Computational Challenges in Large-Scale Optimization for Grid Operations and Planning

Dr. Jean-Paul Watson

Discrete Math and Complex Systems Department

Sandia National Laboratories, Albuquerque, New Mexico

Key Algorithmic Collaborators:

William Hart (Sandia)

Carl Laird (TAMU)

Roger Wets (UC Davis)

David Woodruff (UC Davis)

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company,
for the United States Department of Energy's National Nuclear Security Administration
under contract DE-AC04-94AL85000.

Combinatorial Optimization R&D at Sandia

- Efforts are centered on two primary research thrusts
  - Risk Management
    - Multi-stage, general mixed-integer
    - Efficient risk versus cost tradeoff analysis
    - Scalable Conditional Value-at-Risk (CVaR) computation
  - Multi-Stage Stochastic Optimization
    - Multi-stage, general mixed-integer
    - Massively parallel environments
- Application drivers
  - Contamination sensor network design (INFORMS Edelman Finalist)
  - Network interdiction for critical infrastructure
  - Biofuel network design
  - Electrical grid generation and transmission capacity expansion
  - Scalable unit commitment with large renewables penetration
- Funding sources
  - DOE Office of Science, US EPA, Sandia LDRD

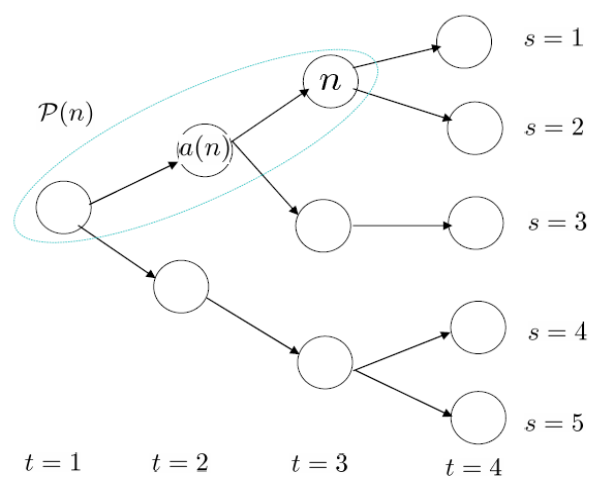
Resource Allocation: Integer and Stochastic Programming

- Deterministic Mixed-Integer Programming (MIP)
  - The PDE of Operations Research

$$
\begin{aligned}
\min \quad & \boldsymbol{c}'\boldsymbol{x} + \boldsymbol{h}'\boldsymbol{y} \\
\text{s.t.} \quad & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y} \le \boldsymbol{b} \\
& \boldsymbol{x} \in Z_+^n (\boldsymbol{x} \ge 0, \boldsymbol{x} \text{ integer}) \\
& \boldsymbol{y} \in R_+^n (\boldsymbol{y} \ge 0)
\end{aligned}
$$

  - Approximable for most real-world problems (NP-Hard)
- Stochastic Mixed-Integer Programming (SMIP)
  - SMIP = MIP + _uncertainty_ + _recourse_

$$
\begin{aligned}
\min \quad & f(\mathbf{x}) = \mathbf{c}^T\mathbf{x} + \mathbb{E}[Q(\mathbf{x}, \omega)] \\
\text{s.t.} \quad & A\mathbf{x} \ge \mathbf{b}, \quad \mathbf{x} \in \mathbb{R}_+^{n_1 - p_1} \times \mathbb{Z}_+^{p_1} \\
Q(\mathbf{x}, \omega) = \quad & \min \quad \mathbf{q}(\omega)^T\mathbf{y} \\
& \text{s.t.} \quad W\mathbf{y} \ge \mathbf{h}(\omega) - T(\omega)\mathbf{x} \\
& \qquad \mathbf{y} \in \mathbb{R}_+^{n_2 - p_2} \times \mathbb{Z}_+^{p_2}
\end{aligned}
$$



  - Still NP-Hard, but far more difficult than MIP in practice

Sandia National Laboratories

Stochastic Optimization and the Grid: Formulations

- A variety of core grid operations and planning problems are naturally and/or commonly expressed as stochastic mixed-integer programs
  - Commonly derived from basic deterministic counterparts
  - Uncertainty either was always present or driven by renewables

- Operations example
  - Unit commitment (2-stage)

- Planning examples
  - Generation expansion (2-stage and n-stage)
  - Transmission expansion (2-stage and n-stage)

- (Severely) Complicating factors
  - With and without security constraints
  - DC versus AC power flow models
  - Linear, quadratic, or higher-order line loss models

Sandia National Laboratories

Stochastic Optimization and the Grid: Challenges

- Challenge #1
  - Computation at regional and national scales
    - Most domain publications deal with “toy” problems
    - Few at-scale benchmarks widely available

- Challenge #2
  - Common definition of core operations and planning problems
    - Unit commitment literature is notoriously inconsistent
    - Makes algorithmic cross-comparison nearly impossible

