

# Modeling for Insight with an Energy Economy Optimization Model

CEDM Seminar  
19 March 2012

Joe DeCarolis  
Assistant Professor

Dept of Civil, Construction, and Environmental Engineering  
NC State University

<http://temoaproject.org>

# Talk outline

Problems with model development and application

Introduction to Temoa

Approach to uncertainty analysis

Simple application of stochastic optimization

# Driving questions

How does the world balance the costs of greenhouse gas mitigation in the near-term versus long-term?

What are the anticipated economic and environmental impacts associated with future environmental policies and energy technology deployments?

How do decision makers craft energy planning strategies that are robust to future uncertainties?

How do decision makers incorporate broader environmental sustainability considerations — beyond simply limits to greenhouse gas emissions — into their strategies?

# Energy-economy optimization (EEO) models

Large uncertainties combined with a mix of technical, moral, and philosophical considerations preclude definitive answers to the questions above.

Model-based analysis can deliver crucial insight that informs key decisions.

Energy-economy optimization (EEO) models refer to partial or general equilibrium models that **minimize cost or maximize utility** by, at least in part, optimizing the energy system over multiple decades

- Self-consistent framework for evaluation
- Explore how effects may propagate through a system
- Expansive system boundaries and multi-decadal timescales

**What can we usefully conclude from modeling exercises where uncertainty is rigorously quantified?**

# High Visibility Model-Based Analyses

IPCC Special Report on Emissions Scenarios

<http://www.ipcc.ch/ipccreports/sres/emission/index.htm>

IEA Energy Technology Perspectives

<http://www.iea.org/techno/etp/index.asp>

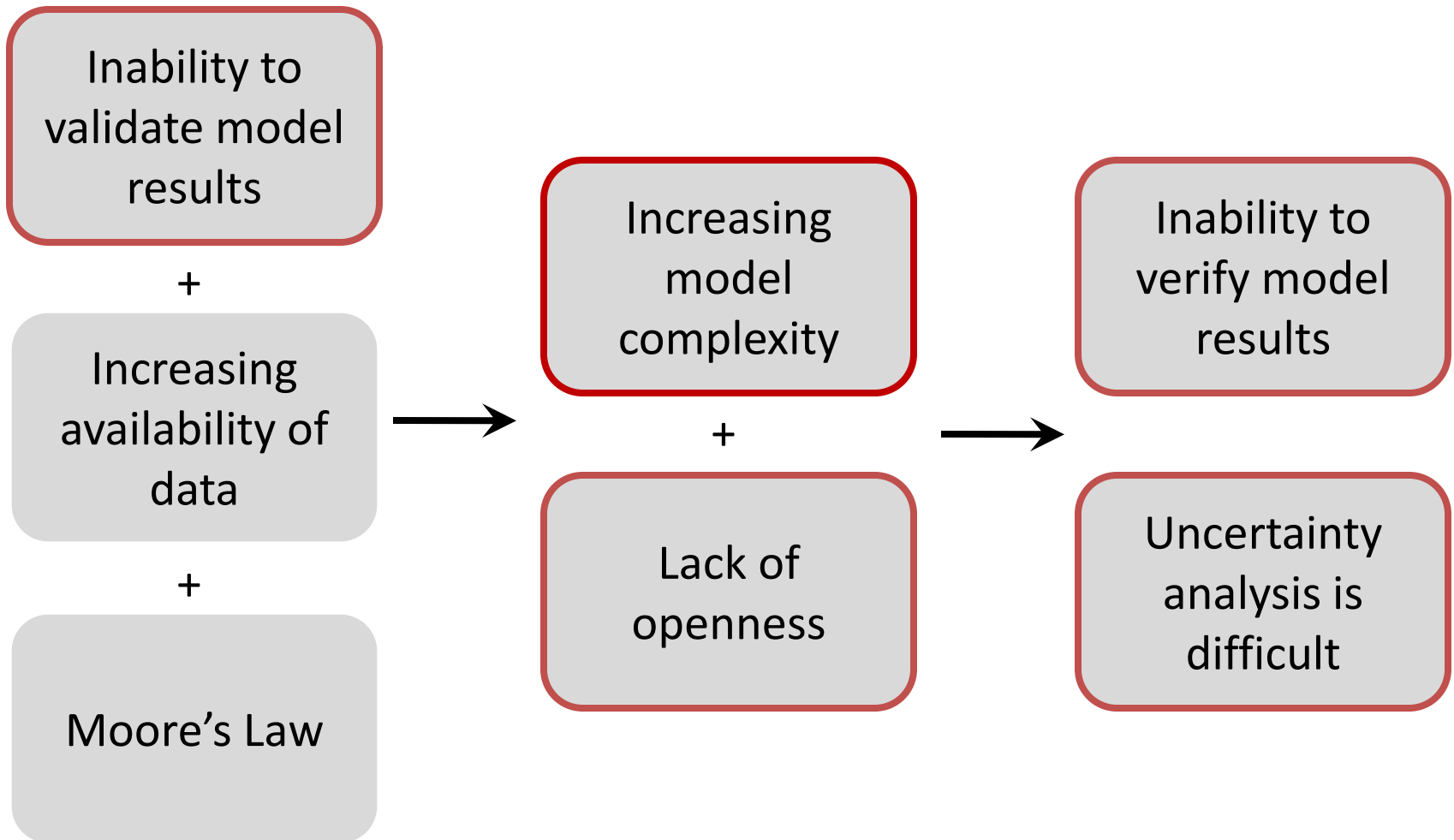
Annual Energy Outlook

<http://www.eia.gov/forecasts/aeo/er/>

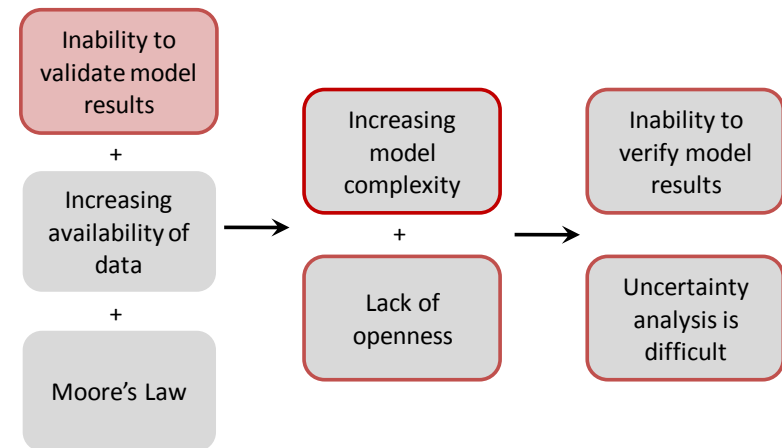
EPA Legislative Analyses

<http://epa.gov/climatechange/economics/economicanalyses.html>

# Problems with the status quo



# Four conditions for validatable models

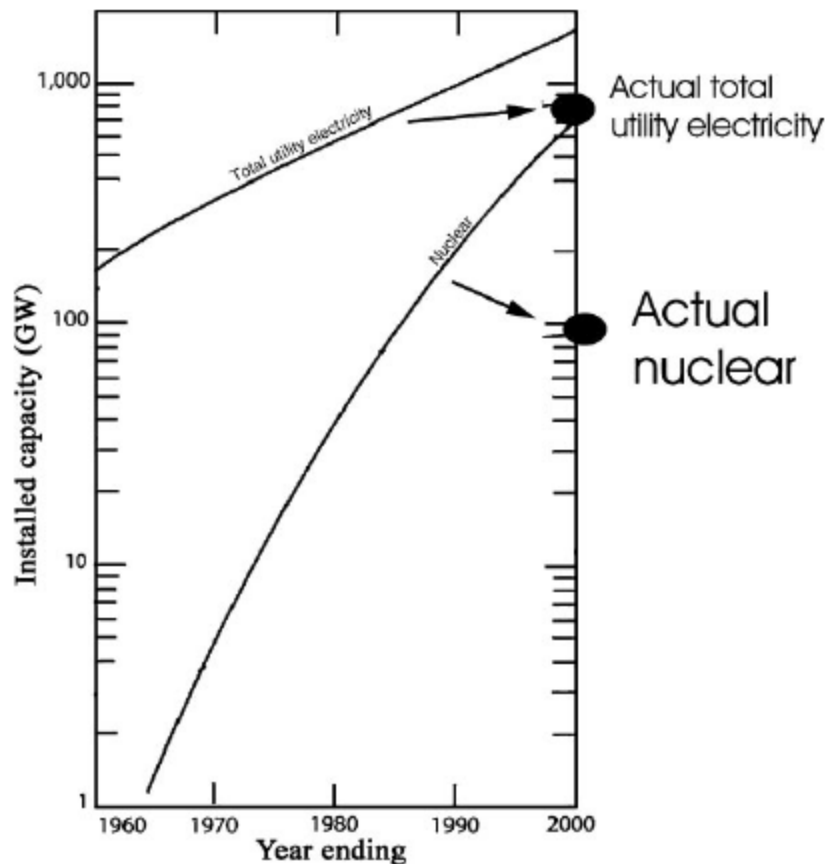


According to Hodges and Dewar (1992) :

- It must be possible to observe and measure the situation being modeled.
- The situation being modeled must exhibit a constancy of structure in time.
- The situation being modeled must exhibit constancy across variations in conditions not specified in the model.
- It must be possible to collect ample data with which to make predictive tests of the model.

