a Research Scientist at Akamai Technologies in 2000–01 in the Distributed Data Collection and Data Analysis groups. Afterwards, he joined the Mathematics and Computer Science department at Argonne National Laboratory (ANL), where he was an Assistant Computational Mathematician, and a Fellow in the Computation Institute at University of Chicago, becoming a Senior Research Associate in 2009. In 2015, he joined the Computational and Applied Mathematics department at Rice University. His research focuses on scientific computation, including scalable algorithms and parallel computing, numerical analysis, software development, high dimensional function approximation, and computational geophysics and biology. He is an author of the widely used PETSc library for scientific computing from ANL, and is a principal designer of the PyLith library for the solution of dynamic and quasi-static tectonic deformation problems. He developed the PETSc scalable unstructured mesh support based upon ideas from combinatorial topology. He was a J. T. Oden Faculty Research Fellow at the Institute for Computation Engineering and Sciences, UT Austin, in 2008, won the R&D 100 Award in 2009, and the SIAM/ACM Prize in Computational Science and Engineering in 2015 as part of the PETSc team.