Planet-in-a-Bottle: A Numerical Fluid-Laboratory System*

Chris Hill[1], Bradley C. Kuszmaul[1], Charles E. Leiserson[2], and John Marshall[1]

[1] MIT Department of Earth and Planetary Sciences, Cambridge, MA 02139, USA
[2] MIT Computer Science and Artificial Intelligence Laboratory, Cambridge, MA 02139, USA

Abstract. Humanity's understanding of the Earth's weather and climate depends critically on accurate forecasting and state-estimation technology. It is not clear how to build an effective dynamic data-driven application system (DDDAS) in which computer models of the planet and observations of the actual conditions interact, however. We are designing and building a laboratory-scale dynamic data-driven application system (DDDAS), called ***Planet-in-a-Bottle***, as a practical and inexpensive step toward a planet-scale DDDAS for weather forecasting and climate model. The Planet-in-a-Bottle DDDAS consists of two interacting parts: a fluid lab experiment and a numerical simulator. The system employs ***data assimilation*** in which actual observations are fed into the simulator to keep the models on track with reality, and employs ***sensitivity-driven observations*** in which the simulator targets the real-time deployment of sensors to particular geographical regions and times for maximal effect, and refines the mesh to better predict the future course of the fluid experiment. In addition, the feedback loop between targeting of both the observational system and mesh refinement will be mediated, if desired, by human control.

1 Introduction

The forecasting and state-estimation systems now in place for understanding the Earth's weather and climate consists of myriad sensors, both in-situ and remote, observing the oceans and atmosphere. Numerical fluid simulations, employing thousands of processors, are devoted to modeling the planet. Separately, neither the observations nor the computer models accurately provide a complete picture of reality. The observations only measure in a few places, and the computer simulations diverge over time, if not constrained by observations. To provide a better picture of reality, application systems have been developed that incorporate ***data assimilation*** in which observations of actual conditions in the fluid are fed into computer models to keep the models on track with the true state of the fluid. Using the numerical simulation to target the real-time deployment of sensors to particular geographical regions and times for maximal effect, however, is only a

---

* This research was supported by NSF Grant 0540248.

Y. Shi et al. (Eds.): ICCS 2007, Part I, LNCS 4487, pp. 1163–1170, 2007.

research area and not yet operational. Under the NSF DDDAS program, we are designing and building a laboratory analogue dynamic data-driven application system (DDDAS) that includes both data assimilation and ***sensitivity-driven observations*** to explore how one might proceed on the planetary scale.

In meteorology, attempts to target adaptive observations to "sensitive" parts of the atmosphere is already an area of active research [11]. Observation-targeting field experiments, such as the Fronts and Atlantic Storm-Track EXperiment (FASTEX) and the NORth Pacific EXperiment (NORPEX), have demonstrated that by using, for example, objective adjoint techniques, it is possible, in advance, to identify regions of the atmosphere where forecast-error growth in numerical forecast models is maximally sensitive to the error in the initial conditions [3]. The analysis sensitivity field is then used to identify promising targets for the deployment of additional observations for numerical weather prediction. Such endeavors, although valuable, are enormously expensive and hence rare, and the experimental results are often not repeatable.

In oceanography, the consortium for Estimating the Circulation and Climate of the Ocean (ECCO [13]) is an operational state-estimation system focusing on the ocean. This consortium is spearheaded by some of the authors of the present project but also involves scientists at the Jet Propulsion Laboratory (JPL), the Scripps Institute of Oceanography (SIO), as well as other MIT researchers. Ours has been a broad-based attack — we have coded new models of the atmosphere and ocean from the start with data assimilation in mind [9,10,8,1]. Both forward and adjoint models (maintained through automatic differentiation [7]) have been developed. We have also collaborated with computer scientists in the targeting of parallel computers [12] and in software engineering [5].

Unfortunately, deploying a DDDAS for ECCO or weather forecasting is currently unrealistic. Scientific methodology is uncertain, the cost would be substantial, and the technical hurdles to scaling to a global, high-resolution system, especially in the area of software, are immense.