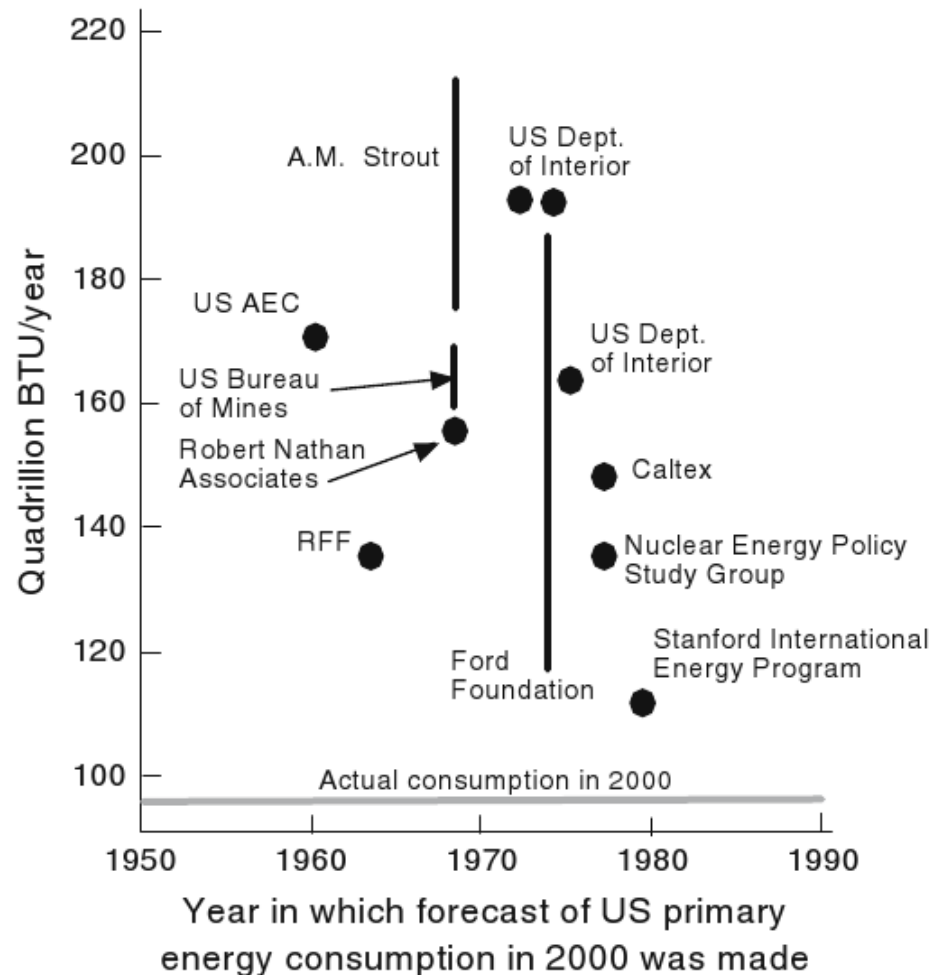
→ Little to guide the modeler and reign in efforts that do not improve model performance

# Past projections are generally dismal



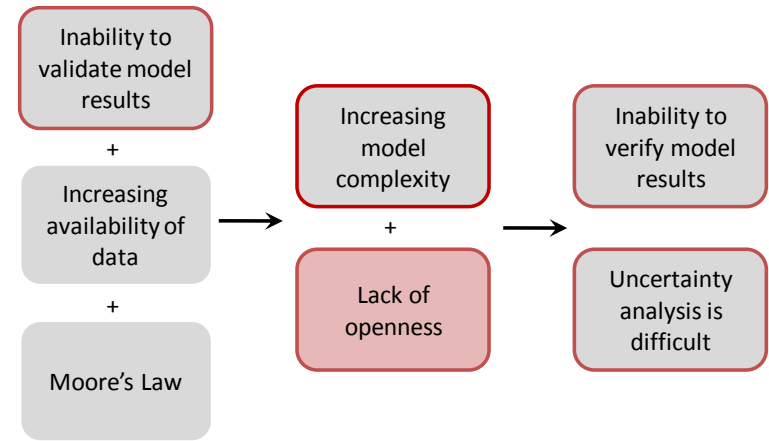
U.S. Atomic Energy Commission  
forecast from 1962

**Source:** Craig et al. (2002). "What Can History Teach Us? A Retrospective Examination of Long-Term Energy Forecasts for the United States." *Ann. Rev. Energy Environ.* 27:83-118.



**Source:** Morgan G, Keith D. (2008). "Improving the way we think about projecting future energy use and emissions of carbon dioxide." *Climatic Change.* 90: 189-215.

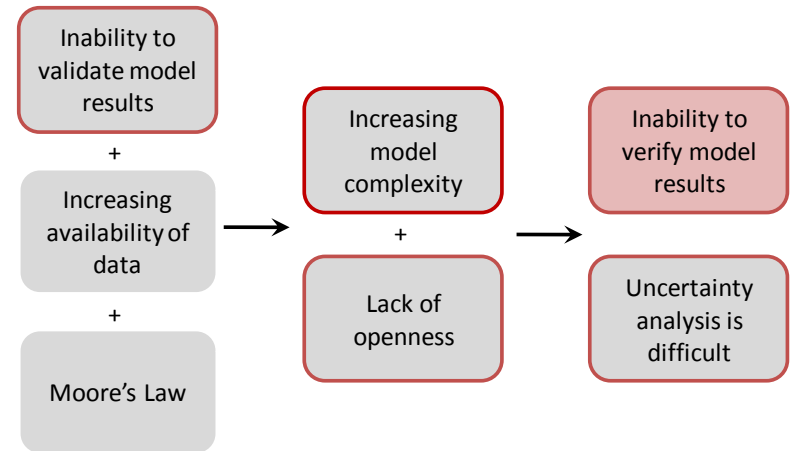
# Lack of openness



Most EEO models and datasets remain closed source. Why?

- protection of intellectual property
- fear of misuse by uninformed end users
- inability to control or limit model analyses
- implicit commitment to provide support to users
- overhead associated with maintenance
- unease about subjecting code and data to public scrutiny

# Inability to verify model results



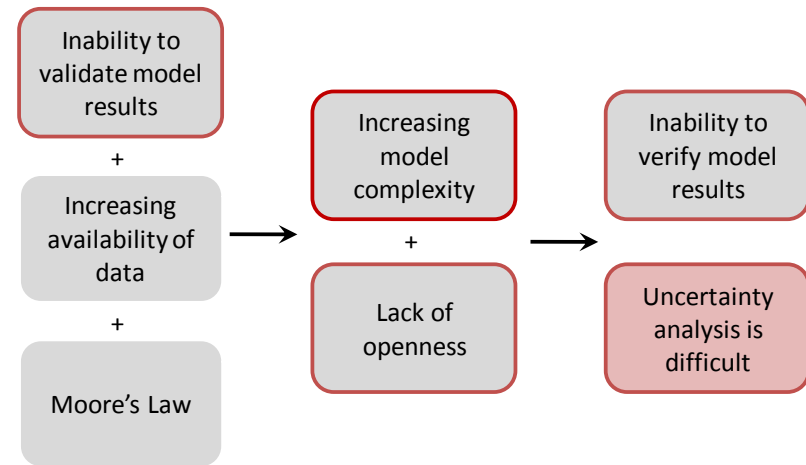
With a couple exceptions,  
**energy-economy models are not open source**

Descriptive detail provided in model documentation and peer-reviewed journals is insufficient to reproduce a specific set of published results

Reproducibility of results is fundamental to science

**Replication and verification of large scientific models can't be achieved without source code and input data**

# Uncertainty analysis is difficult



A common result is false precision

E.g., EPA analysis of S.2191 (Lieberman-Warner), GDP growth predictions to 0.01%!

Large, complex models tuned to look at a few scenarios **by necessity**

Scenario analysis overused

Without subjective probabilities  $p(X|e)$ , scenarios of little value to decision makers

# Problems with scenario analysis

Cognitive heuristics play a role and can lead to misinterpretation of results.

## **Availability heuristic:**

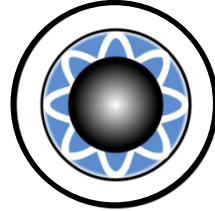
Probabilities of a future event or outcome assessed on the basis of how easily an individual can remember or imagine examples

## **Anchoring and adjustment:**

People start with an initial value or “anchor” and then modify their judgment as they consider factors relevant to the specifics → often insufficient adjustment

→ A few highly detailed scenarios can create cognitively compelling storylines.

Drawn from: Morgan G, Keith D. Improving the way we think about projecting future energy use and emissions of carbon dioxide. *Climatic Change* 2008; 90; 189-215.



# Temoa

Tools for Energy Model Optimization and Analysis

Temoa also means “to seek something” in the Nahuatl (Aztec) language:

**TÊMOĀ** vt to seek something / buscar algo, o inquirir de algún negocio. This contrasts with TEMŌHUA, the nonactive form of TEMŌ ‘to descend.’

