From *SIAM News*, Volume 38, Number 5, June 2005

Equation-Free Computing Gets Something from (Almost) Nothing

A "fundamental and innovative framework for multiscale computation in complex dynamical systems," recognized by the SIAG/DS Crawford Prize, could enable computational scientists to work with models of physical systems for which macroscopic equations are not available.

By Barry A. Cipra

Pose a problem to a mathematician, and chances are an equation is soon being scribbled on the nearest napkin. Mathematicians are addicted to equations the way teenagers are addicted to cell phones—they can't imagine a world without them. But Ioannis Kevrekidis can. And he is on a mission to break mathematicians' dependence on equations.

Kevrekidis, who has joint appointments in the department of chemical engineering and the program in applied and computational mathematics at Princeton University, is spearheading a drive toward "equation-free" computing. He and colleagues, especially C. William Gear of Princeton University, are developing techniques that enable computational scientists to work with models of physical systems for which macroscopic equations are unavailable. Applications range from catalytic surface reactions to computational neuroscience.

Revenge of the Reductionists

"Equation-free" is an eye-catching—and, up to a point, accurate—label. A more prosaic description for this work is "enabling microscopic simulators to perform system-level tasks." The basic idea is that although many phenomena are relatively easy to describe on a fine level, those descriptions, by themselves, give little insight into the macroscopic properties that are of actual, practical interest. Roughly speaking, this is the conundrum of reductionism: Even if psychology boils down to (neuro)biology boils down to (bio)chemistry boils down to quantum physics, it's tough to derive a toddler's temper tantrum from Schrödinger's equation.

The traditional approach of science is to seek valid laws *at each level of interest*. The role of the reductionist is to show how high-level laws arise from the primordial ooze. Many philosophers of science have expressed doubts as to whether this can be done at all. According to anti-reductionists, the behavior of complex systems may not be derivable from the lower-level laws that govern them. A semi-anti-reductionist might add that even when complex behavior *is* implicit in the low-level formulation, making use of that fact computationally may be an intractable problem: The femtosecond scale of molecular dynamics, for example, does not (currently) lend itself to millisecond-scale computations.

Practical scientists and engineers don't spend much time fretting about philosophical issues. As long as there is a good model at the right level, they are perfectly happy to ignore the theoretical chasms—especially if the model comes in the form of equations. Equations are nice because you can "do math" with them. Sometimes you can solve them outright. Other times, they can be manipulated into forms that improve the performance of numerical methods for obtaining answers.

But what happens in the absence of a model, of equations, at the appropriate scale?

Enter equation-free computing.

Micro Meso Macro

"We're going after problems that if we were smart enough we'd have a simple description, but we're not smart enough," Kevrekidis says. "If the difficulty is in obtaining the right equation, then maybe one can get results without it."

Equation-free computing has four key steps: Lift, Simulate, Restrict, and Use the Results to Do the Desired Numerics. Equation-free integration in time, for example, uses estimation and projection forward in time after the Lift/Simulate/Restrict steps (see Figure 1). "Lifting" gives a one-to-many map from initial conditions at the macroscopic level to a consistent ensemble of initial conditions at the microscopic level. "Simulation" is done at the microscopic level. "Restriction" provides a many-to-one map back to the macroscopic level. Typically, the (unknown) macroscopic equations are characterized by "fast" and "slow" variables.

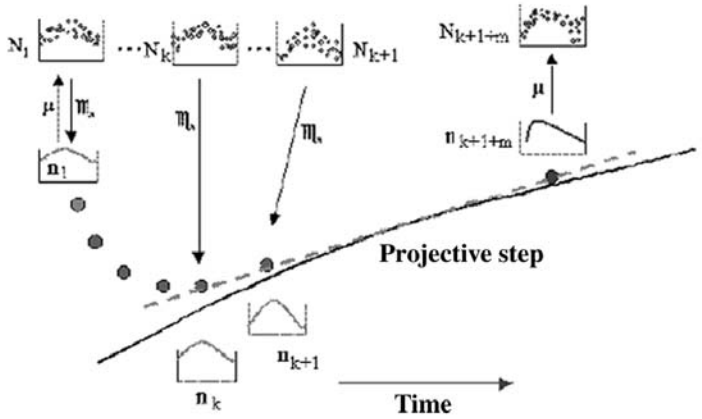

Figure 1. *Equation-free computing (specifically coarse projective integration) in a nutshell.*

The fast variables take you quickly to a slow, invariant manifold, much as shock absorbers damp out the effects of potholes.