- Challenge #3
  - Solving the *real* problem
    - Combining, e.g., unit commitment *and* transmission switching
    - Generation *and* transmission expansion
    - Unit commitment + transmission constraints + security constraints

Capacity Expansion as Stochastic Mixed-Integer Programming

- Many historical planning models are either deterministic or linear (or both)
  - Driven by combinations of data availability and solver maturity

- With advances in IT and solver technology, multi-stage stochastic mixed-integer formulations are becoming more prevalent in the literature
  - Singh et al. (2009), Wang and Ryan (2010), Huang and Ahmed (2009)
  - General paradigm captures key aspects of capacity expansion problems

- Key technological challenges to deploying multi-stage stochastic MIP models
  - No canonical generation and transmission capacity expansion model
  - Multi-stage stochastic MIP solvers are not yet general-purpose
  - The difficulty of multi-stage stochastic MIPs *likely* requires parallelism

- Key requirement to solve the deployment barrier
  - Modeling and solver framework to facilitate rapid prototyping of alternative solution strategies, supporting built-in parallelism

Stochastic Mixed-Integer Programming: The Algorithm Landscape

- The Extensive Form or Deterministic Equivalent
  - Write down the full variable and constraint set for all scenarios
  - Write down, either implicitly or explicitly, non-anticipativity constraints
  - *Attempt* to solve with a commercial MIP solver
    - Great if it works, but often doesn't due to memory or time limits
- Time-stage or "vertical" decomposition
  - Benders / L-shaped methods (including nested extensions)
  - Pros: Well-known, exact, easy for (some) 2-stage problems, parallelizable
  - Cons: Master problem bloating, multi-stage difficulties
- Scenario-based or "horizontal" decomposition
  - Progressive hedging / Dual decomposition
  - Pros: Inherently multi-stage, parallelizable, leverages specialized MIP solvers
  - Cons: Heuristic (depending on algorithm), parameter tuning
- Important: *Development of general multi-stage SMIP solvers is an open research area*

Progressive Hedging: A Review and/or Introduction

1. $k := 0$
2. For all $s \in \mathcal{S}$, $x_s^{(k)} := \text{argmin}_x (c \cdot x + f_s \cdot y_s) : (x, y_s) \in \mathcal{Q}_s$
3. $\bar{x}^k := (\sum_{s \in \mathcal{S}} p_s \, d_s x_s^{(k)}) / \sum_{s \in \mathcal{S}} p_s \, d_s$
4. For all $s \in \mathcal{S}$, $w_s^{(k)} := \rho(x_s^{(k)} - \bar{x}^{(k)})$
5. $k := k + 1$
6. For all $s \in \mathcal{S}$, $x_s^{(k)} := \text{argmin}_x (c \cdot x + w_s^{(k-1)} x + \rho/2 \left\| x - \bar{x}^{(k-1)} \right\|^2 + f_s \cdot y_s)$ $: (x, y_s) \in \mathcal{Q}_s$
7. $\bar{x}^{(k)} := (\sum_{s \in \mathcal{S}} p_s \, d_s x_s^{(k)}) / \sum_{s \in \mathcal{S}} p_s \, d_s$
8. For all $s \in \mathcal{S}$, $w_s^{(k)} := w_s^{(k-1)} + \rho \left(x_s^{(k)} - \bar{x}^{(k)} \right)$
9. $g^{(k)} := \frac{(1-\alpha)|\mathcal{S}|}{\sum_{s \in \mathcal{S}} p_s \, d_s} \sum_{s \in \mathcal{S}} \left\| x^{(k)} - \bar{x}^{(k)} \right\|$
10. If $g^{(k)} < \epsilon$, then go to step 5. Otherwise, terminate.

Rockafellar and Wets (1991)

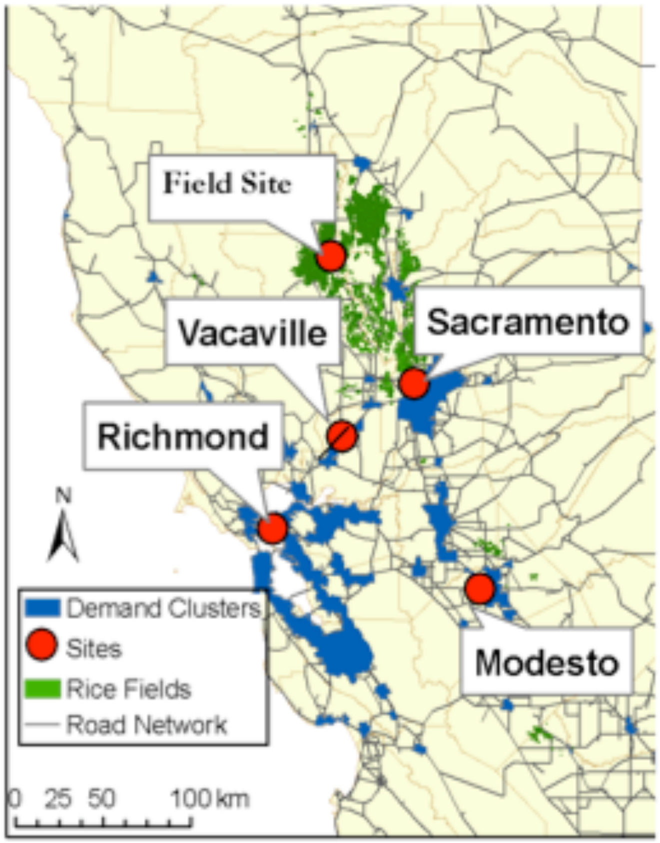
Progressive Hedging as a Stochastic Mixed-Integer Heuristic