Taken from: *An analytical dictionary of Nahuatl*  
by Frances E. Karttunen

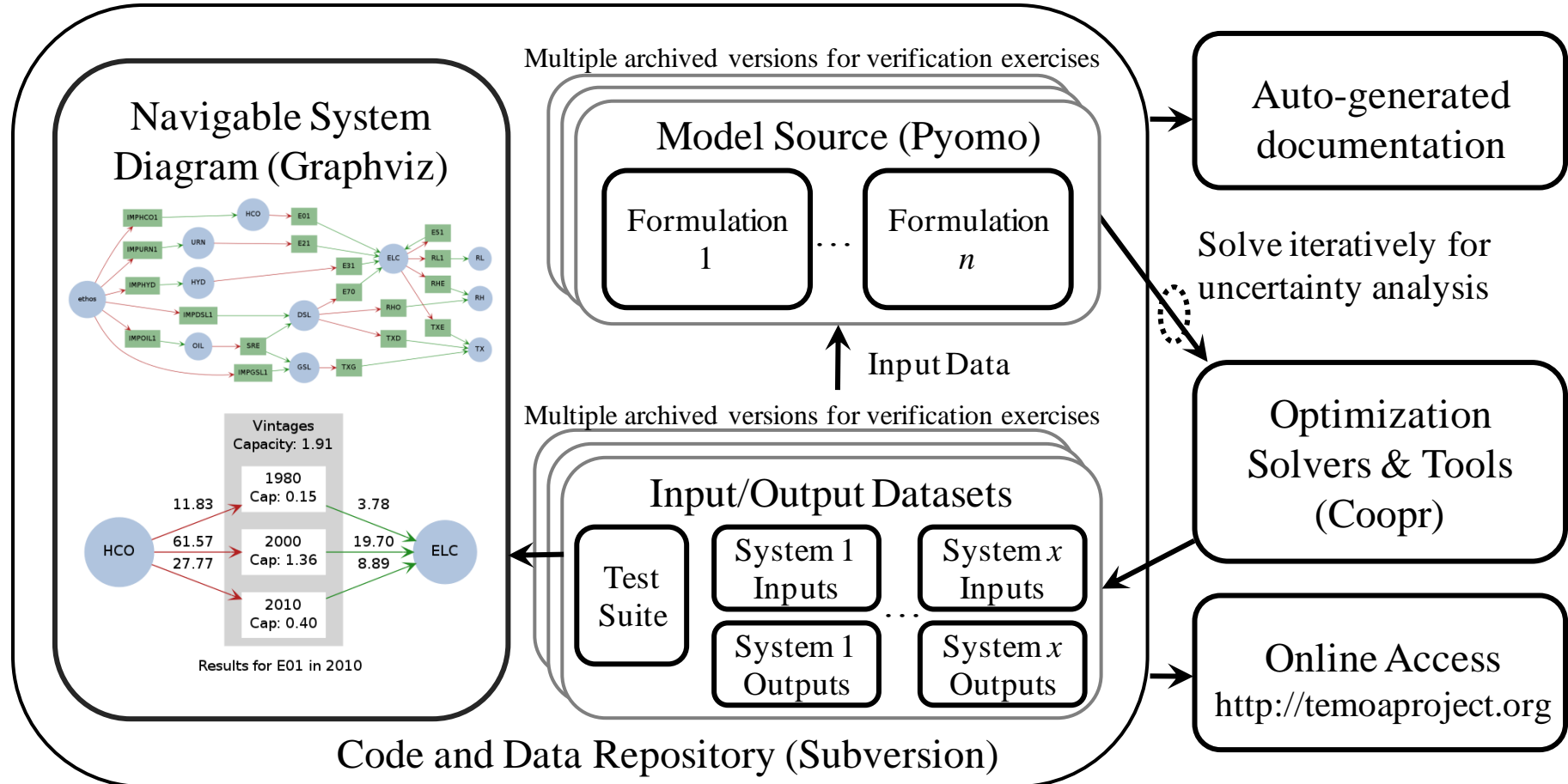
# Temoa goals and approach

**Goal:** Create an open source, technology explicit EEO model

## **Our Approach:**

- Public accessible source code **and data**
- No commercial software dependencies
- Data and code stored in a web accessible electronic repository
- A version control system
- Programming environment with links to linear, mixed integer, and non-linear solvers
- Built-in capability for sensitivity and uncertainty analysis
- Utilize multi-core and compute cluster environments
- Input and output data managed directly with a relational DB\*

# Framework for Temoa



# Version Control

We are using an open source version control system (Subversion)

Why? Ensure the integrity, sustainability and traceability of changes during the entire software lifecycle.

## Version control enables:

- Multiple developers to work simultaneously on software components; automatic integration of non-conflicting changes
- Display the modifications to model source code
- Create software snapshots (releases) that represent well-tested and clearly defined milestones
- **Public access to code and data snapshots used to produce published analysis → enables third party verification**

You can view our code online: <http://svn.temoaproject.org/trac/browser>

Most current branch: [branches/energysystem-process-Coopr3](#)

Works on all major (Unix, Windows, MacOS) platforms

# Programming environment

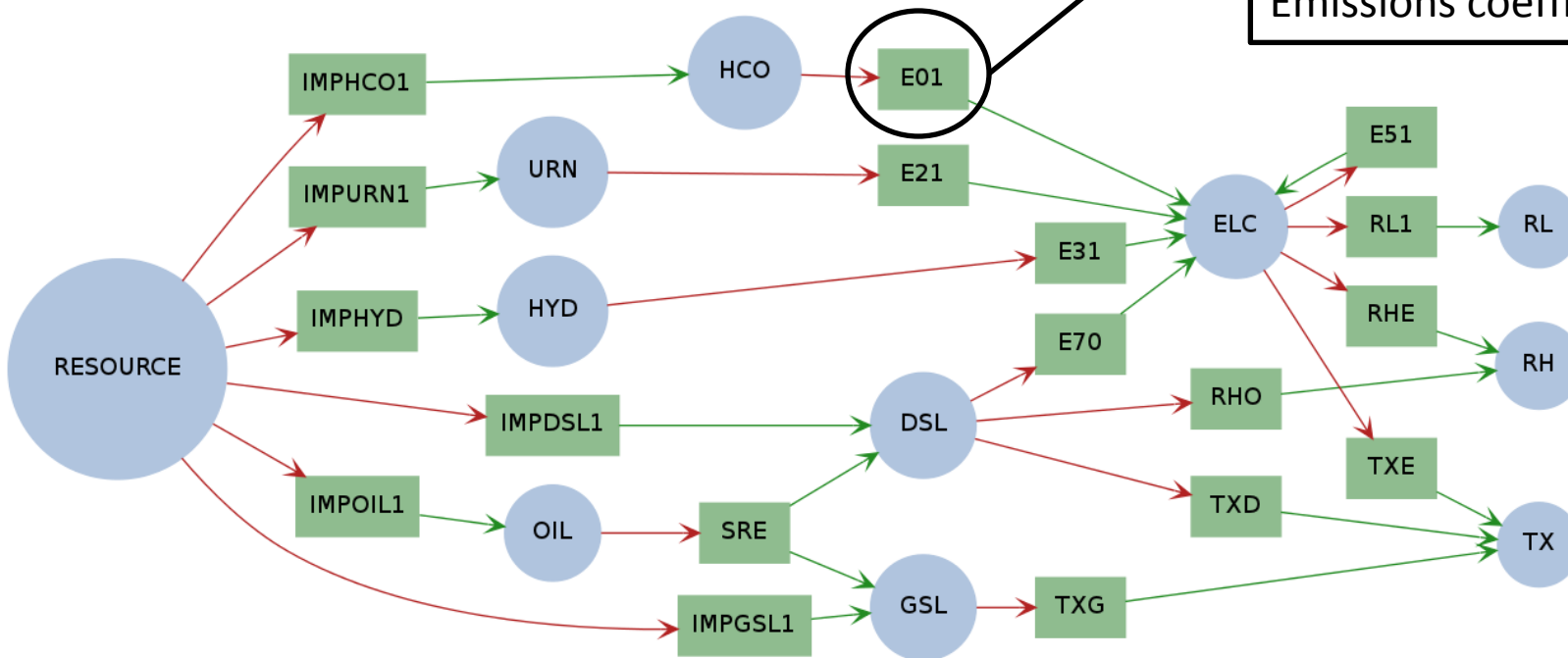
A Common Optimization Python Repository (**Coopr**) is a collection of Python optimization-related packages that supports a diverse set of optimization capabilities for formulating and analyzing optimization models.

- Algebraic model formulation using Python Optimization Modeling Objects (Pyomo)
- Capability to formulate linear, mixed integer, and non-linear model formulations
- Includes a stochastic programming package
- **Part of a rich Python ecosystem (Numpy, Scipy)**

Developed by the Discrete Math and Complex Systems Department at Sandia National Laboratories: <https://software.sandia.gov/trac/coopr/>

# Technology explicit modeling

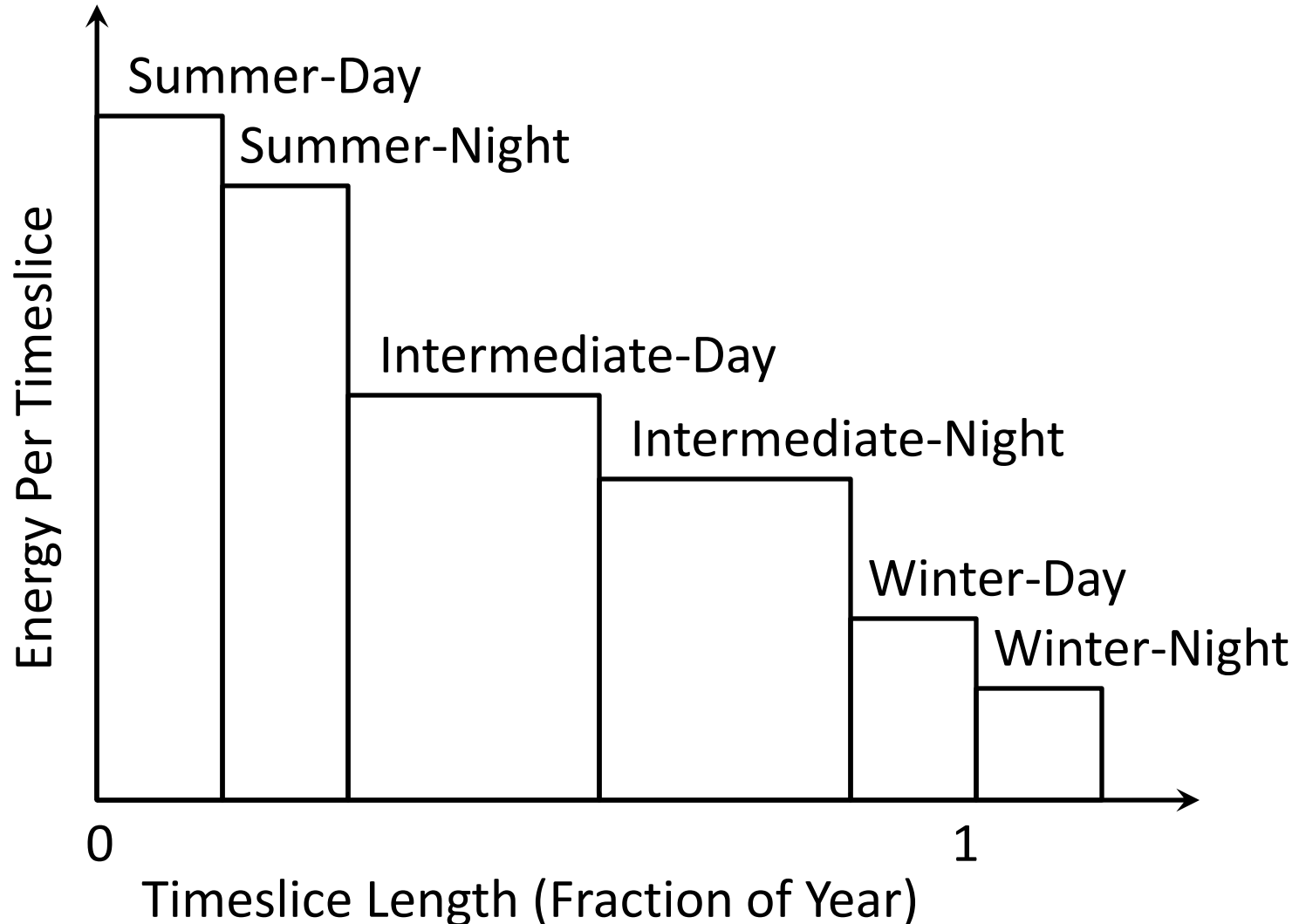
‘Utopia’ (18 technologies included)



