Numerical Methods in Economics

Notes for Lecture 1: Introduction

Kenneth Judd
Hoover Institution

February 17, 2020

The Role of Computation in Economic Analysis

- Traditional roles
  - Empirical analysis
  - Applied general equilibrium
- Nontraditional roles
  - Substitute for theory
  - Complement for theory
- Questions:
  - What can computational methods do?
  - Can we trust the results?
  - Where does computation fit into economic methodology?

Progress in Hardware

- Moore's law for semiconductors implied speed doubling every two years
- "Moore's law" was true for the entire 20th century
- Future technologies
  - Three-dimensional semiconductors
  - Asynchronous chips
  - Optical computing
  - Quantum computing

Progress in Computer Architecture and Hardware

- Parallelism
  - Combine many cheap processors on fast communication networks
  - Combine CPUs over ethernet connections: network computing, distributed computing, grid computing
- GPU
- FPGA (Field programmable gate array)

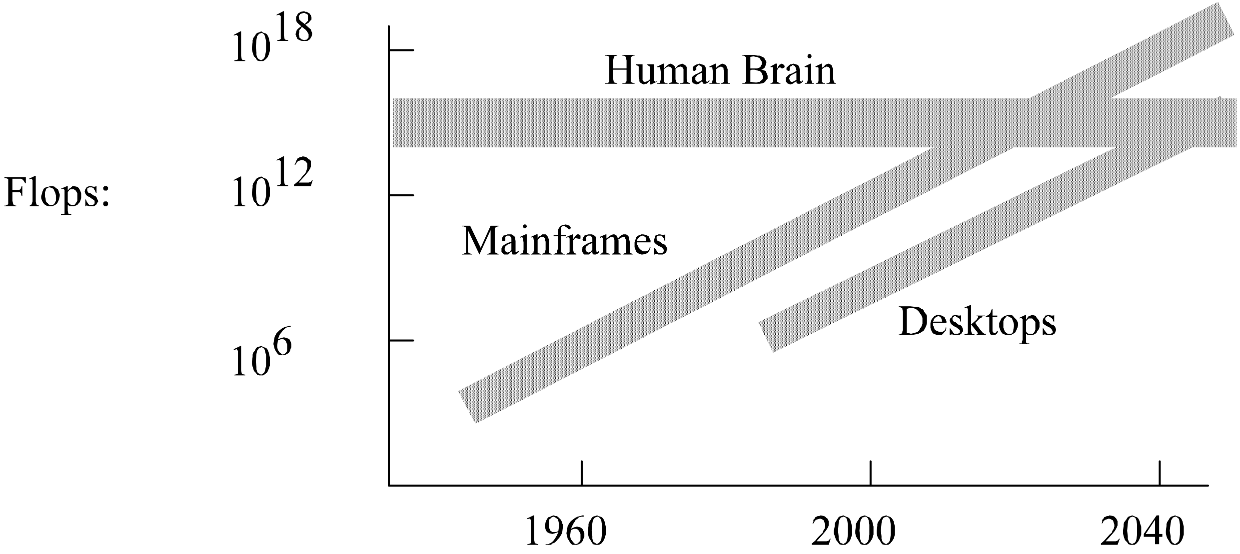


Figure 1: Trends in computation speed: flops vs. year

Progress in Numerical Analysis

- Great progress in last 30 years
  - Optimization
    - Linear programming - Interior point methods
    - Mathematical Programming - Interior point methods
    - MPEC - Mathematical Programming with Equilibrium Constraints
    - MPCC - Mathematical Programming with Complementarity Constraints
  - Equations
    - Nonlinear equations
    - nonlinear complementarity problems (e.g., general equilibrium)
  - Symbolic math
    - Computational algebra
    - Automatic differentiation
  - Approximation
    - Sparse grids
    - Machine learning
    - Neural networks

Progress in Computer Software

- Languages and software tools
  - C++, Python, Julia
  - MPI, OpenMP, TAO
  - CASADI, AdiMAT
  - TensorFlow
  - Bertini, PHCPack, Singular, Magma
- User-friendly interfaces - AMPL, GAMS, Mathematica, R, Python, Maple

US Gov't and Computation: NNSA

- Mandate: Make sure that POTUS can incinerate hundreds of millions of people on short notice.
- What are computational needs of NNSA?
  - Congress: "The Administrator for Nuclear Security shall develop and carry out a plan to develop exascale [billion billion flops] computing."
  - DOE: "Since nuclear weapon simulations must extrapolate far beyond available data and must predict coupled, multi–scale physical phenomena that are difficult to isolate in experiments, verification and validation is a significant unifying challenge to Stockpile Stewardship."
- They take their mandate seriously, and use ...