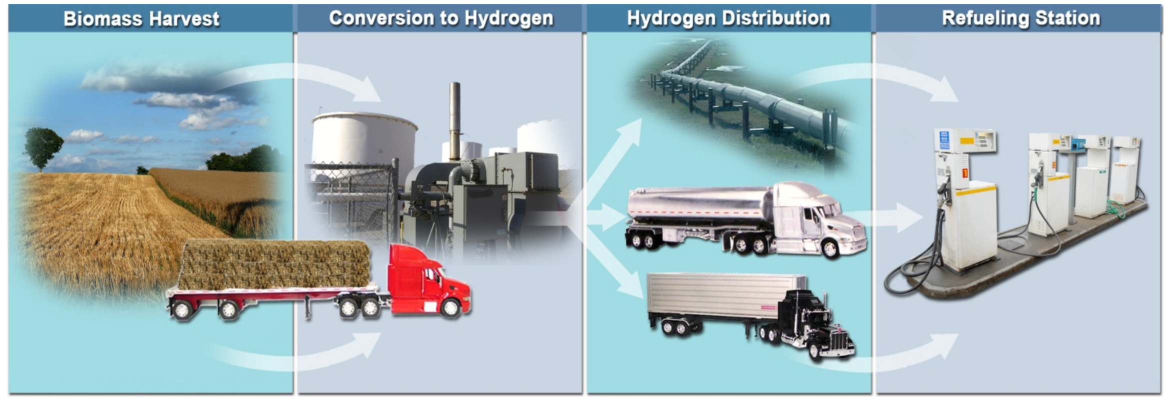
- Progressive Hedging does provably converge in the <u>*convex*</u> case, in linear time
  - NOTE: As practitioners know well, linear time can take a *long* time

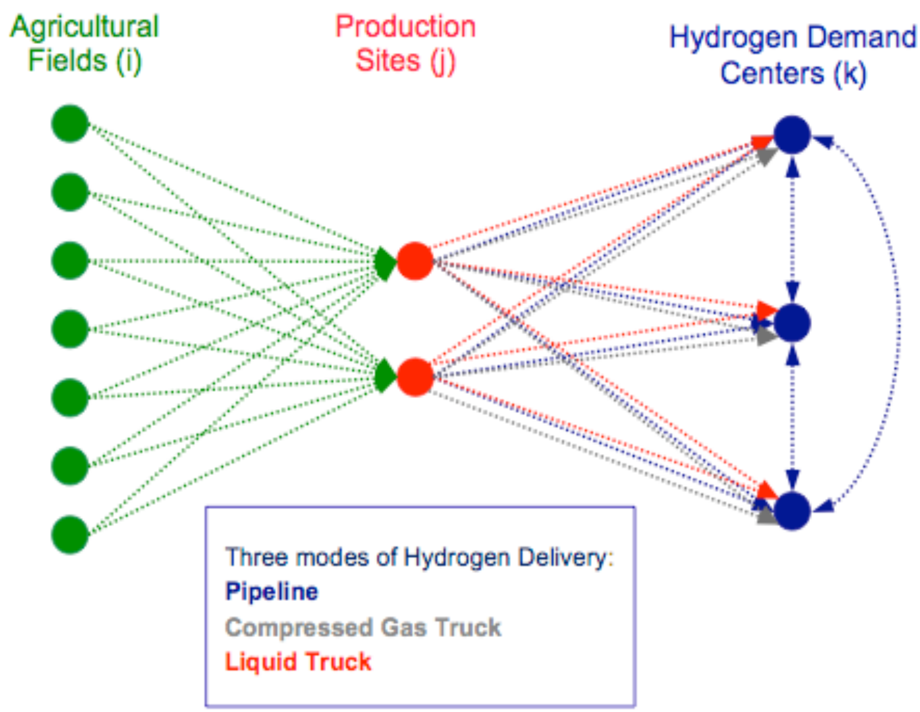
- Progressive Hedging (PH) has been successfully used as a heuristic for multi-stage mixed-integer stochastic programming
  - Løkketangen and Woodruff (1996)
  - Numerous others (Birge, Gendreau, Crainic, Rei)

- Practical and critical issues of note
  - How to pick ρ?
  - Cycle detection
  - Convergence acceleration
    - Variable fixing
    - Slamming