**Objective function:** minimize present cost of energy supply

**Decision variables:** activity (PJ) and capacity (PJ/yr) for each technology

# End-Use Demand Specification



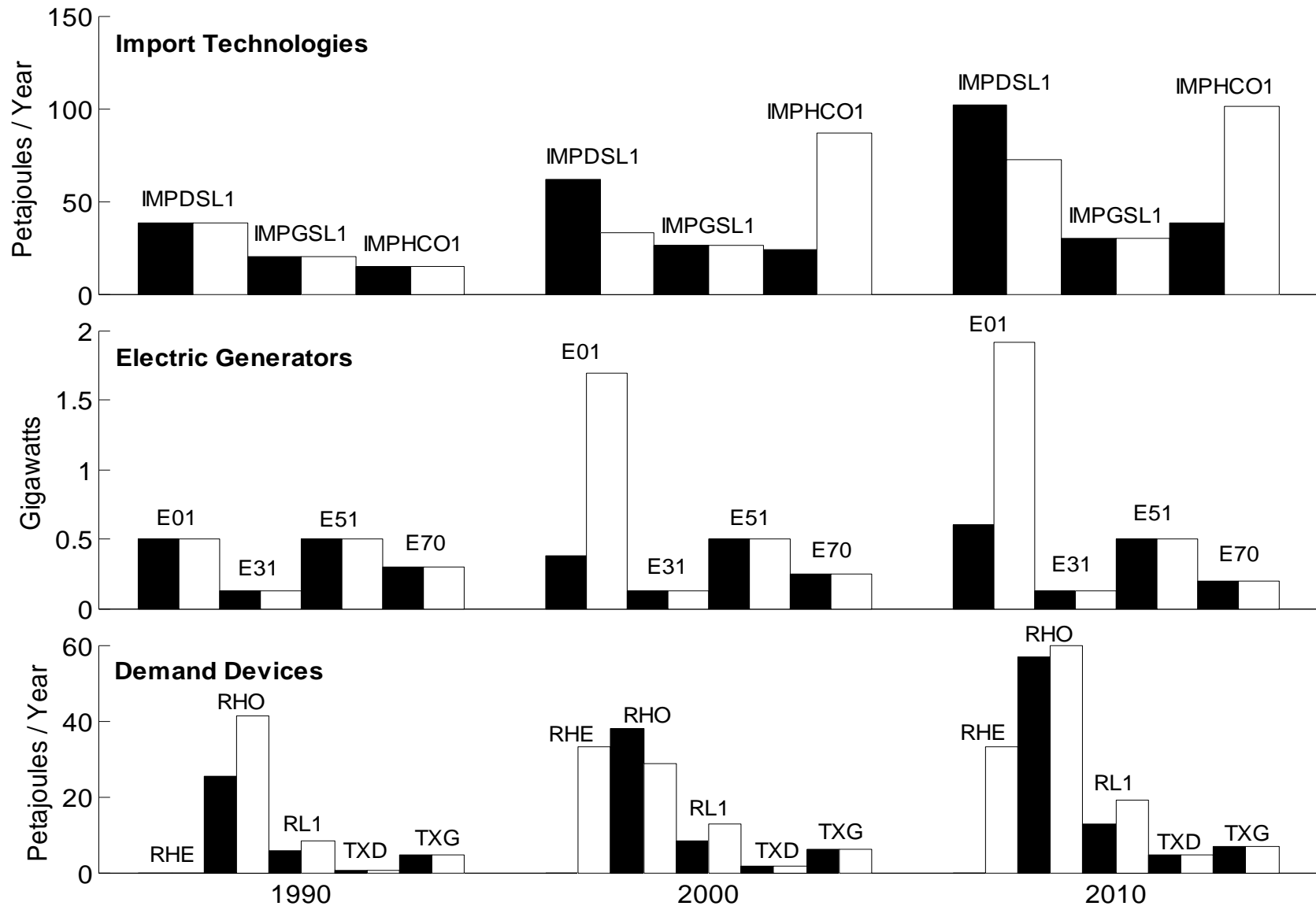
# TEMOA Model Features

**A technology explicit model with perfect foresight**, similar to the TIMES model generator.

- Flexible time slicing by season and time-of-day
- Variable length model time periods
- Technology vintaging
- Separate technology loan periods and lifetimes
- Global and technology-specific discount rates
- Capacity determined by commodity flows at the timeslice level

# 'Utopia' verification exercise

MARKAL: Black  
Temoa: White



# Approach to uncertainty analysis

Use the following techniques in series:

Sensitivity analysis and Monte Carlo simulation

→ Determine key sensitivities

Multi-stage stochastic optimization

→ Develop a hedging strategy

Explore near-optimal, feasible region

(Modeling-to-Generate-Alternatives)

→ Test robustness of hedging strategy

# Stochastic Optimization

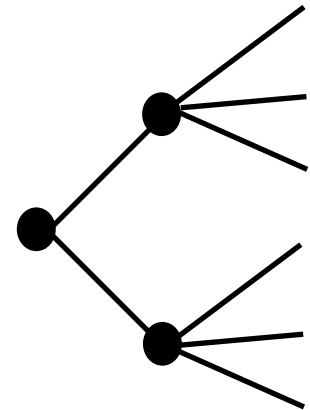
Decision-makers need to make choices before uncertainty is resolved → requires an “act then learn” approach

Need to make short-term choices that hedge against future risk

→ Sequential decision-making process that allows recourse

Stochastic optimization

- Build a scenario tree
- Assign probabilities to future outcomes
- Optimize over all possibilities



# Simple example of stochastic optimization

Suppose we have two technologies, A and B. Let  $x$  and  $y$  represent the installed capacity in Stages 1 and 2, respectively.

Stage 1 Decision Variables:

$x_A, x_B$

Stage 2 Decision Variables:

$y_{A,s_1}, y_{B,s_1}$

Scenario 1:  $s_1$

$y_{A,s_2}, y_{B,s_2}$

Scenario 2:  $s_2$

Minimize:  $c^T x + \sum_{s=1}^N p_s \cdot d_s^T \cdot y_s$

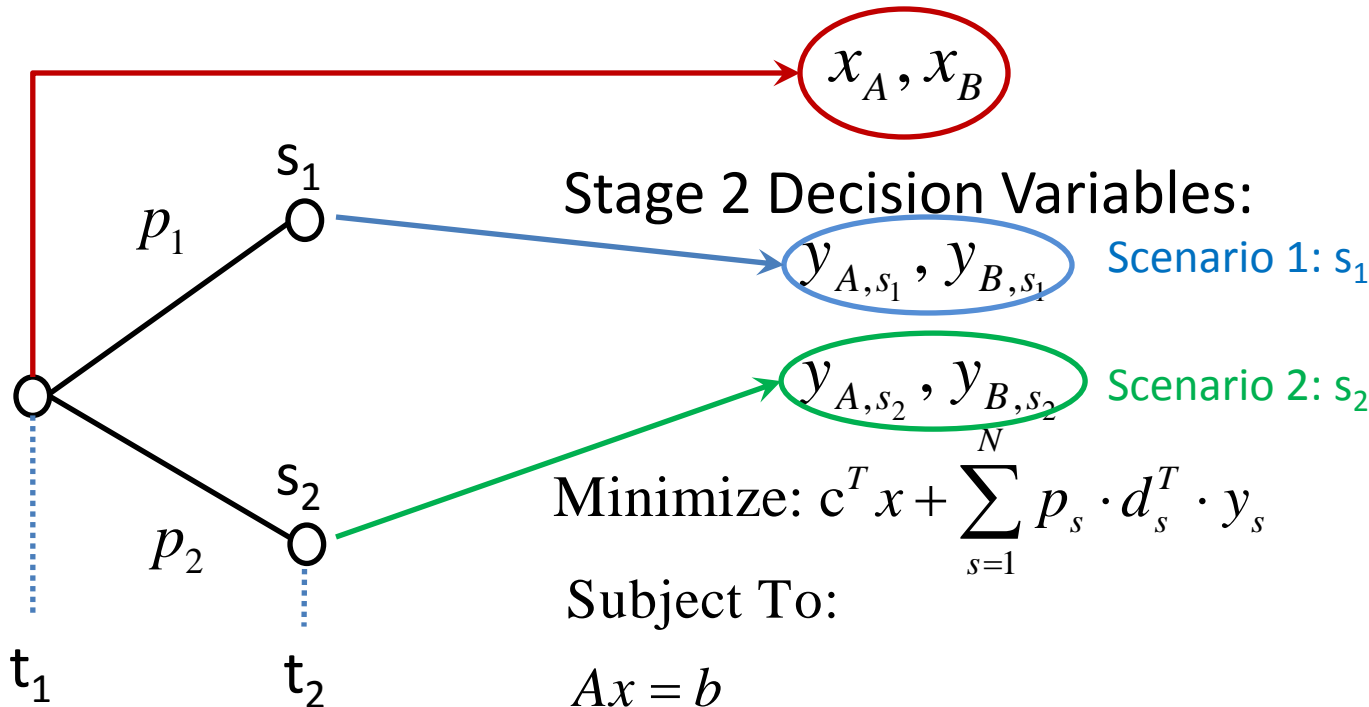
Subject To:

$$Ax = b$$

$$T_s x + W_s y_s = h_s \quad \text{for } s = 1, \dots, N$$

$$x \geq 0$$

$$y_s \geq 0 \quad \text{for } s = 1, \dots, N$$



# Stochastic optimization with PySP

Python-based Stochastic Programming (PySP) is part of the Coop package.

To perform stochastic optimization, specify a Pyomo reference model and a scenario tree

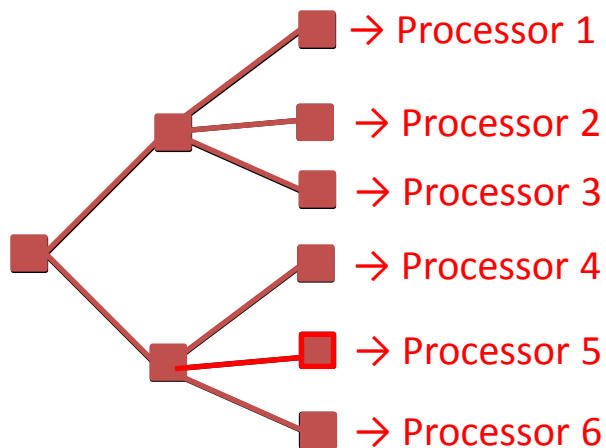
PySP offers two options:

1. **runef**: builds and solves the extensive form of the model.  
“Curse of dimensionality” → memory problems
2. **runph**: builds and solves using a scenario-based decomposition solver (i.e., “Progressive Hedging) based on Rockafellar and Wets (1991).  
Can be implemented in a compute cluster environment; more complex scenario trees possible.