We therefore are designing and building a laboratory-scale DDDAS, which we call ***Planet-in-a-Bottle*** (or ***Bottle***, for short), a practical and inexpensive step towards a planet-scale DDDAS. The Bottle DDDAS will emulate many of the large-scale challenges of meteorological and oceanographic state-estimation and forecasting but provide a controlled setting to allow systematic engineering strategies to be employed to devise more efficient and accurate techniques. The DDDAS will consist of two interacting parts: a fluid lab experiment and a numerical simulator. Observations taken from the laboratory experiment will feed into the simulator, allowing the simulator to refine the irregular mesh underlying the simulation and better predict the future course of the fluid experiment. Conversely, results from the numerical simulation will feed back to the physical system to target observations of the fluid, achieving a two-way interplay between computational model and observations. In addition, the feedback loop between targeting of both the observational system and mesh refinement will be mediated, if desired, by human control.

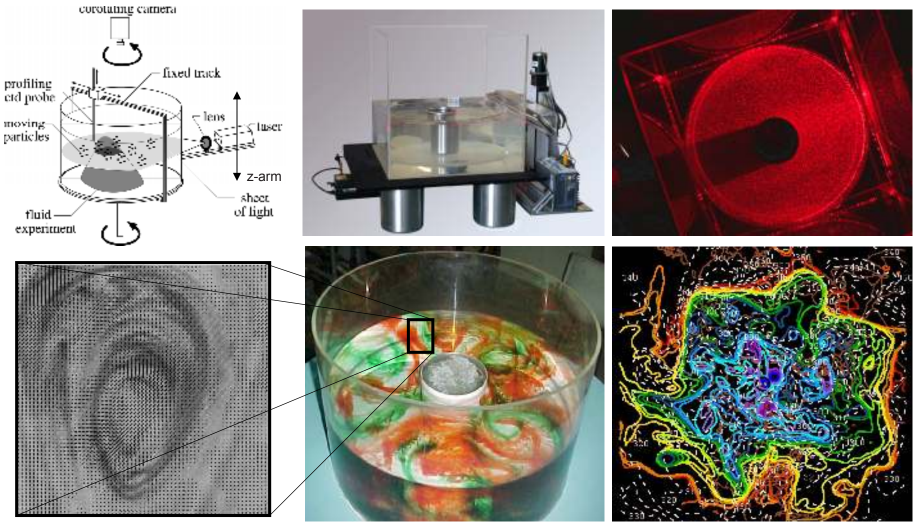

Fig. 1. The Planet-in-a-Bottle laboratory setup. The top row shows, on the left, a schematic of the tank-laser assembly, in the middle, the apparatus itself and, on the right, the laser sheet illuminating a horizontal plane in the fluid for the purpose of particle tracking via PIV. Note that the whole apparatus is mounted on a large rotating table. The bottom row shows, on the left, dye streaks due to circulation in the laboratory tank with superimposed velocity vectors from PIV, in the middle, eddies and swirls set up in the tank due to differential heating and rotation and, on the right, a snapshot of the temperature of the tropopause in the atmosphere (at a height of roughly 10km) showing weather systems. The fundamental fluid mechanics causing eddies and swirls in the laboratory tank in the middle is the same as that causing the weather patterns on the right.

The Planet-in-a-Bottle DDDAS will provide an understanding of how to build a planet-scale DDDAS. By investigating the issues of building a climate modeling and weather forecasting DDDAS in a laboratory setting, we will be able to make progress at much lower cost, in a controlled environment, and in a environment that is accessible to students and other researchers. The MITgcm [10, 9, 6, 2] software that we use is exactly the same software used in planet-scale initiatives, such as ECCO, and so our research promises to be directly applicable to planet-scale DDDAS's of the future. MITgcm is a CFD engine, built by investigators from this project, designed from the start with parallel computation in mind, which in particular runs on low-cost Linux clusters of processors.

When this project started, both the Bottle lab and the Bottle simulator already existed (see Figure 1), but they had not been coupled into a DDDAS system. In particular, the simulator was far too slow (it ran at about 100 times real time), and it lacked sufficient accuracy. Data assimilation of observations occured off-line after the fluid experiment has been run, and the feedback loop that would allow the simulator to target observations in the fluid is not yet available. Building the Planet-in-a-Bottle DDDAS required substantial research that we broke into two research thrusts: dynamic data-driven science for fluid modeling, and algorithms and performance.

The first research thrust involves developing the dynamic data-driven science for fluid modeling in the context of MITgcm running in the Bottle environment. We are investigating ***adjoint models***, which are a general and efficient representation of model sensitivity to any and all of the parameters defining a model. We are studying how adjoint models can be used to target observations to yield key properties of the fluid which cannot be directly measured, but only inferred by the synthesis of model and data. We are also studying their use in guiding the refinement of irregular meshes to obtain maximal detail in the Bottle simulation. We estimate that the computational demands for a practical implementation will be 5–10 times that of an ordinary forward calculation. The implementation also challenges the software-engineering infrastructure we have built around MITgcm, which today is based on MPI message passing and data-parallel computations.