The Impact of Decomposition: Biofuel Infrastructure and Logistics Planning







<u>Example of PH Impact:</u>

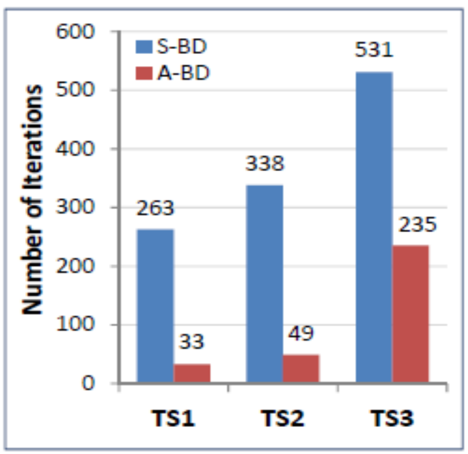
- Extensive form solve time: >20K seconds
- PH solve time: 2K seconds

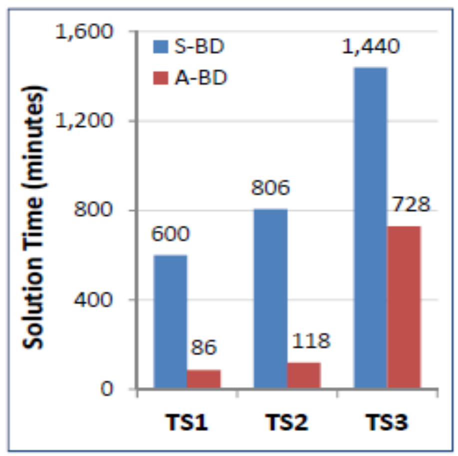
Slide courtesy of Professor YueYue Fan (UC Davis)

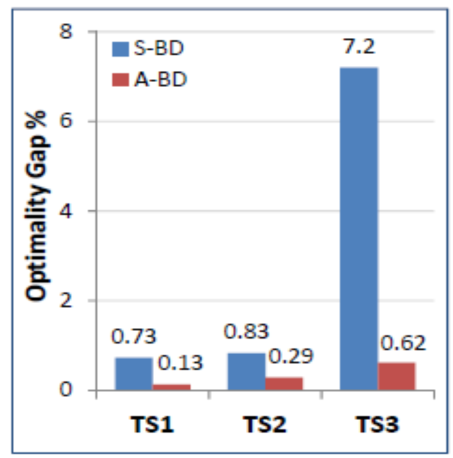
ies

The Impact of Decomposition: Wind Farm Network Design

- Where to site new wind farms and transmission lines in a geographically distributed region to satisfy projected demands at minimal cost?
- Formulated as a two-stage stochastic mixed-integer program
  - First stage decisions: Siting, generator/line counts
  - Second stage “decisions”: Flow balance, line loss, generator levels
- 8760 scenarios representing coincident hourly wind speed, demand
- Solve with Benders: Standard and Accelerated