R.T. Rockafellar and R. J-B. Wets. Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of Operations Research*, pages 119–147, 1991.

# Progressive hedging (runph)

- Decomposes a stochastic program by scenarios (i.e., pathways through the event tree) instead of time stages.
- Calculates scenario-specific solutions and proceeds in an iterative manner by updating the scenario-specific solutions for a modified objective.
- Combines them to form a unified solution, and repeats the process until convergence is reached .
- The scenario-specific solutions required at each iteration can be evaluated in parallel on a compute cluster.



## NCSU Cluster “Cygnus”:

- 11 nodes, each with 2 AMD quad-core Opteron processors (2.0 GHz with 512 KB Cache/core)
- 1.8 TB of storage
- 176 GB memory
- OpenSuse 10.3 (Linux)
- FLOPS = 704 Gigaflops
- 1 GigE interconnect

# Stochastic application of 'Utopia'

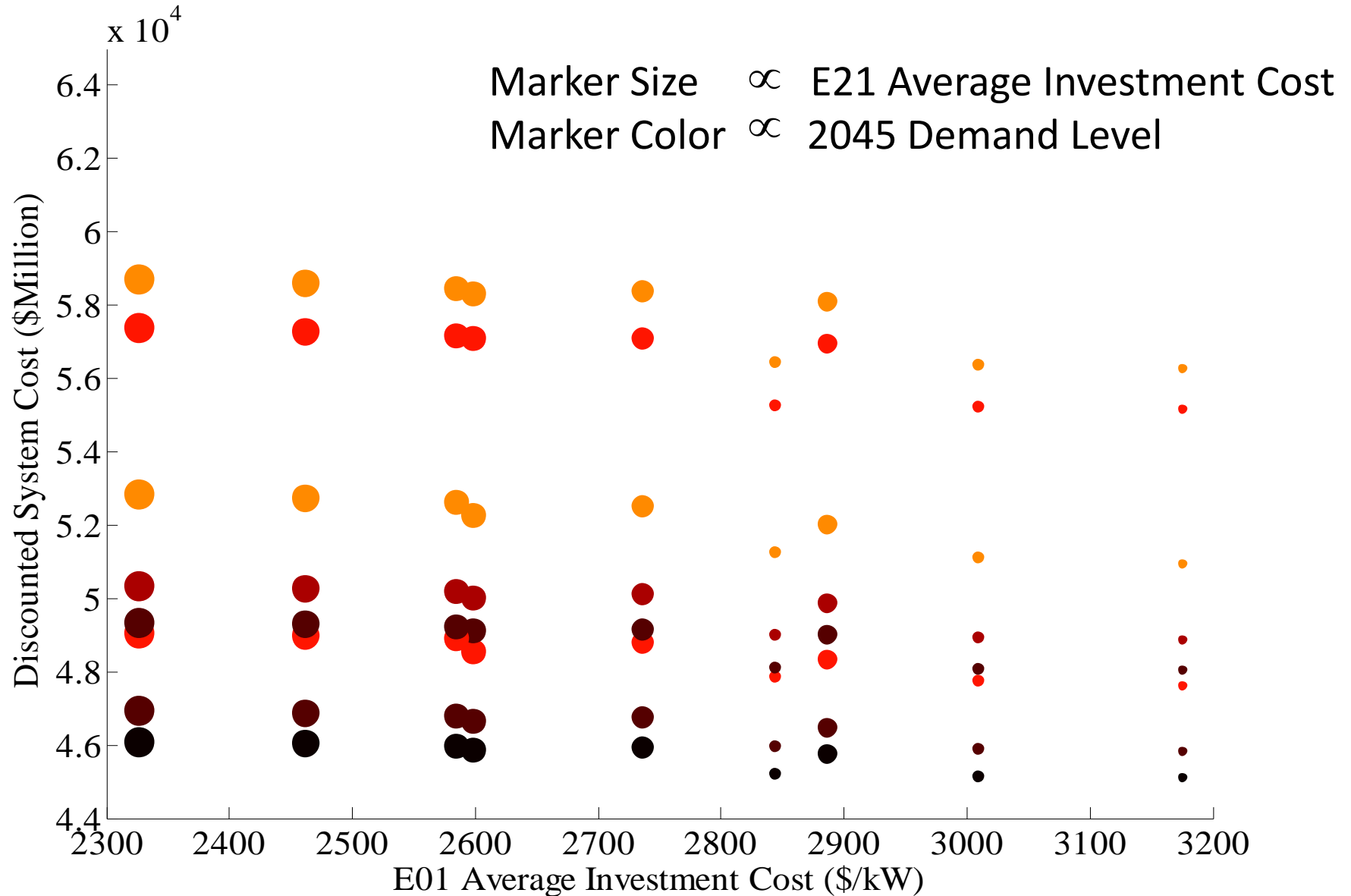
A proof-of-concept application

9 branches per node / 2 uncertain time stages → 81 scenarios

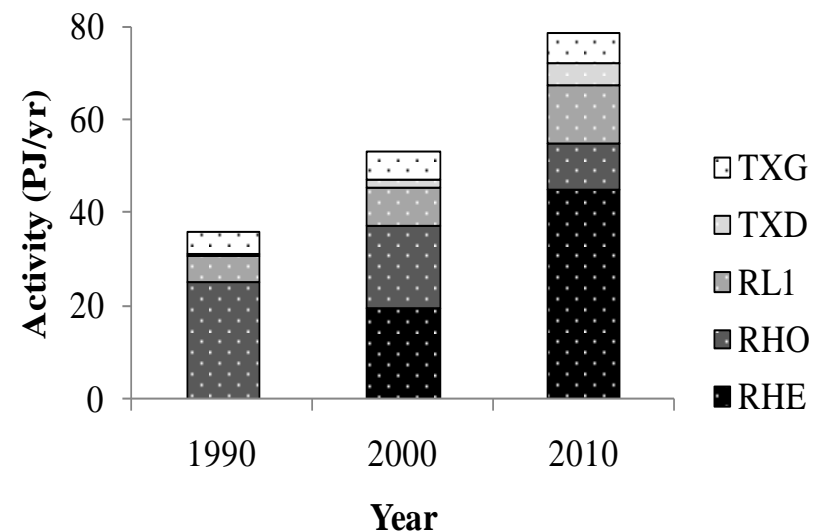
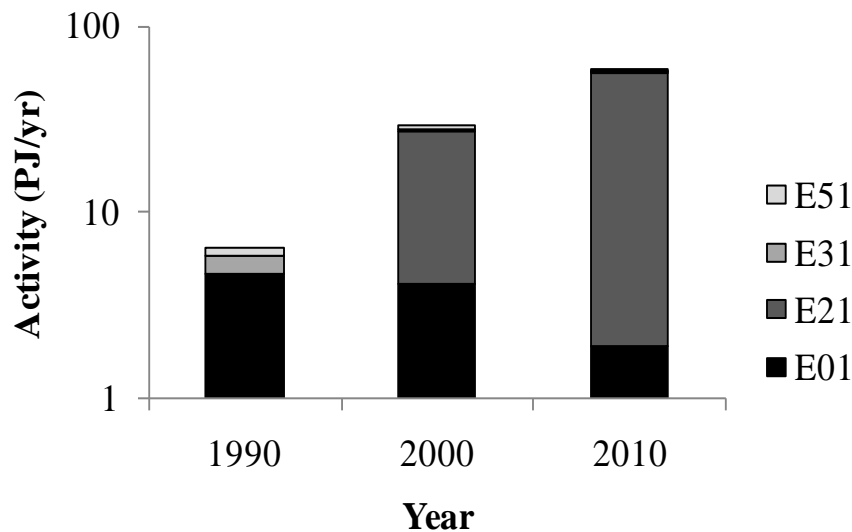
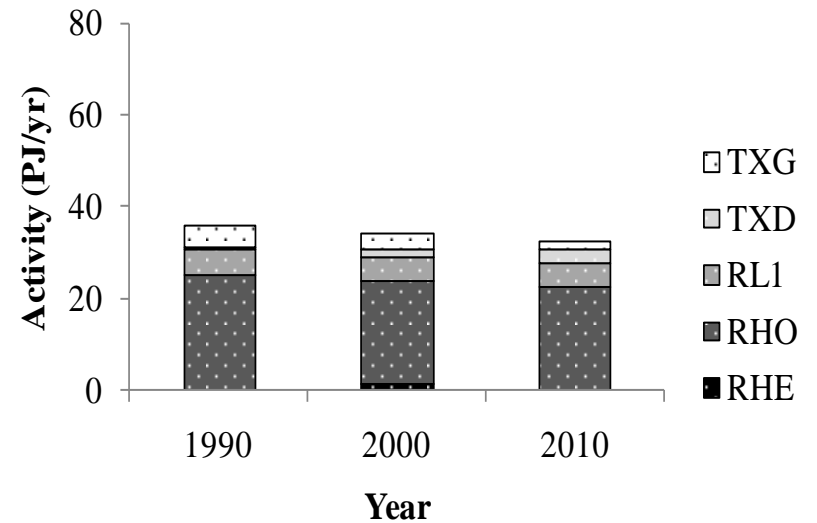
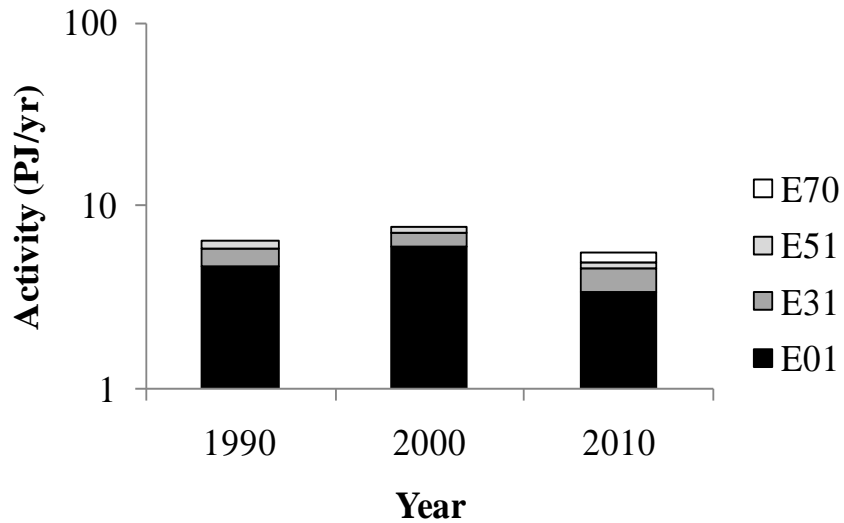
Decadal growth rates used in stochastic utopia:

| Scenario              | E01 | E21   | RH   | RL   | TX   |
|-----------------------|-----|-------|------|------|------|
| Cost (L) / Demand (H) | 1.2 | -0.80 | 0.48 | 0.48 | 0.48 |
| Cost (L) / Demand (M) | 1.2 | -0.80 | 0.11 | 0.11 | 0.11 |
| Cost (L) / Demand (L) | 1.2 | -0.80 | 0.05 | 0.05 | 0.05 |
| Cost (M) / Demand (H) | 1.0 | -0.30 | 0.48 | 0.48 | 0.48 |
| Cost (M) / Demand (M) | 1.0 | -0.30 | 0.11 | 0.11 | 0.11 |
| Cost (M) / Demand (L) | 1.0 | -0.30 | 0.05 | 0.05 | 0.05 |
| Cost (H) / Demand (H) | 0.8 | 0.20  | 0.48 | 0.48 | 0.48 |
| Cost (H) / Demand (M) | 0.8 | 0.20  | 0.11 | 0.11 | 0.11 |
| Cost (H) / Demand (L) | 0.8 | 0.20  | 0.05 | 0.05 | 0.05 |

