Climate Policy Modeling with HTCondor

Dr. Kenneth L. Judd
Senior Fellow*
Hoover Institution⌗

* *NOT* a Stanford professor

⌗ Located on the Stanford University campus

Abstract: HTCondor was used to develop DPSOL, a framework for solving dynamic programming problems. It was the foundation for DSICE which merged dynamic and stochastic factors in economics and the climate, and analyzed the social cost of carbon, the cost of a two-degree target and the value of carbon capture and sequestration. DPSOL is just one example of how economics can exploit high throughput computing.

Nature of Economics Modeling

Data Limitations of Economics

- Policy cannot be based on actual experience
  - We don’t want to wait until we have dozens of “Great Recessions”
  - We need to use the information we have to create models
  - Then solve the models to evaluate effects of alternative policies
  - We can’t do real experiments.
- Economists should proceed like engineers
  - You understand each part, but do not understand how the complex system works
  - Build computer models of designs and find efficient ones
  - Look carefully at failure modes
- Economists
  - Have a fuzzy understanding of individuals
  - Focus on “tractable” models in search for good policies
  - Focus on average performance, not bad scenarios

Stanford University Social Scientists and Computing

I have worked hard trying to get Stanford PhD students and professors to use serious computing.

I could not overcome the enormous resistance.

Here are two of my failures:

John C. Williams

President and Chief Executive Officer
Federal Reserve Bank of New York
33 Liberty Street
New York, NY 10045

john.c.williams@ny.frb.org

You probably have a low opinion of macroeconomists at the Federal Reserve.

Excuses on why macroeconomists did not prevent 2008
- Weakness of computational power of laptops
- "… the limits of the human mind …"

You get what you pay for:
- Average salary for PhD economist in Fed System is $200K (for a total > $200M)
- Top economists at public universities get $600K

JOHN KARL SCHOLZ

John Karl Scholz is the former UW-Madison provost and Dean of the College of Letters & Science, now currently serving as the 19th president of the University of Oregon.

Many times I told him he could do much better research if used good computational methods.

He decided to go into university administration.

HTCondor

Stanford is not interested in supporting my computational needs

I once told Provost Drell that some universities provide substantial computational resources for free.

Her response was “That won’t happen here.”

HTCondor made it possible for me to do serious computing

Miron gave me “unlimited access at zero priority”

This was perfect 20 years ago

I am here now because I want to get on OSG

Wavefront abstraction -- Perfect for Economics

Cluster Comput (2010) 13: 243–256
DOI 10.1007/s10586-010-0134-7

Harnessing parallelism in multicore clusters with the All-Pairs, Wavefront, and Makeflow abstractions

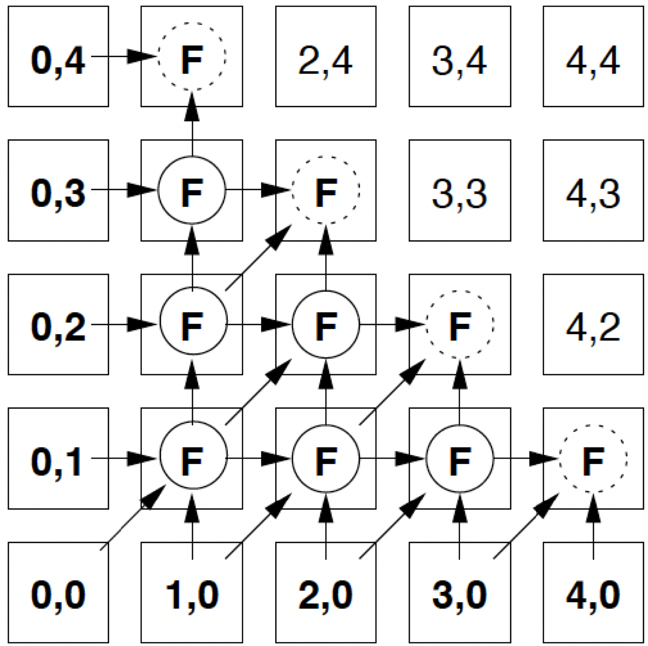
Li Yu · Christopher Moretti · Andrew Thrasher · Scott Emrich · Kenneth Judd · Douglas Thain

Many big problems can be broken into many small ones

2-D:

0 = F[R[i, j], R[i-1, j], R[i-1, j-1], R[i, j-1]] implictly defines R[i, j] in terms of (R[i-1, j], R[i-1, j-1], R[i, j-1])

n-D is a direct generalization



HTCondor tools:

- Master-Worker
- DAGMAN

Progress in computations

Two-D

- Start with initial condition at (0,0)
- Solve problems at (1,0), (0,1) -- 2 problems
- Solve problems at (2,0), (1,1), (0,2) -- 3 problems
- ...
- Solve n problems at level n -- asynchronously!!

n-D

- Even better

Ideal for High-Throughput computing

- Enormous number of individual problems
- Mostly asynchronous, with few load balancing problems
- Can adjust computing burdens to available resources

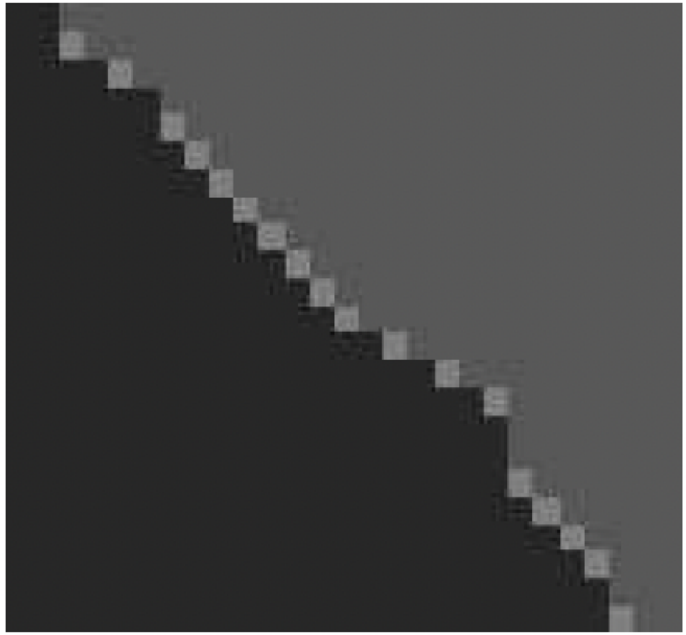


Figure 8: Asynchronous Progress in Wavefront

A progress display from a Wavefront problem. Each cell shows the current state of a portion of the computation: the darkest gray in the lower left corner indicates incomplete, the lighter gray in the upper right indicates complete, and the light cells in between are currently running. The irregular progress is due to heterogeneity and asynchrony in the system.

Dynamic Programming

Simple example

Objective: Manage portfolio to maximize value of wealth at retirement time T.

Decisions: If portfolio is Z at time t, then make investments I(Z, t).

Dynamic law of motion: Assets change value between times t and t+1

Mathematical problem: Approximate investment rule Inv(Z,t)

Useful fact: Once you know Inv(Z, T-1), Inv(Z, T-2), .., Inv(Z, t+1) you can compute Inv(Z,t)

Published paper

> Comput Econ (2015) 45:261–284
> DOI 10.1007/s10614-014-9419-x
>
> **Solving Dynamic Programming Problems on a Computational Grid**
>
> **Yongyang Cai · Kenneth L. Judd · Greg Thain · Stephen J. Wright**

Yongyang Cai PhD thesis (2008)

Used Master-Worker

Used up to 200 CPUs at Condor at UW.

There was nothing at Stanford that he could use -- “The cobbler’s children have no shoes”

DSICE: Climate-Economics Application

State-of-the art in 2010

US government issued report on climate policy based on Nordhaus book.

- Based on laptop computations
- Made large errors in computing social cost of capital -- results were 25% too high
- Why? Nordhaus assumed that there is warming today in response to CO_2 emitted 10 years *in the future*.

In 2018, Nordhaus receives a Nobel(??) Prize for this work.

Cai & Judd advances the modeling (2010-2015)

Uncertainty in economics

Previous models assume that the economy is perfectly predictable -- a differential equation

Cai-Judd add economic uncertainty -- a HJB problem (fortunately a parabolic PDE)

Uncertainty in climate

Cai-Judd allows for tipping points -- adding a jump process to the problem

Robust analysis

Typical models assume they know the key functions in the model.

Cai-Judd uses computer power to examine robustness of results.

Used Blue Waters (a supercomputer)

High throughput systems were not big enough for this work in 2010.

The lessons for climate change discussions

Physics versus economics

- Physics is an IVP: initial conditions imply future path
- Economics is a 2PBVP: expectations matter!

The challenge

- Create models that incorporate economic and climate uncertainty
- Use data to tell us which models are most plausible
- Identify the potential risks
- Develop appropriate policies
  - Consider the insurance value of policies, not just the rate of return
  - Development of new technology avoids the political difficulties of enforcing mitigation.

The biggest challenge: getting economists on board!

Climate Change Policy Analysis

Question: What can and should be the policy response to rising CO2 concentrations?