Sky Bridge

(Grandfather of Sky Net??)

Trinity


CRAY
Alliance for Computing at Extreme Scale
NNSA
Los Alamos

Sierra

- What are the methodological demands on (and innovations of) NNSA?
  - Verification: Check that your code actually solves your equations
  - Uncertainty Quantification: Determine how parameter uncertainty affects results
- We bring these tools to economics

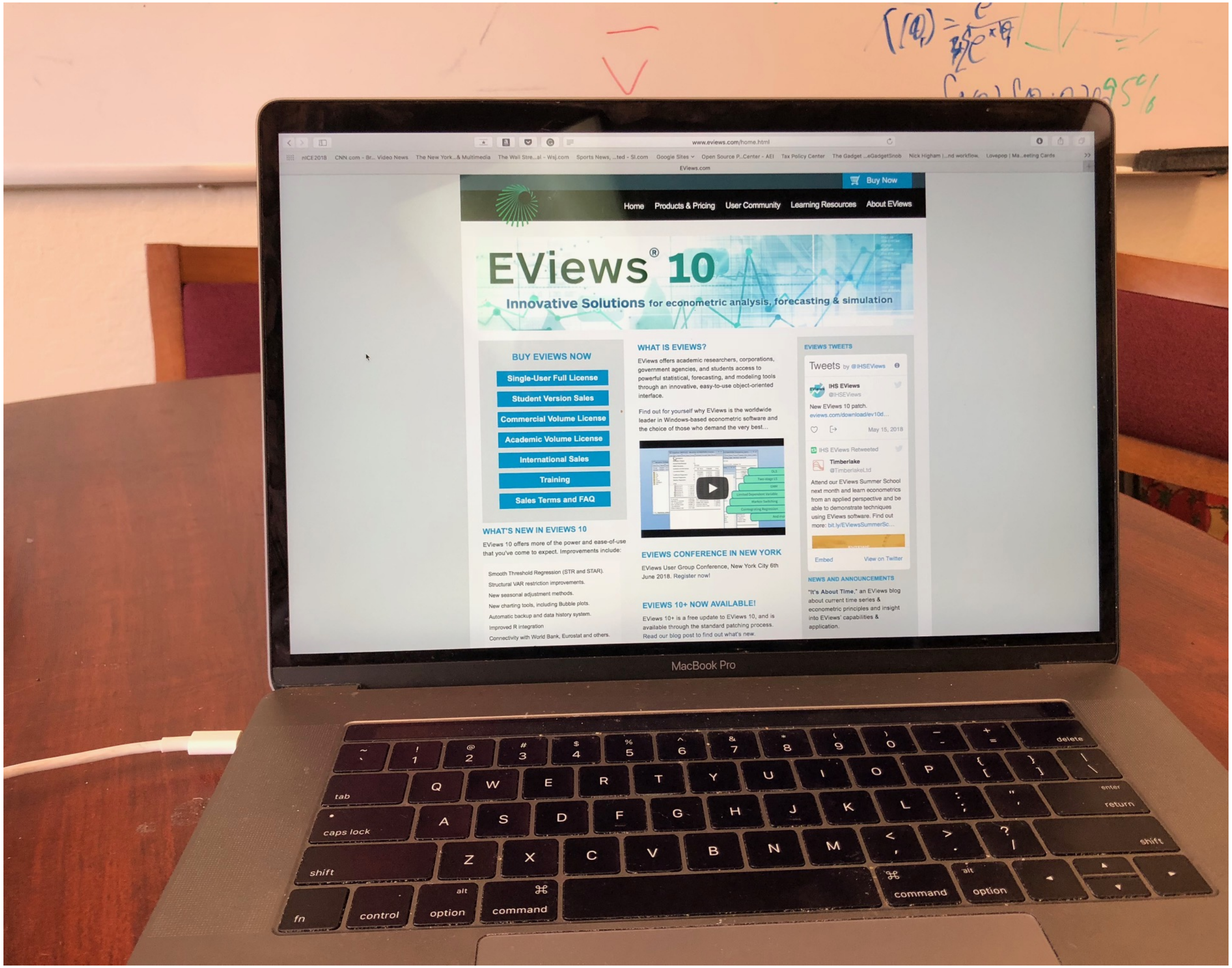
Where is Economics?