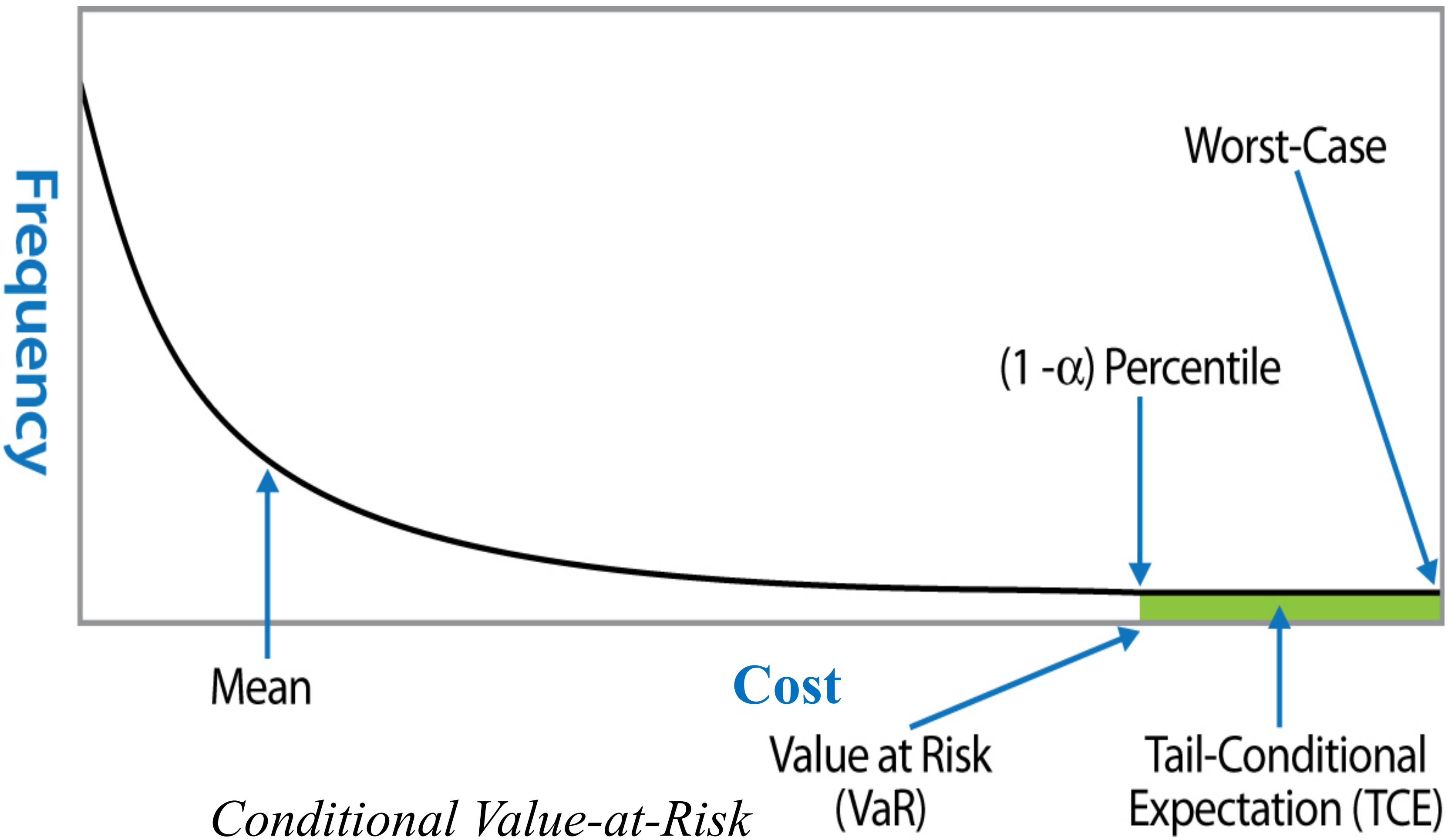


- Summary: A non-trivial Benders variant is *required* for tractable solution

Slide courtesy of Dr. Richard Chen (Sandia California)

Sandia National Laboratories

Mean versus Risk? Some Terminology



Conditional Value-at-Risk (CVaR) is a linear approximation of TCE

Progressive Hedging and Conditional Value-at-Risk

- Scenario-based decomposition of Conditional Value-at-Risk models is conceptually straightforward (Schultz and Tiedemann 2006)

Proposition 5.1. *Assume that* μ *is discrete with finitely many scenarios* $h_1, \ldots, h_J$ *and corresponding probabilities* $\pi_1, \ldots, \pi_J$*. Let* $\alpha \in (0, 1)$*. Then the stochastic program*

$$\min\{Q_{CVaR_\alpha}(x) \; : \; x \in X\} \tag{11}$$

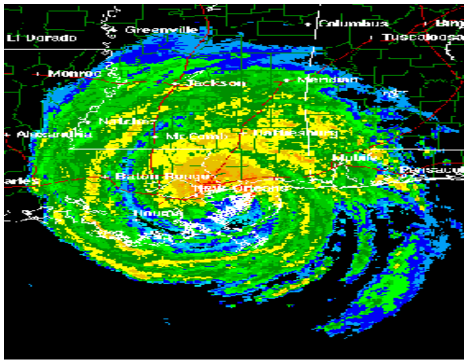
can be equivalently restated as

$$\min_{x,y,y',v,\eta} \Bigg\{ \eta + \frac{1}{1-\alpha} \sum_{j=1}^{J} \pi_j v_j \; : \; \begin{aligned} & W y_j + W' y'_j \; = \; h_j - T x, \\ & v_j \geq c^\top x + q^\top y_j + q'^\top y'_j - \eta, \\ & x \in X, \quad \eta \in \mathbb{R}, \quad y_j \in \mathbb{Z}_+^{\bar{m}}, \\ & y'_j \in \mathbb{R}_+^{m'}, \; v_j \in \mathbb{R}_+, \quad j = 1, \ldots, J \end{aligned} \Bigg\}. \tag{12}$$

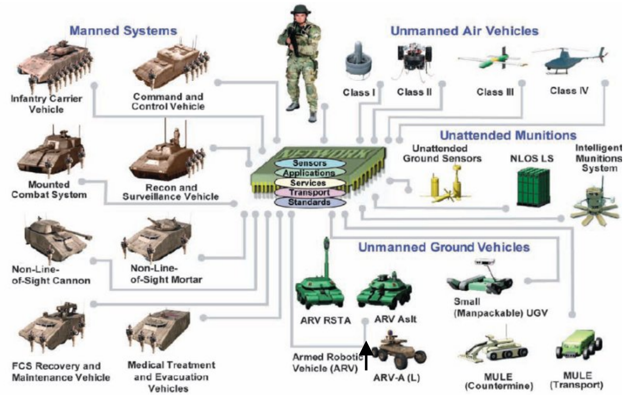
- But
  - Computational issues (e.g., trade-off curves) are largely unexplored

Selecting Scenarios to Ignore in Stochastic Optimization: Advances in Probabilistic Integer Programming Solvers

Ignoring the 100-year Flood
(Infrastructure Planning)



Capacitated Storage
(US Army Future Combat Systems)



Force-on-Force "Anomalies"
(Mission Planning)