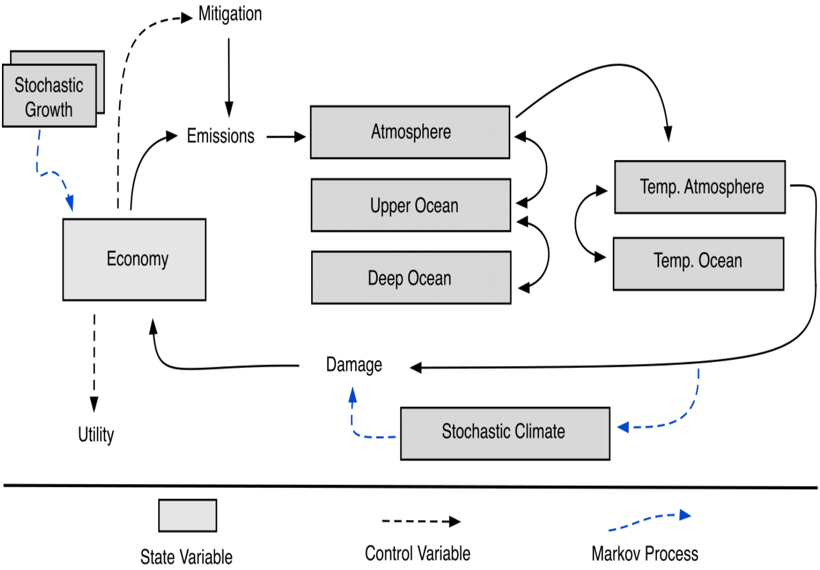
We build a dynamic and stochastic integrated framework for models of climate and the economy (DSICE)

- Economic risk
  - uncertain economic growth with persistence in growth rates
  - flexible preferences that represent risk aversion (Epstein–Zin)
- Climate risk
  - damages interact with economic shocks
  - climate events are stochastic; e.g., glaciers melting, THC collapse
- Model uncertainty
  - We do not know what models for economy and climate are best
  - We do not exactly know key parameters in any specific model

II: The DSICE Framework

We build on Nordhaus DICE model
It is the most widely used model
A useful benchmark which allows us to determine how adding uncertainty and risk affects results from well-known analyses

Dynamic Stochastic Integration of Climate and Economy



Bellman Equation

- Bellman equation for the dynamic stochastic problem:

$$
\begin{aligned}
V_t(\mathbf{S}) = \max_{C,\mu} \quad & u_t(C, L_t) + \beta \left[\mathbb{E}_t \left\{ \left(V_{t+1}\left(\mathbf{S}^+\right)\right)^{\frac{1-\gamma}{1-\frac{1}{\psi}}} \right\} \right]^{\frac{1-\frac{1}{\psi}}{1-\gamma}}, \\
\text{s.t.} \quad & K^+ = (1-\delta)K + Y_t - C - \Psi_t, \\
& \mathbf{M}^+ = \Phi_M \mathbf{M} + (E_t, 0, 0)^\top, \\
& \mathbf{T}^+ = \Phi_T \mathbf{T} + \left(\xi_1 \mathcal{F}_t\left(M_{\mathrm{AT}}\right), 0\right)^\top, \\
& \zeta^+ = g_\zeta(\zeta, \chi, \omega_\zeta), \\
& \chi^+ = g_\chi(\chi, \omega_\chi), \\
& J^+ = g_J(J, \mathbf{T}, \omega_J)
\end{aligned}
$$

- Nine-dimensional state vector: $\mathbf{S} = (K, \mathbf{M}, \mathbf{T}, \zeta, \chi, J)$
- Two control variables: C, μ
- 600-year horizon; annual time steps; terminal value function

Computational Method

- DSICE:
  - six-dimensional continuous state variables $\mathbf{x} \equiv (K, \mathbf{M}, \mathbf{T})$
  - three-dimensional discrete state variables $\theta \equiv (\zeta, \chi, J)$ with $91 \times 19 \times 16$ time-dependent values
- Solve backwards in time
  - A value function $V_t(\mathbf{S})$ represents economic system at time t as a function of $\mathbf{S} = (\mathbf{x}, \theta)$
  - Terminal condition: $V_T(\mathbf{S})$ known for time T
  - Decisions today depend on expectations of what will be done tomorrow
  - Backward induction:

$$V_t = \mathfrak{F}_t V_{t+1}$$

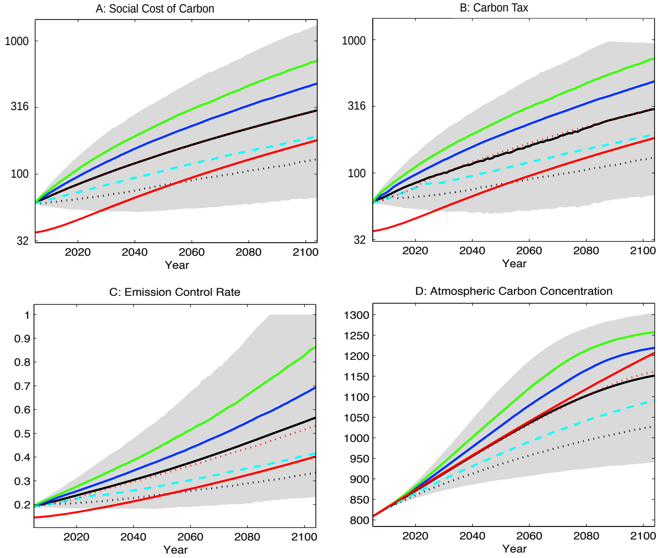
Computational Challenges and Parallelization

- Computational Challenges
  - 100:1 ratio in maximum to minimum capital stock over next 200 years
  - substantial range in climate state variables
  - value function is strongly nonlinear
- Size of computation
  - Approximation: 1.5 billion approximation nodes
  - Optimization: 372 billion optimization problems
- Massive Parallelization in DSICE
-

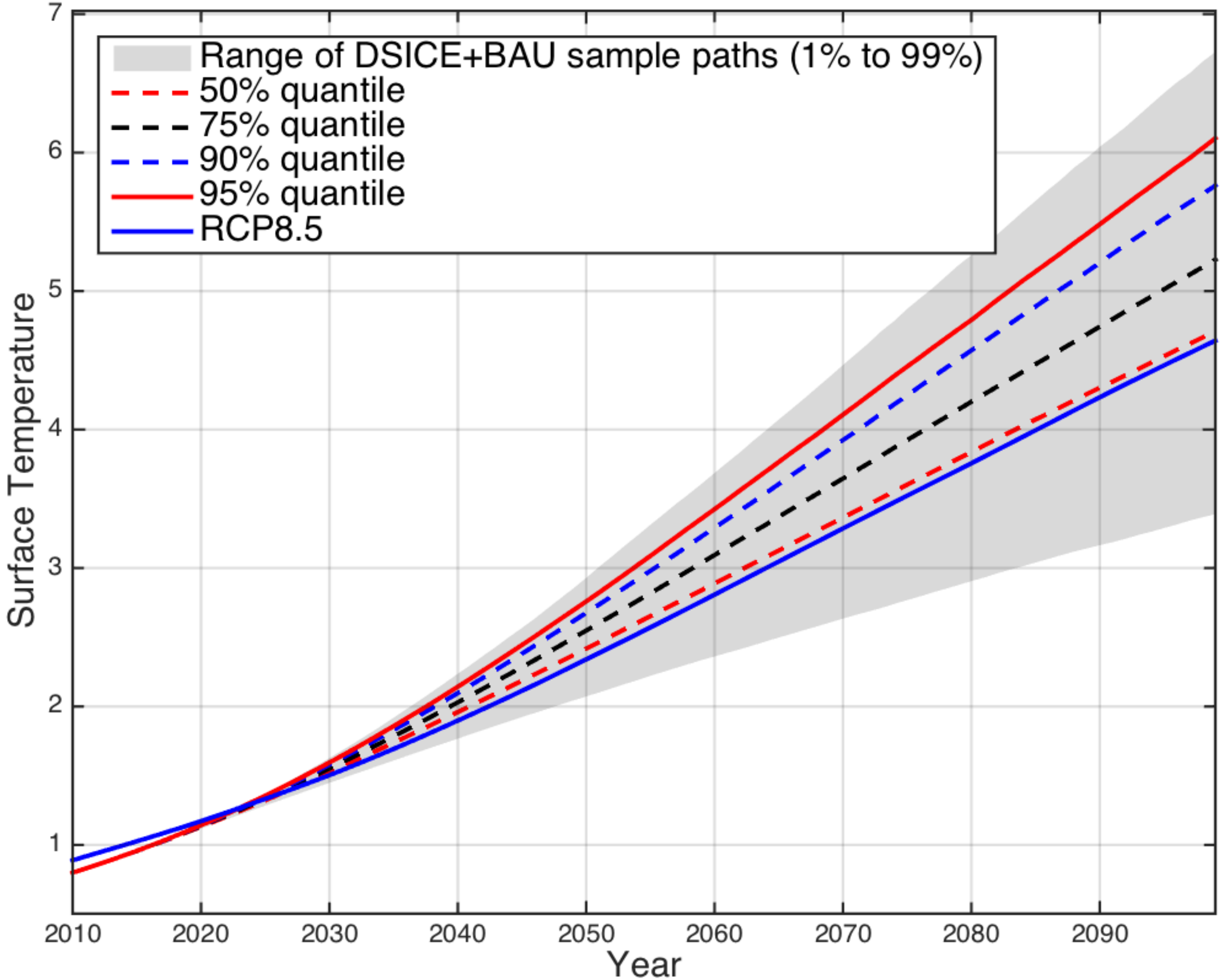
	Parallelization	No Parallelization
Number of cores	84K	1
Running time	8 hours	77 years