Once the slow variables take over, it's possible to estimate time derivatives. This leads to the crucial projection step, in which these estimates are used to extrapolate on a "mesoscopic" scale: huge compared with the microscale, but still small on the macroscale. An equation-free approach to integration is, in effect, the forward Euler method for solving equations of the form $x'(t) = f(x,t)$ when you don't have a formula for the function $f(x,t)$ but can obtain estimated values of x' by incrementing the initial condition $x(0)$ forward in time through the extremely small steps of the microscopic simulator (see Figure 2).

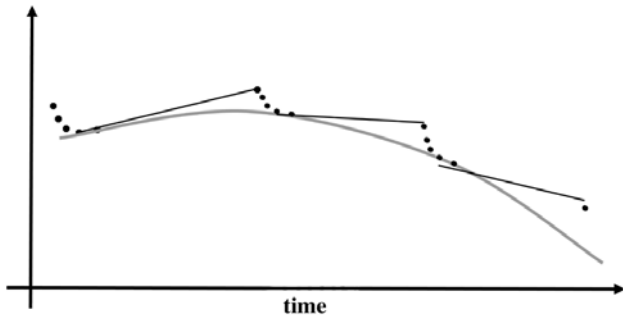

Figure 2. *Lather, rinse, repeat.*

Kevrekidis likens this to what a laboratory experimentalist might do if it were easy to initialize exper-imental conditions at will. Suppose, for example, that you're studying a process that changes A to B (pressed grapes into a fine wine, say) over the course of many years; you want to know exactly (meaning approximately) how long it takes for this to happen. Suppose also that the rate of change depends only on the ratio of A to B, but that the exact (meaning exact) dependence is unknown. Then, starting from a particular ratio, it might be possible to predict what the ratio will be a year later by closely observing the process over a few days. A new experiment could then be conducted, starting from the predicted ratio, and so forth. In this way, a 20-year process could be studied in a matter of weeks.

Choosing an appropriate step size for the great leap forward is, of course, a delicate matter. But numerical analysts put that problem to rest a long time ago, Kevrekidis points out: "The technology for adaptively choosing the right step size is already there. You don't have to reinvent it."

Endless Opportunities

Equation-free computing can also handle spatially distributed systems. Here the unknown equation is a PDE. The equation-free approach starts with microscopic descriptions (i.e., liftings) on a set of short, widely spaced intervals (or squares, cubes, and so forth, if there is more than one spatial dimension). Microsimulation and restriction turn these liftings into patches. The patches propagate forward by projection, based on the estimated time derivatives; for adjustments, the patches are allowed to "talk" to one another via their estimated values and space derivatives (see Figure 3).

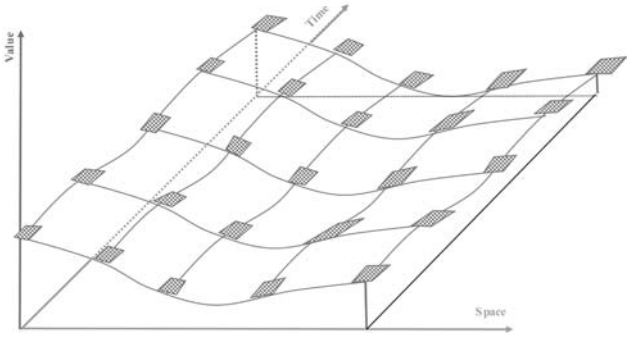

Figure 3. *Patch dynamics handles problems governed by an unknown PDE. (The grid shown here uses curved connecting lines to suggest that, when all is said and done, it may be desirable to smooth out the computed solution using, say, splines.)*

Nature and technology offer endless opportunities for applications of equation-free computing. One intriguing possibility arises with "legacy codes": old computer programs so complex—and often so poorly documented—that trying to change them is out of the question. Imagine, for example, that you have a legacy code for some kind of multidimensional dynamics, and you would like to find its fixed points. The code acts as a function call, taking x from its present value to a future value, $f(x)$. If the function were given by a formula, the Jacobian for the function $F(x) = x - f(x)$ would suffice, via the Newton–Raphson method, to solve the problem. But even without a formula, you can estimate the action of the Jacobian by computing $F(x)$, using the legacy code for the $f(x)$ part, at a slew of nearby points (the microsimulation). The possibility of doing this was proposed in 1993, by Gautam Shroff, now at Tata Consultancy Services in India, and Herb Keller of Caltech (*SIAM Journal on Numerical Analysis*, Vol. 30, pages 1099–1120). Kevrekidis credits the paper as providing the basic inspiration for his work.