The second thrust of our research focuses on algorithmic and performance issues. The previous Bottle simulation is far too slow to be usable in a DDDAS environment. We believe that the performance of the simulation can be improved substantially by basing the simulation on adaptive, irregular meshes, rather than the static, regular meshes now in use, because many fewer meshpoints are required for a given solution. Unfortunately, the overheads of irregular structures can negate their advantages if they are not implemented efficiently. For example, a naive implementation of irregular meshes may not use the memory hierarchy effectively, and a poor partitioning of an irregular mesh may lead to poor load balancing or high communication costs. We are investigating and applying the algorithmic technology of ***decomposition trees*** to provide provably good memory layouts and partitions of the irregular meshes that arise from the fluid simulation.

2 The Laboratory Abstraction: Planet-in-a-Bottle

The Bottle laboratory consists of the classic ***annulus experiment*** [4], a rotating tank of water across which is maintained a lateral temperature gradient, cold in the middle (representing the earth's pole), warm on the outside (representing the equator). The apparatus is shown in Figure 1. Differential heating of the rotating fluid induces eddies and geostrophic turbulence which transfer heat radially from "equator to pole," and hence the "Planet-in-a-Bottle" analogy. This class of experiment has been a cornerstone of geophysical fluid dynamics and serves as a paradigm for the fluid dynamics and hydrodynamical instability processes that underlie our weather and key processes that maintain the pole-equator temperature gradient of the planet.

We use a video camera to capture the flow in a chosen horizontal plane by introducing neutrally buoyant pliolite particles into the fluid and illuminating them with a laser sheet. The top right panel of Figure 1 shows the laser illuminating pliolite particles at a particular depth. The captured images are recorded at a rate of 30 frames per second with a resolution of 1024 × 780 pixels. The video images are fed to an image-processing system, which calculates the velocity field of the particles using PIV (particle imaging velocimetry). In addition, an array of sensors in the fluid record the temperature. The resulting data is fed into the Bottle simulator. The Bottle simulator consists of a computer system running an ensemble of 30 simulation kernels, each of which runs the MITgcm, a computational fluid dynamics (CFD) code that we have developed.

As illustrated in Figure 2, the simulator process divides time into a series of epochs, each with a duration about the same as one rotation of the laboratory experiment (representing one day, or in the laboratory, typically on the order of 10 seconds). At the beginning of each epoch, the simulator initializes each of the 30 kernels with an ensemble Kalman filter derived estimate of the current state of the fluid based on the observed fluid state and the simulation state from the preceeding epoch. The simulator perturbs the initial state of each kernel in a slightly different manner. The 30 kernels then run forward until the end of the epoch, at which point the results are combined with the next frame of observations to initialize the next epoch.

The DDDAS we are building will enhance the current data assimilation with real-time, sensitivity-driven observation. The assimilation cycle will be preceeded by a forecast for the upcoming epoch. At the end of the forecast, using the adjoint of the forward model, a sensitivity analysis will determine which locations in the fluid most affect the outcomes in which we are interested. The simulator will then direct the motor controlling the laser sheet to those particular heights in the fluid and direct the camera to zoom in on the particular regions. A dynamically driven set of observations will then be made by the camera, the image-processor will calculate the velocity field of the particles, and the simulator will compute a new estimate of the fluid state. In addition, the Planet-in-a-Bottle system will dynamically refine the mesh used by the Kalman filter simulation kernels so that more computational effort is spent in those regions to which the outcome is most sensitive. This forecast and assimilation process will be repeated at each epoch.