- Speed can be improved by 10-100x

LRR Benchmark: SCC and Carbon Tax



- Optimal Initial carbon tax: \$65 (deterministic model: \$37)



Range of DSICE+BAU sample paths (1% to 99%)
50% quantile
75% quantile
90% quantile
95% quantile
RCP8.5
Surface Temperature
Year
1
2
3
4
5
6
7
2010
2020
2030
2040
2050
2060
2070
2080
2090

Conclusions: IAMs (and economics!) can be far more realistic

DSICE shows that

- It is possible to add both climate and economic risk to climate change policy and impact analyses
- It is possible to add sector and/or international disaggregation
- It is possible to incorporate economic uncertainty into dynamic scoring of tax proposals

Changes are necessary to implement the potential

- Economists need to think about leaving their laptops, Excel, Matlab, EViews, and other second millenium tools
- Economists need to collaborate with computational scientists – like everyone else is!

Main Message and Themes

Everything in economics is multidimensional and uncertain.

We cannot define an "Economics of Everything" model

"Everything should made as simple as possible, *but not simpler*."

We must use computer power to examine a wide range of flexible and plausible models.

Reception in economics

The first Journal of Political Economy editor said "Ken, it's nice to write papers for smart people but that is not what the referees and readers want."

Second editor accepted the paper *soon after* I took my name off (I wanted to make sure that Cai could get the paper published before his tenure case).
(Do a google search for "Heckman Judd war").

My blog showmethemath.org has examples of how bad economics is.

Will soon have many more!

Types of Computational Problems inEconomics

Constrained optimization problem

Dynamic programming decomposes enormous optimization problems into many connected small ones.

Use Knitro, SNOPT, NPSOL, Filter, ...

System of polynomials

The total system of polynomials in G could be enormous BUT solution is possible if we can apply the Wavefront abstraction.

Bertini will produce all solutions

Singular will produce Grobner basis of system

Nonlinear equations or a nonlinear complementarity problem

Use PATH (Munson and Ferris)

My goals

- Develop useful software easily accessible for economists
  - “Accessibility” is a challenge
  - E.g., most economists do not know what ssh means.
- Deploy on OSG
- Find collaborators who can help me exploit computational resources
  - Stanford
    - Takes out my trash, processes my paycheck and health insurance
    - *Provides no research support*
  - Collaborators will not be helping Stanford

Optimal Taxation

We developed the first method that can solve "degenerate optimization problems" (i.e., solution has indeterminate shadow prices for constraints).

There are many degenerate problems in engineering.

NVidia is implementing NCL on their newest GPUs

Stabilized Optimization Via an NCL Algorithm

Check for updates

Ding Ma, Kenneth L. Judd, Dominique Orban and Michael A. Saunders

 173
M. Al-Baali et al. (eds.), *Numerical Analysis and Optimization*,
Springer Proceedings in Mathematics & Statistics 235,
https://doi.org/10.1007/978-3-319-90026-1_8

Maximum Likelihood Estimation

- Standard practice
  - Use primitive methods to find estimate
  - Compute standard errors for each parameter
- Our project
  - Take code that computes a likelihood function
  - Create a surrogate (approximation)
    - Use asynchronous adaptive methods
    - Sparse grids, adaptive least squares
    - NOT neural networks
  - Using surrogate to
    - Find all local max
    - Compute confidence *sets*

Game theory

- Games often have multiple equilibria
  - Multiplicities dues to diverse expectations of players -- not something seen in physics
  - Set of equilibria can be represented by a collection of convex sets
- We have an algorithm for computing that collection of convex sets

DSICE

- Was designed to be a framework
  - Ready to add multiple sectors and countries (using PATH)
  - Ready to add a methane equation

Opportunities in Economics

- Economics needs to use modern computational methods
- There is a lot of low-hanging fruit
- HTCondor and OSG can be very helpful for my efforts