Legacy codes can be used in similar ways to do bifurcation analysis, controller design, and optimization, among other tasks. In other words, it's possible to trick legacy code into computing things it was not designed to compute, by "wrapping" it inside an appropriate equation-free algorithm. This approach "opens new possibilities," Kevrekidis says.

The equation-free literature is burgeoning. Kevrekidis, Gear, and Gerhard Hummer of the National Institutes of Health recently wrote an overview of the subject for the American Institute of Chemical Engineers (*AIChE Journal*, Vol. 50, July 2004, pages 1346–1355), with dozens of references. A more extensive review article, "Equation-free Coarse-grained Multiscale Computation: Enabling Microscopic Simulators to Perform System-level Tasks," by Kevrekidis, Gear, J.M. Hyman, P.G. Kevrekidis, O. Runborg, and K. Theodoropoulos, appeared in *Communications in Mathematical Sciences* in 2003 (Vol. 1, pages 715–762); the original version (2002) can be found at arxiv.org/abs/physics/0209043.

Kevrekidis has worked with numerous researchers on a variety of applications. He and Hummer, for example, have worked on accelerating molecular dynamics computations for the interactions of water with a carbon nanotube (Figure 4) and for the alanine dipeptide molecule.

Even the idea of equation-free experimentation is not outlandish, Kevrekidis says. It's reasonable to imagine a physical control system in which extensive sensors and actuators are used for computationally based experiments. "What we would really like to do," he says, "is to accelerate experiments by doing only a few short, appropriately initialized experiments—find where the system

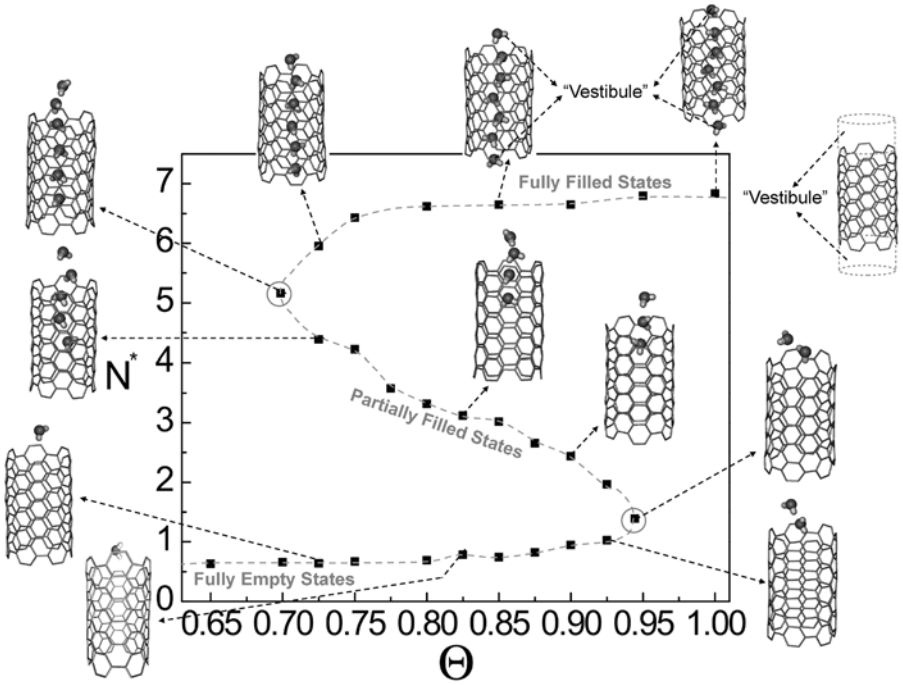

Figure 4. *The effects of the attraction (Θ) between water and a carbon nanotube on pore hydration (N) can be computed via an equation-free approach.*

will go, without waiting for it to go there by itself, which might take a long time." Indeed, future scientists might "do math" directly on complex systems without ever knowing the laws that govern them.

Barry A. Cipra is a mathematician and writer based in Northfield, Minnesota.