This is a direct analogue of the kind of large scale planetary estimates described in, for example, [13]. Everything about the laboratory system — the assimilation algorithms, the simulation codes, the observation control system, the real-time constraints, and the scalable but finite compute resources is practically identical to technologies in operational use in oceanography and meteorology today. Moreover — and this is at the heart of the present project — it provides an ideal opportunity to investigate real-time data-driven applications because we can readily (i) use the adjoint of our forward model to target observations of an initial state to optimise the outcome (ii) refine the numerical grid in regions of particular interest or adjoint sensitivity (iii) experiment with intervention by putting humans in the loop, directly controlling the observations and/or grid refinement process. Although the physical scale of the laboratory experiment is small, it requires us to bring up the end-to-end combined physical-computational system and consider many issues encountered in the operational NWP and

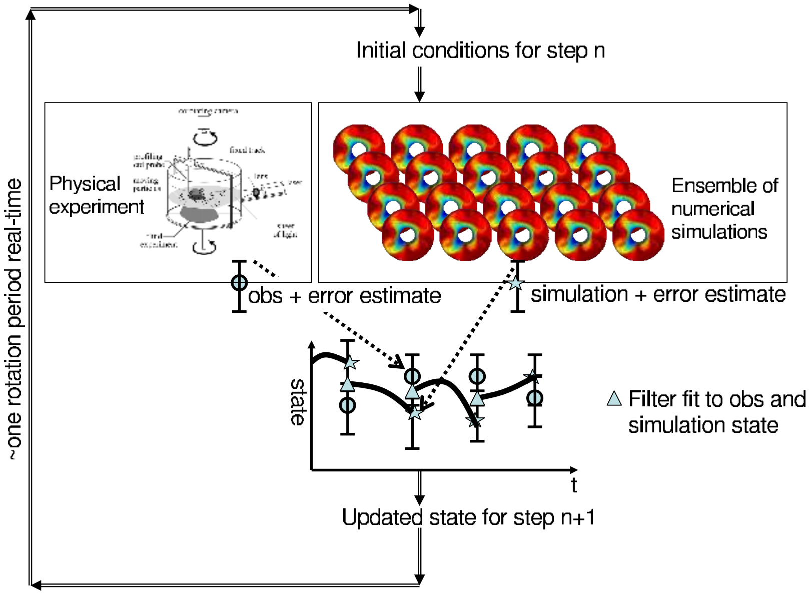

Fig. 2. Schematic of the ensemble Kalman filter. The interval between Kalman filter updates is one rotation period of the fluid experiment (typically ≈ 10s). The physical system (on the left) is observed with noisy and incomplete observations. An ensemble (on the right) of simulation kernels is run, each from a perturbed initial state, producing simulations with an error estimate. A filter combines the observations with the ensembles to yield a new updated state for the next epoch.

large-scale state-estimation communities. Notably, the MITgcm software we currently use for numerical simulation of the laboratory fluid is also in use in planetary scale initiatives. Consequently, innovations in computational technologies tested in the laboratory will directly map to large-scale, real-world problems.

3 Progress

Our team has made significant progress on our Bottle DDDAS in the first year. We have finished developing about 70% of the physical infrastructure, 80% of the computational infrastructure, and have developed a first-version of the end-to-end system demonstrating real-time simulation, measurement and estimation processes.

We have developed robust protocols for physical simulation, instrumented the system to acquire and process data at a high-bandwidth. Currently the system can process about 60 MBytes/sec of raw observational and model data to produce state estimates in real-time. The system includes a camera and a fiber-optic rotary joint, a laser light-sheet apparatus, a robotic arm to servo the light sheet to illuminate a fluid plane, rotating fluid homogenized with neutrally buoyant particles. A thermal control subsystem is presently being integrated to produce climatological forcing signals of temperature gradient (and hence heat flux).

Observations are produced using a particle image velocimetry method, procedures for which were perfected in the first year. A distributed computing infrastructure has been developed for generating velocity measurements. We now frequently use this subsystem for gathering observations.

We have made refinements to the MIT-GCM using a nonuniform domain decomposition, and this is coupled with a 3rd order accurate advection scheme. The model functions in realtime on an Altix 350.

Our goal has been to develop a high-performance application that is also superbly interactive. The observation subsystem is designed to be easily reconfigurable, but the details of the distributed computation are entirely hidden from the user. Similarly, easy to configure model interface is being developed to allow the user to dynamically reconfigure the model parameters whilst leaving the actual implementation of computation out of the picture. Both observations and models perform in realtime, thus the effect of parameter changes are observed in realtime, and this is the basis for user-perceived interactivity. Data-assimilation is implemented using matlab; a language most researchers know. We are incorporating the StarP system and thus the implementation of distributed computation is hidden from the user. Students and researchers alike can use, almost verbatim, their code on the large-scale problem and change algorithms rapidly. We believe that this architecture strikes the right balance between performance and interaction, and we hope to see its benefits as we incorporate it in research and the classroom in the next two years.