- The computational approach has enormous potential for economic analysis, but very little is being exploited.
- In the opinion of an applied mathematician at MIT, who knows economists, "Economists will soon be so far behind they will not be able to catch up."
- Math vs. Hardware: Substitutes or Complements?
  - Computational mathematicians: Give them a computer 100x faster, and they will then develop algorithms 100x better than old algorithms - net gain is a factor of 10,000
  - Economists: Give them a computer 100x faster, and they will say "Great! I can forget 90% of the math I know and still get a 10x speedup!"

US Federal Reserve

- Mandate:
  - Price stability
  - Full employment
- Other responsibilities include
  - Lender of last result
  - Regulation of financial institutions
- My view follows the DOE: "Since dynamic economic simulations must extrapolate far beyond available data and must predict coupled, multi–scale social phenomena that are difficult to isolate in experiments, verification and validation should be a significant unifying challenge to studies of economic policies."
- What computational tools do they use?

Federal Reserve



Attitude of Economics Towards Numerical Methods

- Very few economics departments offer students serious training in computational methods.
- A sample of what is taught in one actual "course":
  - Use the simplest possible methods.
  - Use methods that are as transparent as possible (i.e., the computer code reflects the economic structure of the problem).
  - Use methods where you can watch the computations as they proceed.
  - Use one-dimensional algorithms as much as possible.

Conventional Wisdom versus Facts

- Much of what I teach in this course will contradict what you were supposed to “learn” from economics papers, lectures, and seminars.
- Some say that you cannot trust the computations done in economics papers; they are correct

Optimization Methods

- CW: There have been no advances in optimization algorithms in the past 45 years that would be useful in economics (for example, see lead article of the diamond anniversary volume of Econometrica).
- CW: Automatic differentiation is not a valuable tool
- CW: Stay with simple methods, motivated by economic intuition; stay away from "magical black boxes"
- Facts:
  - Robert Wilson's 1964 Ph. D. thesis introduced "sequential quadratic programming," a key method even today.
  - "Interior point" methods were first proposed by Ragnar Frisch, but became tractable only in the 1980's
  - Bob Hall implemented automatic differentiation ideas in the first versions of TSP in the mid-1960's.
  - A box ceases to be black when you open your eyes and turn on the lights.

Structural Estimation

- CW: Due to computational challenges, one must sacrifice statistical efficiency and use "computationally light" estimators.
- CW: The Nelder-Mead algorithm (a.k.a. "amoeba", a slug with no brain) is good enough for empirical IO.
- CW: Econometrica and Econometric Society dogma declares that estimation requires solving for all equilibria for each parameter value examined by the estimation procedure, a particularly difficult task for estimating games.
- Fact: Several papers by Che-Lin Su and collaborators show otherwise.

Dynamic Programming

- CW: It is difficult to write DP code that is stable, efficient, and accurate, particularly for multidimensional problems.
- Fact: It is easy to do this for concave problems once you learn a little math.
- Fact: Not hard for smooth problems of moderate dimension
- Fact: Not hard for high-dimensional economic problems

Heterogeneous Agents

- CW: It is difficult to solve problems with substantial heterogeneity.
- Fact: There are basic methods capable of solving models with dozens, sometimes hundreds, of types of agents without sloppy aggregation.