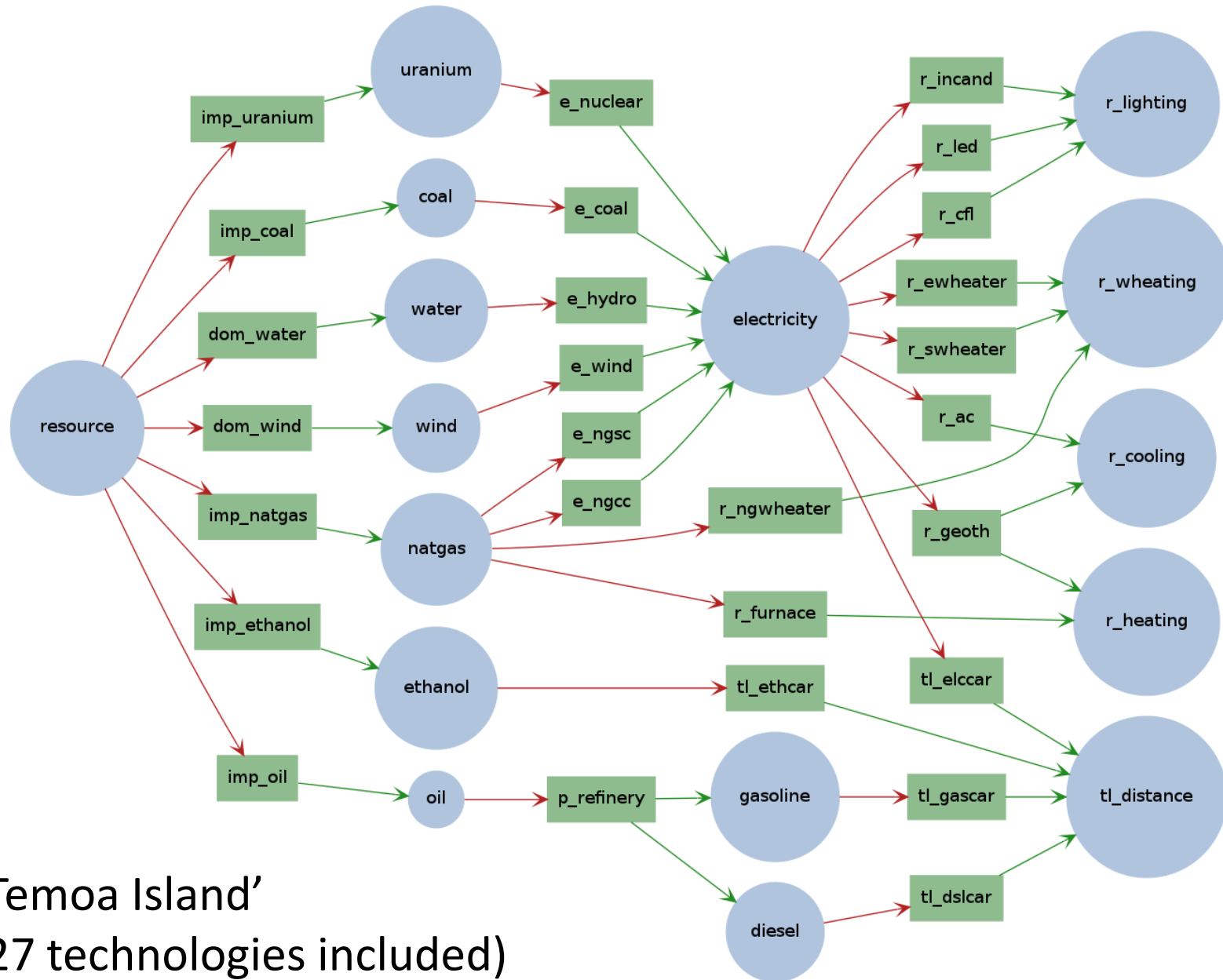
# Results from stochastic 'utopia'



# 'Utopia' scenario-specific results



# A slightly more complicated stochastic application

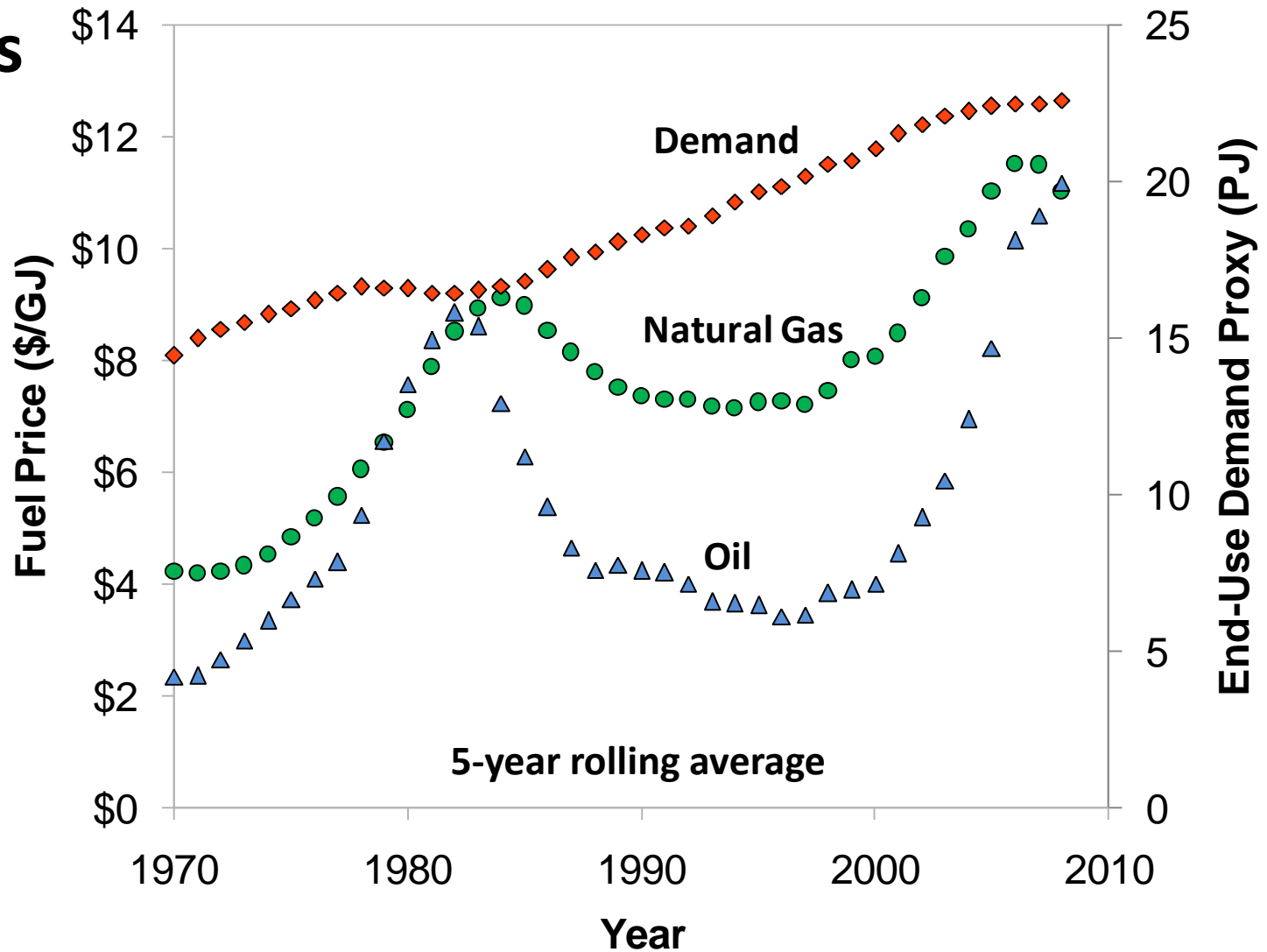


‘Temoa Island’  
(27 technologies included)

# Stochastics

3 stochastic parameters:

- End-use demand
- Natural gas price
- Crude oil price



Historical data drawn from the U.S. EIA *Annual Energy Review 2011*

Source: <http://www.eia.gov/totalenergy/data/annual/>

(Demand proxy is U.S. total residential energy demand)

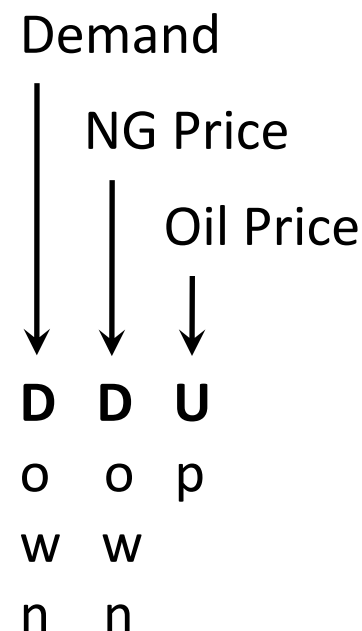
# A Simple Markov Chain

- Using the rolling average data, calculate the growth rate associated with each stochastic parameter from one period to the next
- Calculate conditional probabilities and associated growth rates based on examining results from each pair of successive periods