Central Theme: The Need to Ignore a Small Fraction α of Scenarios During Optimization

$$
\begin{array}{lll}
\text{minimize} & c \cdot x + \sum_{s \in \mathcal{S}} p_s (f_s \cdot y_s) & \text{(E)} \\
\text{subject to:} & (x, y_s) \in \mathcal{Q}_s, \quad \forall s \in \{\mathcal{S} : d_s = 1\} & \\
& \sum_{s \in \mathcal{S}} p_s d_s \geq (1 - \alpha) & \\
& d_s \in \{0, 1\}, \quad \forall s \in \mathcal{S} &
\end{array}
$$

Results for network design:

- 2-8% better solutions than CPLEX, 1440m versus ~10m

Impact: - Best available heuristic for solving probabilistic integer programs
- First demonstration on large-scale, real-world problems

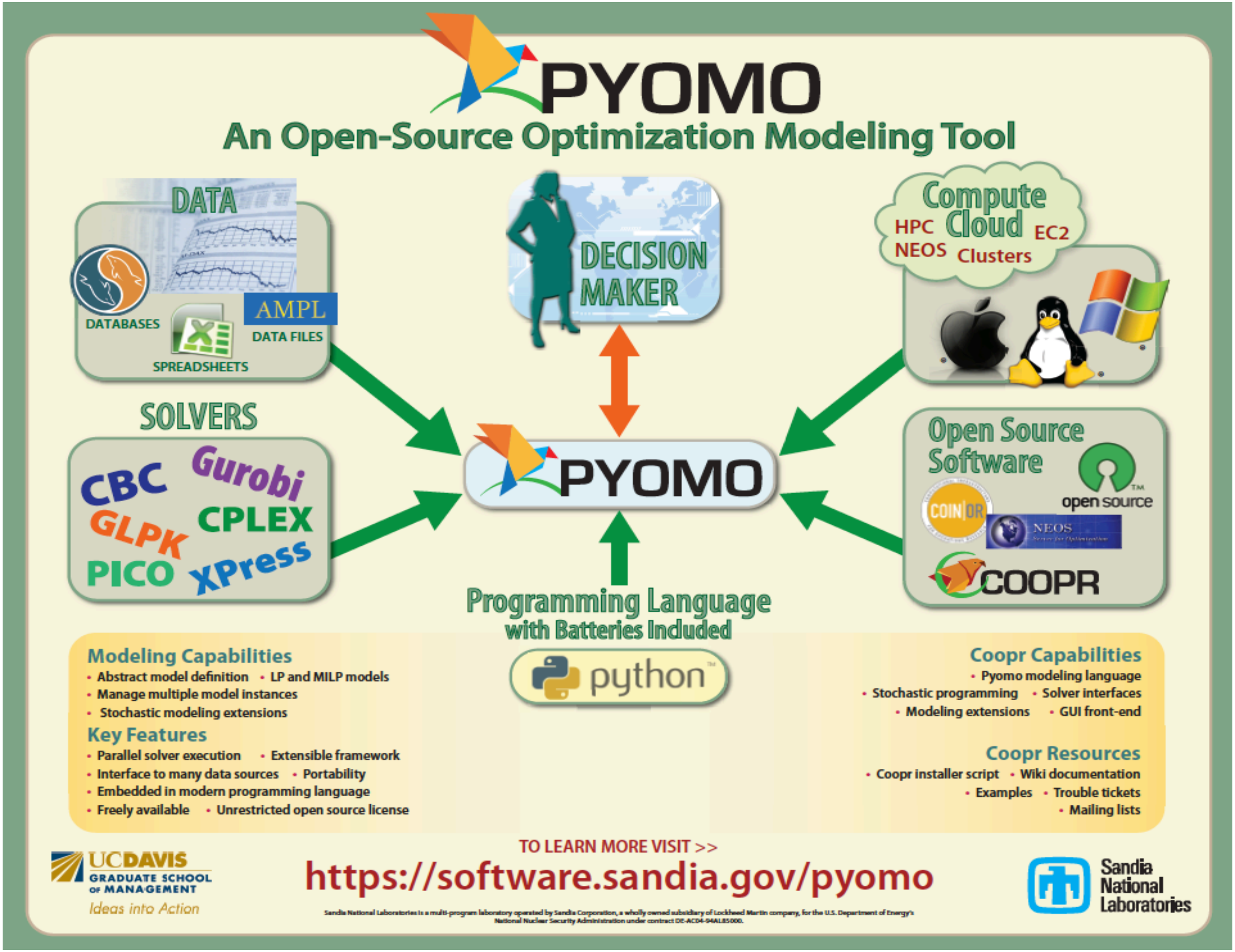
Sandia National Laboratories

Scenario Selection and the Power Grid

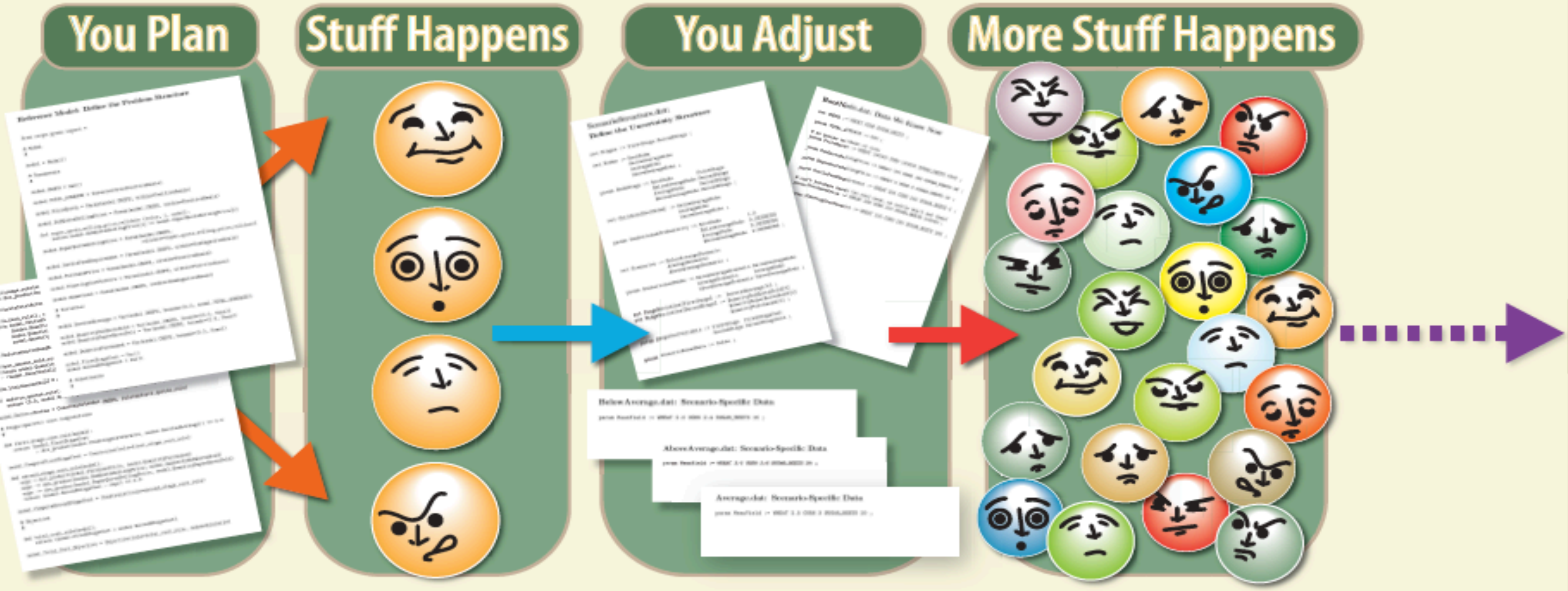
- Transmission and generation capacity expansion
  - The 100-year flood analog to risk management in the grid
  - Regulations are often expressed as, e.g., demand must be satisfied with some probability α (LOLP – Loss of Load Probability)
  - LOLP can be formulated as a math program with scenario selection

- Growth in computational difficulty is substantial for even toy grid problems
  - Often tackled with "soft" heuristics (e.g., GRASP or tabu search)
  - Stochastic programming alternatives are rare to non-existent
    - A key opportunity and challenge
  - Driver for heuristics is predominantly use of AC power flow models

- Stochastic unit commitment is another open application driver