We have demonstrated that data-assimilation inference from data and models can also be performed in real-time. We have developed a two-stage approach that automatically switches assimilation from a weak prior (low model skill) mode to a strong prior mode. Both these function in real-time. The latter mode uses an ensemble to construct subspace approximations of the forecast uncertainty. The ensemble is generated robustly by combining perturbations in boundary conditions with time-snapshots of model evolution. This new approach entirely diminishes the computational bottleneck on the model performance because only few model integrations are required, and surrogate information of the state's uncertainty is gleaned from exemplars constructed from the temporal evolution of the model-state. A forty member ensemble (typical in meteorological use) thus requires around 4 separate model simulations; an easy feat to accomplish.

Work on two other ensemble-based assimilation methods has also progressed. First, a fast ensemble smoother (Ocean Dynamics, to appear) shows that fixed-interval smoothing is $O(n)$ in time, and fixed-lag smoothing is independent of the lag length. Second, we have developed a scale-space ensemble filter that combines graphical, multiscale models with spectral estimation to produce rapid estimates of the analysis state. This method is superior to several popular ensemble methods and performs in $O(n \log n)$ of state size n, and is highly parallelizable.

References

1. A. Adcroft, J-M Campin, C. Hill, and J. Marshall. Implementation of an atmosphere-ocean general circulation model on the expanded spherical cube. *Mon. Wea. Rev.*, pages 2845–2863, 2004.
2. A. Adcroft, C. Hill, and J. Marshall. Representation of topography by shaved cells in a height coordinate ocean model. *Mon. Wea. Rev.*, pages 2293–2315, 1997.
3. N. Baker and R. Daley. Observation and background adjoint sensitivity in the adaptive observation-targeting problem. *Q.J.R. Meteorol. Soc*, 126(565):1431–1454, 2000.
4. M. Bastin and P. Read. A laboratory study of baroclinic waves and turbulence in an internally heated rotating fluid annulus with sloping endwalls. *J. Fluid Mechanics*, 339:173–198, 1997.
5. C. Hill, C. DeLuca, Balaji, M. Suarez, and A. DaSilva. The architecture of the Earth System Modeling Framework. *Computing in Science and Engineering*, 6(4):18–28, 2004.
6. C. Hill and J. Marshall. Application of a parallel Navier-Stokes model to ocean circulation. In *Proceedings of Parallel Computational Fluid Dynamics: Implementations and Results Using Parallel Computers*, pages 545–552, 1995.
7. J. Marotzke, R. Giering, K. Q. Zhang, D. Stammer, C. Hill, and T. Lee. Construction of the adjoint MIT ocean general circulation model and application to Atlantic heat transport sensitivity. *J. Geo. Res.*, 104(C12):29,529–29,547, 1999.
8. J. Marshall, A. Adcroft, J-M. Campin, C. Hill, and A. White. Atmosphere-ocean modeling exploiting fluid isomorphisms. *Monthly Weather Review*, 132(12):2882–2894, 2004.
9. J. Marshall, A. Adcroft, C. Hill, L. Perelman, and C. Heisey. A finite-volume, incompressible Navier Stokes model for studies of the ocean on parallel computers. *J. Geophys. Res.*, 102, C3:5,753–5,766, 1997.
10. J. Marshall, C. Hill, L. Perelman, and A. Adcroft. Hydrostatic, quasi-hydrostatic and nonhydrostatic ocean modeling. *J. Geophys. Res.*, 102, C3:5,733–5,752, 1997.
11. T.N. Palmer, R. Gelaro, J. Barkmeijer, and R. Buizza. Vectors, metrics, and adaptive observations. *J. Atmos. Sci*, 55(4):633–653, 1998.
12. A. Shaw, Arvind, K.-C. Cho, C. Hill, R. P. Johnson, and J. Marshall. A comparison of implicitly parallel multi-threaded and data-parallel implementations of an ocean model based on the Navier-Stokes equations. *J. of Parallel and Distributed Computing*, 48(1):1–51, 1998.
13. D. Stammer, C. Wunsch, R. Giering, C. Eckert, P. Heimbach, J. Marotzke, A. Adcroft, C. Hill, and J. Marshall. Volume, heat, and freshwater transports of the global ocean circulation 1993–2000, estimated from a general circulation model constrained by World Ocean Circulation Experiment (WOCE) data. *J. Geophys. Res.*, 108(C1):3007–3029, 2003.