**Probabilities:** **TO**

| FROM | DDD | DDU | DUD | DUU | UDD | UDU  | UUD | UUU |
|------|-----|-----|-----|-----|-----|------|-----|-----|
| DDD  |     |     |     |     |     |      |     |     |
| DDU  |     |     |     |     |     | 100% |     |     |
| DUD  |     |     |     |     |     |      |     |     |
| DUU  |     | 67% |     |     |     |      |     | 33% |
| UDD  |     |     |     |     | 78% | 11%  | 11% |     |
| UDU  |     |     |     |     | 33% |      |     | 67% |
| UUD  |     |     |     |     | 25% | 25%  | 50% |     |
| UUU  |     | 6%  |     | 6%  |     |      | 6%  | 82% |

**Example:**

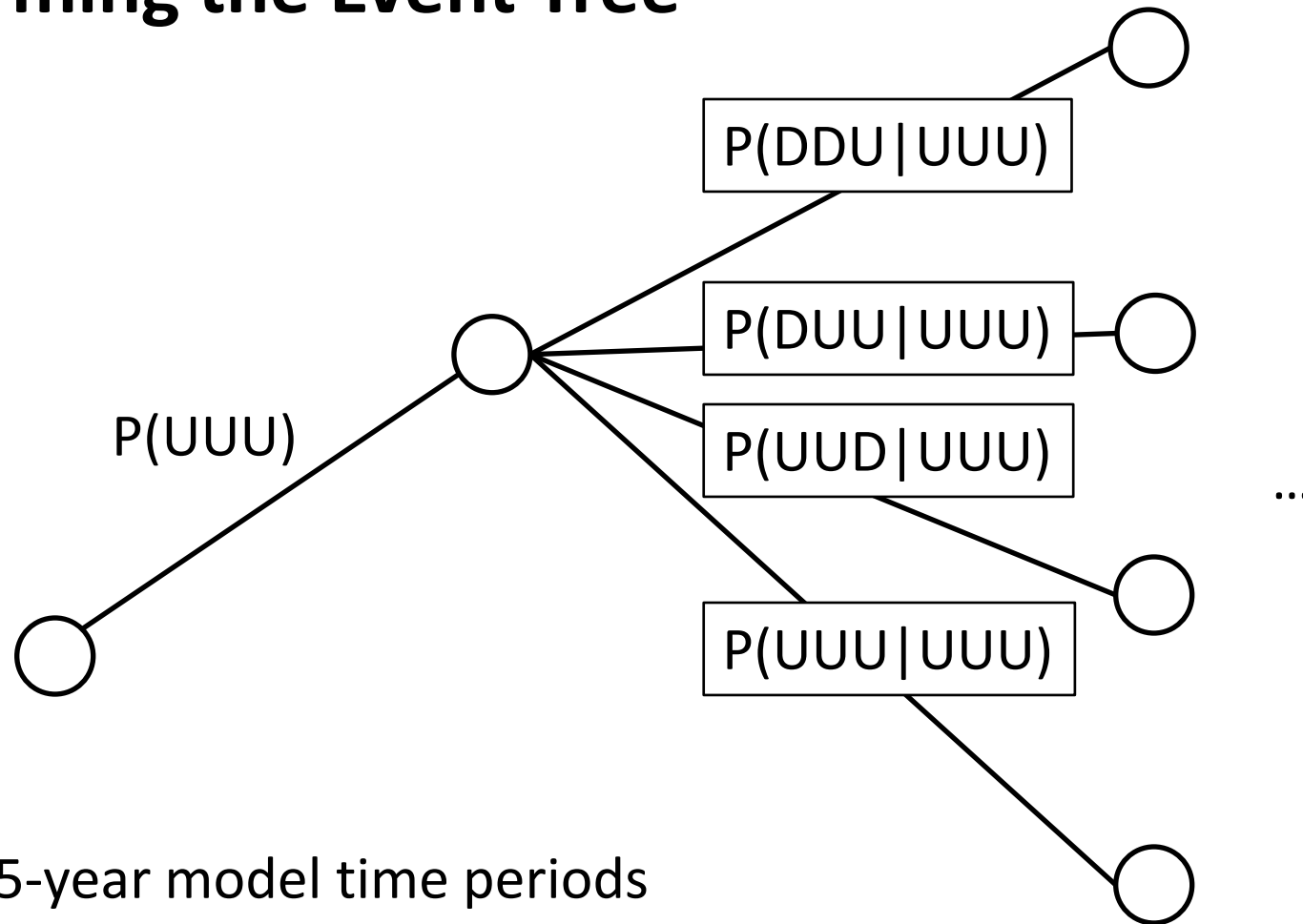


Historical data drawn from the U.S. EIA *Annual Energy Review 2011*

Source: <http://www.eia.gov/totalenergy/data/annual/>

(Demand proxy is U.S. total residential energy demand)