  - Large numbers of scenarios required for stable solutions further complicates computational tractability


PYOMO
An Open-Source Optimization Modeling Tool
DATA
DATABASES
AMPL
DATA FILES
SPREADSHEETS
DECISION MAKER
Compute Cloud
HPC
EC2
NEOS
Clusters
SOLVERS
CBC
Gurobi
GLPK
CPLEX
PICO
XPress
PYOMO
Open Source Software
open source
COIN|OR
NEOS
COOPR
Programming Language with Batteries Included
python
Modeling Capabilities
• Abstract model definition • LP and MILP models
• Manage multiple model instances
• Stochastic modeling extensions
Key Features
• Parallel solver execution • Extensible framework
• Interface to many data sources • Portability
• Embedded in modern programming language
• Freely available • Unrestricted open source license
Coopr Capabilities
• Pyomo modeling language
• Stochastic programming • Solver interfaces
• Modeling extensions • GUI front-end
Coopr Resources
• Coopr installer script • Wiki documentation
• Examples • Trouble tickets
• Mailing lists
UCDAVIS
GRADUATE SCHOOL OF MANAGEMENT
Ideas into Action
TO LEARN MORE VISIT >>
https://software.sandia.gov/pyomo
Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin company, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.
Sandia National Laboratories

Hedging Against Uncertainty:
A Modeling Language and Solver Library



PYOMO

PySP: Stochastic Programming in Python

COOPR

Multi-Stage Planning for Uncertain Environments

- Explicitly capture recourse
- Uncertainty modeling framework
- Integrated solver strategies

What We Do:

- Mixed decision variables
  - Continuous
  - Integer/Binary
- General multi-stage
- Stochastic programming
  - Expected value
  - Conditional Value-at-Risk
  - Scenario selection
- Cost confidence intervals

How We Do It:

- Deterministic equivalent
- Scenario-based decomposition
  - Progressive Hedging
  - Customizable accelerators
- Algebraic modeling via Pyomo
- SMP and cluster parallelism
- Integrated high-level language support
- Multi-platform, unrestrictive license
- Open source, actively supported by Sandia
- Co-Managed by Sandia and COIN-OR

UC DAVIS GRADUATE SCHOOL OF MANAGEMENT *Ideas into Action*

TO LEARN MORE VISIT > https://software.sandia.gov/trac/coopr/wiki/PySP

Sandia National Laboratories

Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin company, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Stochastic Programming and High-Performance Computing

- Decomposition algorithms for solving multi-stage stochastic mixed-integer programs are "naturally" parallelizable
  - L-shaped and Progressive Hedging are particularly amenable
- Practical issues arise as the number of scenarios grows
  - Even the most modest branching processes in multi-stage decision environments lead to thousands to millions of scenarios
  - MIP solve times are heterogeneous, leading to poor parallel efficiency
- Current capabilities in PySP:
  - Scalability to order-thousand scenarios and processors
- In-progress efforts
  - Asynchronous decomposition algorithms
  - IBM Research Blue Gene deployment
  - EC2 / Gurobi deployment
- Major deployment issue: MIP solver licensing to thousands of processors
  - Mitigated in part by Gurobi EC2 deployment

Scenario Sampling: How Many is Enough?

- Discretization of the scenario tree is "standard" in stochastic programming
  - Often, no mention of solution or objective stability
  - Let alone rigorous statistical hypothesis-testing of stability
  - *Don't trust anyone who doesn't show you a confidence interval*

- Various approaches in the literature
  - Sample Average Approximation (asymptotic results)
  - Multiple Replication Procedure (finite sample results)

- Formal question we are concerned with
  - What is the probability that $\hat{x}$'s objective function value is suboptimal by more than α%?
  - But making due with a fixed set or "universe" or scenarios

- Initial implementation available in PySP
  - Mixed results, as we'll see on the next two slides

Scenario Sampling and Wind Farm Network Design

1000 scenarios, randomly sampled from a universe of 8760 scenarios

$\hat{n}$	n_g	n	Obj	E(Obj)	Gap(0.05)
70	2	465	89956	90639	828
70	5	186	89934	90639	764
70	10	93	89941	90639	870
70	20	46	89929	90639	1127
70	40	23	89929	90639	1356
140	2	430	89734	89779	354
140	5	172	89721	89779	272
140	10	86	89721	89779	462
140	20	43	89721	89779	792
140	40	21	89657	89779	1178
280	2	360	89755	89648	198
280	5	144	89744	89648	435
280	10	72	89750	89648	628
280	20	36	89750	89648	956
280	40	18	89750	89648	1403
420	2	290	90324	88832	251
420	5	116	90333	88832	555
420	10	58	90328	88832	718
420	20	29	90331	88832	996
420	40	14	90284	88832	1664
560	2	220	90577	89108	431
560	5	88	90587	89108	456
560	10	44	90583	89108	800
560	20	22	90585	89108	1252
560	40	11	90584	89108	2042

- Number of scenarios to form the baseline solution
- Number of groups used to form the confidence interval
- Number of scenarios in each group

Result of Mak, Morton, and Wood (1999) Multiple Replication Procedure:

- Objective function value is remarkably stable across different parameterizations of the procedure
- Confidence interval widths are relatively small for a planning problem
- Results are stable across replications of the same parameterization of the MRP procedure

Practical impact: We don't need 8760 scenarios!

Scenario Sampling: Research Directions

- Generation expansion problems seem (based on a problem sample size of 3) to not require as many scenarios as one might think necessary
  - Tight confidence intervals with relatively low numbers of scenarios

- Unfortunately, this result is far from universal
  - For stochastic unit commitment, results suggest the need for orders-of-magnitude more scenarios than currently in use (100 to 10K)
  - Only hold for expectation-minimization variations of expansion problems – Conditional Value-at-Risk is problematic

- Major open challenge:
  - Multiple Replication Procedures for *multi-stage* stochastic programs
  - Transition to computing confidence intervals for *policies*

Conclusions

- Stochastic mixed-integer programs are a natural modeling paradigm for solving many core grid operations and planning problems

- Solver technologies capable of solving realistic instances are emerging
  - But many challenges remain, both in terms of research and deployment

- Sandia is developing software to address what we view as the challenges (or at least challenges we can effectively address!)
  - Frameworks to support rapid modeling and solver prototyping
  - Scalable parallelization of decomposition strategies
  - Rigorous quantification of uncertainty bounds on solution costs
  - Open-source solutions
    - Sandia is mandated to collaborate with and aid industry – not compete

- For more information:
  - https://software.sandia.gov/trac/coopr/wiki/PySP -or- jwatson@sandia.gov

Questions?

- Thanks!