Dynamic Games

- CW: Finding feedback equilibria (a.k.a. MPE) is very difficult.
- Fact: Life is always hard if you use only Gauss-Jacobi and Gauss-Seidel methods, methods whose authors have long been dead.
- Fact: Nonlinear equation solvers can solve these games
- Fact: Homotopy methods can explore possibility of multiple equilibria

Computer Resources

- CW: It is difficult to use parallel systems.
- CW: Economists cannot get access to supercomputers.
- Fact: I got 1.4 million node hours (which was 10-20 million core hours depending on node choice) over the past seven years on Blue Waters, the lead NSF supercomputer
- Fact: Swiss researchers doing economics have gotten access to Piz Daint, the fastest supercomputer in Europe.
- Fact: See JPE (December, 2019) for a paper that applied supercomputers to basic issues in climate change policy analysis.
- Fact: There are multiple ways to get substantial computer time

Blue Waters

Polynomial Equations

- CW: There are no closed-form solutions for polynomial systems of equations
- Fact: Hilbert's Nullstellensatz, Buchberger
- Fact: Software developed in last 20 years allows for solution of nontrivial economics problems
- Fact: We now know how to use massively parallel architectures

Numerical Integration

- CW: It is not tractable to accurately compute multidimensional integrals with numerical quadrature; you must use Monte Carlo
- CW: Bakhvalov (1959) proved the intractability of numerical integration
- CW: Monte Carlo integration is good enough for econometrics
- Fact: READ Bakhvalov and you will see he also showed that the integrals computed in economics are tractable
- Fact: READ Stroud, Secrest, Weyl, Neiderreiter, Smolyak, ... and you will find many tractable methods
- Fact: We can often greatly improve econometric estimation by using non-random methods

Two great questions from students

- If economics is so hostile to computational science, why am I taking this course?
  - You want to do sound research
  - As an economist, you only need to do the computations and verify the results
  - If people don't ask how you got the results, then don't tell them
  - USE computation to DO economics
- How can I know what papers I should trust?
  - Check to see if they cite papers written by experts in computational mathematics.
  - I always will; most economists won't (and can't).
  - I once asked a macroeconomist to describe the relationship between his use of a mathematical term and how it is used by mathematicians. His response was "We should talk about this after my talk because you are the only person who will understand my answer." Some of the roughly 200 macroeconomists in the room understand that he had insulted them.

What can we compute now?

- Dynamic optimization: optimal control, dynamic programming
- Mechanism design: contracting, optimal taxation, nonlinear pricing
- General equilibrium
  - Arrow-Debreu general equilibrium, complete and incomplete markets
  - Dynamic, perfect foresight models and stochastic recursive models
- Games
  - Finite games
  - Correlated equilibria
  - Feedback equilibrium of dynamic games (a.k.a., Markov perfect equilibrium)
  - Supergames: APS, supergames with states, reputation models
- Econometrics
  - Structural estimation: maximum likelihood, method of moments
  - Bayesian methods
  - Level sets of likelihood functions, moment criteria

Why am I doing this?

- Money? Fame? yeah. sure. dream on.
- Almost no one else is. If not me, who?
- Economics needs researchers who know computational science
- I need RAs and Postdocs!

Why are you here?

- You realize (correctly) you have nothing better to do with your time
- You want to learn computational methods
- You want to do solid, pathbreaking research
- You want to meet others who want to work on serious computational methods
- You might like to spend some time in sunny California

Themes of this course

- Economic modeling is applied mathematics.
  - Describe models using math
  - Determine what kind of problem you have
- Use the most reliable and accurate methods.
  - They are also likely the fastest methods
  - Learn the basic ideas behind them, so you have some idea of how to proceed when your efforts fail.
- DO NOT spend time developing or coding up algorithms.
  - Learn how to use the professionally written software.
  - Your job is economics, not computer science or math.
- DO NOT trust the results. VERIFY their accuracy.
  - Doveryáy, no proveryáy (Trust but verify)
  - Verification is a key step in modern computational science
  - You will learn how to verify the quality of your solutions
- Use the most powerful hardware you can get
  - Power makes debugging go much faster
  - Power allows you to handle larger problems
  - Power allows you to do serious verification

A Possible Future for Computational Economics

- Hardware and Software
  - Computing costs continue to decrease
  - New computing environments and technologies can be exploited
  - Economists (hopefully) catch up to the frontiers in computational math, computer hardware, and computer algorithms.
  - Numerical analysis develops methods to exploit new technologies
  - Economists develop specific methods (as in CGE, Nash equilibrium)
- An Economic Theory of the Future
  - Inputs: Human time and computers
  - Outputs: Understanding of economic systems
  - Trend: Falling price of computation
  - Prediction: Comparative advantage principles imply
    - Humans specialize on formulating economic concepts and models.
    - Computers use their power for tedious tasks such as solving equations, and describing the implications of the concepts and models