# Forming the Event Tree



5-year model time periods

Non-anticipative stages: 2010 - 2020

Uncertainty is resolved: 2025 - 2045

Resulted in a total of 309 scenarios

# Solve statistics

Solved the extensive form using `runef`

## **Raw LP:**

Variables: 3,907,626

Constraints: 5,384,195

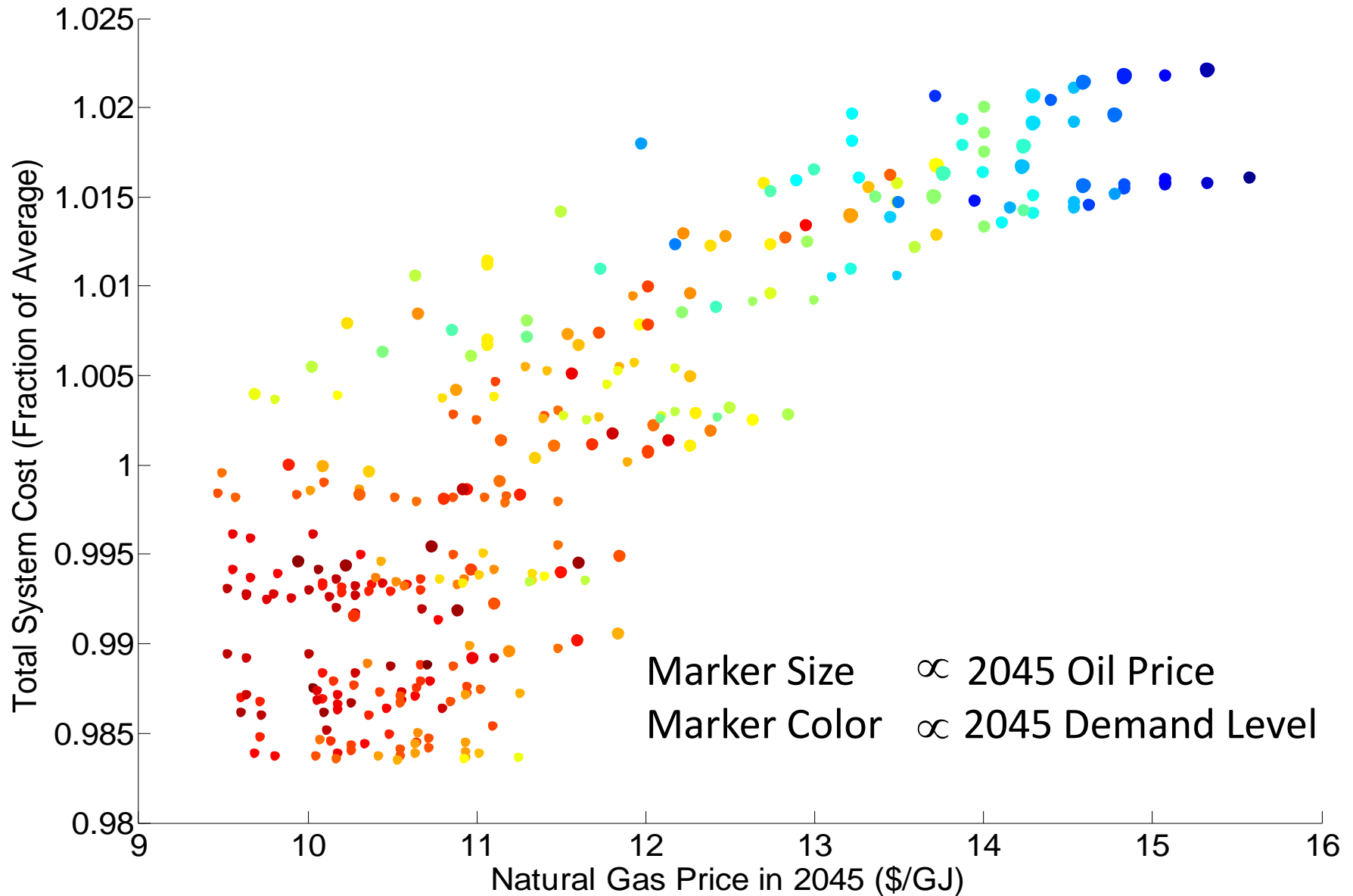
## **Presolved LP:**

597,421 constraints

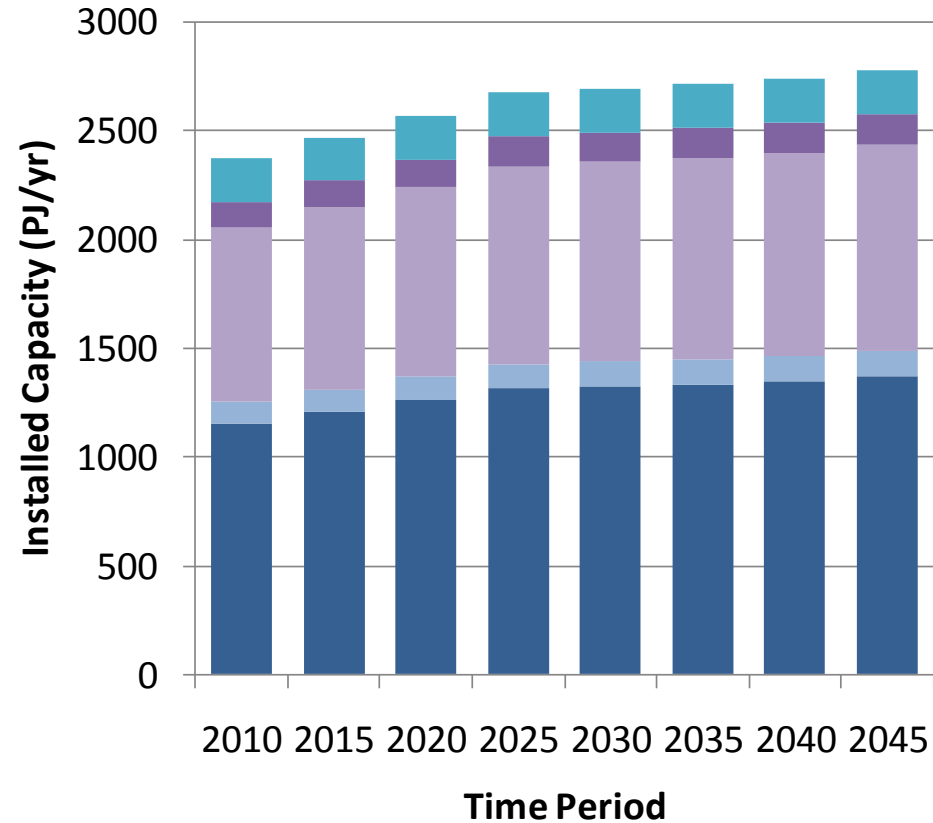
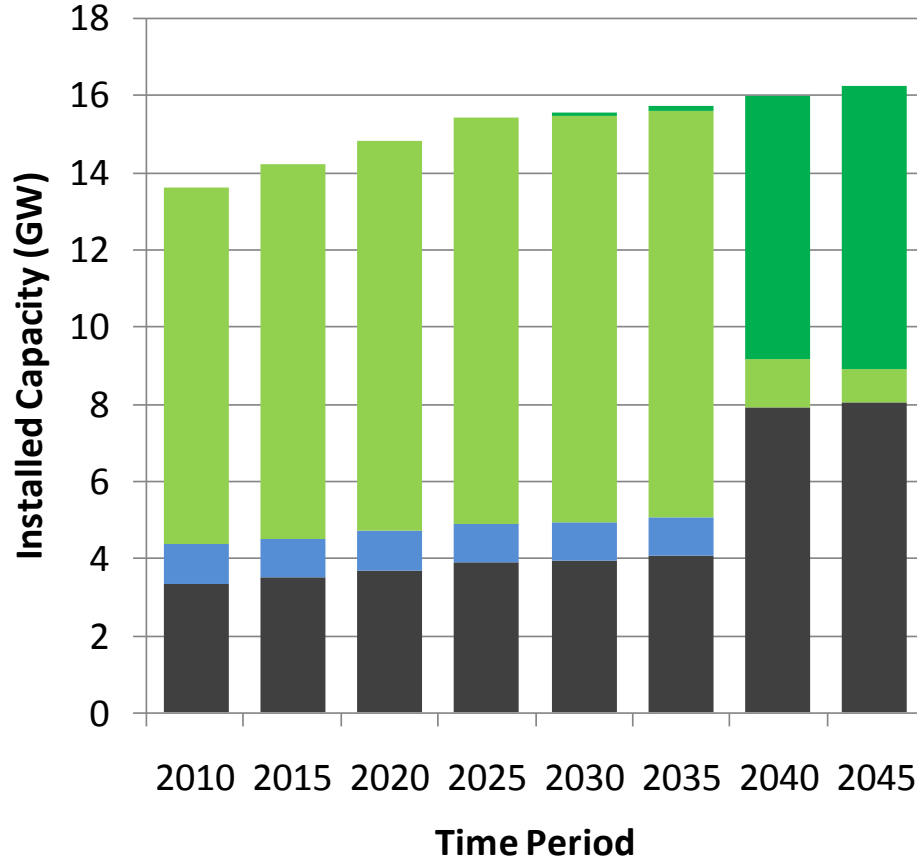
196,274 variables

It took CPLEX 133,772.65 seconds (38 hours) to solve.

# Total System Cost versus Natural Gas Price



# Capacity Results: Lowest Cost Scenario



- Electric AC
- CFL Lighting
- NG Furnace
- Solar Water Heater
- Gasoline Car (bvmt/yr)

# Modeling to Generate Alternatives

Still haven't dealt with structural uncertainty in the model

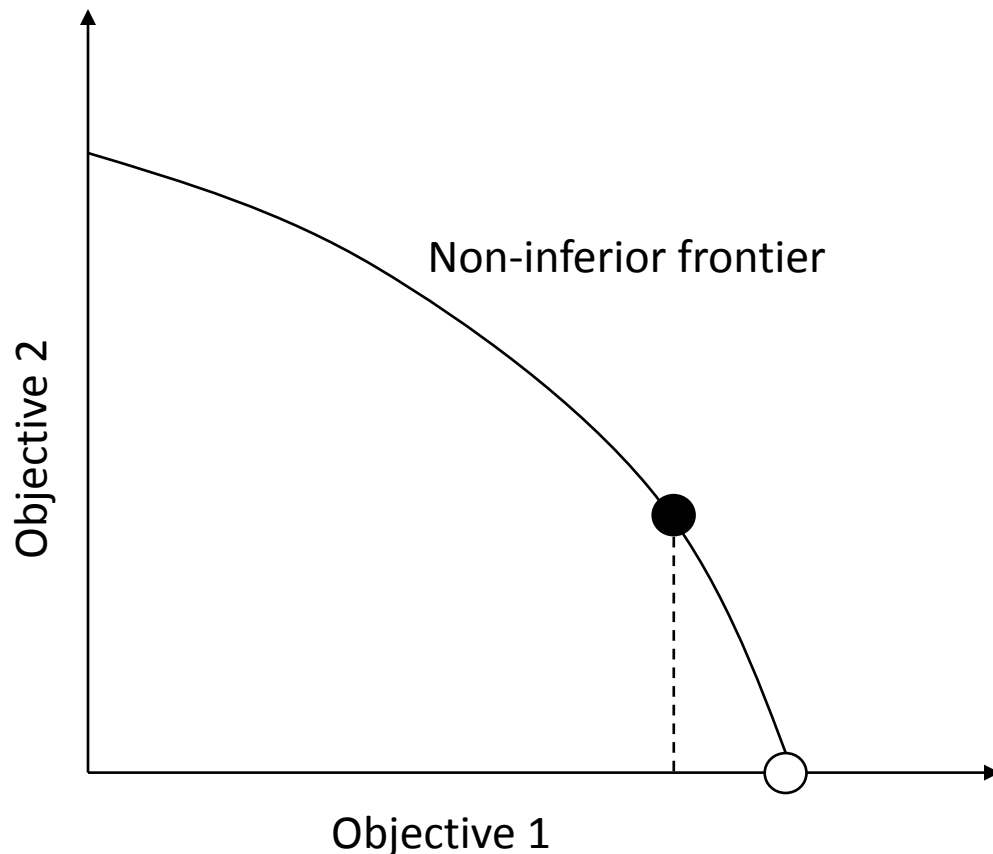
Need a method to explore an optimization model's feasible region → “Modeling to Generate Alternatives”<sup>†</sup>

MGA generates alternative solutions that are **maximally different in decision space** but perform well with respect to modeled objectives

The resultant MGA solutions provide modelers and decision-makers with a set of alternatives for further evaluation

<sup>†</sup>Brill (1979), Brill et al. (1982), Brill et al. (1990)

# How Optimal is the “Optimal” Solution?



Consider an optimization model that only includes **Objective 1** and leaves **Objective 2** unmodeled. The true optimum is within the feasible, suboptimal region of the model's solution space.

Viable alternative solutions exist within the model's feasible region.

# Hop-Skip-Jump (HSJ) MGA

Brill et al. (1982)

Steps:

1. Obtain an initial optimal solution by any method
2. Add a user-specified amount of slack to the value of the objective function
3. Encode the adjusted objection function value as an additional upper bound constraint
4. Formulate a new objective function that minimizes the decision variables that appeared in the previous solutions
5. Iterate the re-formulated optimization
6. Terminate the MGA procedure when no significant changes to decision variables are observed in the solutions

# HSJ MGA

## Mathematical formulation

$$\begin{aligned} \min \quad & p = \sum_{k \in K} x_k \\ \text{s.t.} \quad & f_j(\vec{x}) \leq T_j \quad \forall j \\ & \vec{x} \in X \end{aligned}$$

where:

$K$  represents the set of indices of decision variables with nonzero values in the previous solutions

$f_j(\vec{x})$  is the  $j^{\text{th}}$  objective function

$T_j$  is the target specified for the  $j^{\text{th}}$  modeled objective

$X$  is the set of feasible solution vectors

# Conclusions

Most EEO models and model-based analyses are opaque to external parties

The TEMOA project represents a new, transparent modeling framework designed for rigorous uncertainty analysis

- Archival copies of source code and data publicly available for replication
- Uncertainty analysis enabled by a high performance computing environment

Combine sensitivity analysis, stochastic optimization, and modeling-to-generate-alternatives to identify robust hedging strategies for greenhouse gas mitigation

# Temoa Next Steps

## Development

- Develop more refined approach to generating stochastic data and branch probabilities
- Find ways to prune the event tree
- Improve ways to analyze outputs from stochastic runs
- Implement MGA in Temoa framework
- Develop a relational database schema for I/O data

## Application

- Adapt single-region US TIMES model to Temoa
- Begin addressing the driving questions listed on Slide 3

<http://temoaproject.org>

# Relevant papers (published or submitted)

DeCarolís J.F. (2011). Using modeling to generate alternatives (MGA) to expand our thinking on energy futures. *Energy Economics*, 33: 145-152.

Howells M., Rogner H., Strachan N., Heaps C., Huntington H., Kypreos S., Hughes A., Silveira S., DeCarolís J., Bazillian M., Roehrl A. (2011). OSeMOSYS: The open source energy modeling system: An introduction to its ethos, structure and development. *Energy Policy*, 39(10), 5850-5870.

DeCarolís J.F., K. Hunter K., S. Sreepathi. The Case for Repeatable Analysis with Energy Economy Optimization Models. *Energy Economics* (under revision).

Hunter K., S. Sreepathi S., J.F. DeCarolís , Tools for Energy Model Optimization and Analysis. Preparing submission to *Environmental Modeling and Assessment*.

# Acknowledgments

- Kevin Hunter, MS student, Civil Engineering, NCSU
- Sarat Sreepathi, PhD student, Computer Science, NCSU
- Jean-Paul Watson and Bill Hart, Sandia National Laboratory

This work would made possible through the generous support of the National Science Foundation. CAREER: *Modeling for Insights with an Open Source Energy Economy Optimization Model*. Award #1055622



National Science Foundation  
WHERE DISCOVERIES BEGIN