An Overview of Sampling Methods in Stochastic Programming

Andrew Schaefer

University of Pittsburgh
Department of Industrial Engineering

October 16, 2008

Outline

- What is stochastic programming?
- How do I formulate a stochastic program?
- Properties of stochastic programs.
- Have these been used in industry?
- How can I solve a stochastic program?
- Sampling Approaches
- Scenario Reduction Techniques

Deterministic Optimization

- Two broad categories of optimization models:

1)Deterministic models

- Parameters/data are known with certainty
- Can be thought as a special case of stochastic models
- Four main sub-categories:
  - Linear Programs
  - Nonlinear Programs
  - Integer Programs
  - Dynamic Programs

2)Stochastic models

- Parameters/data are not known with certainty

What is Stochastic Programming (SP)?

- A Stochastic Program is a mathematical program in which some of the parameters defining a problem instance are random
- A stochastic linear program (SLP) is the simplest case of stochastic program

Common Assumptions in SP:

- The probability distribution for the uncertainty is known and independent of the decisions (exceptions, e.g. Goel and Grossmann)
- The stages interact linearly

Formulating Stochastic Programs

Stages (decision epochs)

The sequence of events and decisions in a two-stage SLP with recourse (2SLPwR):

$x \rightarrow$ Realization of $\omega \rightarrow y(\omega)$

•First-stage decisions (x) are taken before the uncertainty ω is realized (known)

•Second-stage decisions ($y(\omega)$) are taken as corrective actions after the actual value of ω becomes known

Standard Objective: minimize first-stage cost plus **expected** second-stage costs

Formulating Stochastic Programs

- Familiar Linear Program (LP):

$$\begin{aligned} \min\ & c^T x \\ \text{s.t. } & Ax = b, \\ & x \geq 0 \end{aligned}$$

- **2SLPwR**

$$\begin{aligned} \min\ & c^T x + Q(x) \\ \text{s.t. } & Ax = b \\ & x \geq 0 \end{aligned}$$

where **Q(x),** the *expected recourse function*, gives the expected cost of the optimal second-stage decision given first-stage decision x

What can be random?

- In the second stage, the objective, constraints and right-hand sides are permitted to be random. Also, the matrix describing the relationship between x and y

- **For scenario ω:**
  - Objective: $q(\omega)$
  - Right-hand side: $h(\omega)$
  - Constraint (*recourse*) matrix: $W(\omega)$
  - Matrix relating x to y (*technology matrix)*: $T(\omega)$

Expected Recourse Function

- How do we define $Q(x)$?

- For any scenario ω, Define $Q(x,\omega)$ by

$$Q(x,\omega) = \min q(\omega)^t y(\omega)$$
$$\text{s.t.} \quad W(\omega)y(\omega) = h(\omega) - T(\omega)x$$
$$y(\omega) \geq 0$$

And $Q(x) = E_\omega Q(x,\omega)$

Notice that the relationship between x and $y(\omega)$ must be described linearly

Properties of the Expected Recourse Function

- What do we know about Q(x)?
- Q(x) is convex (whew!)
  - But not if y(ω) must be integer-valued
  - This convexity is critical for solving SLPs
- If there are a finite number of scenarios, Q(x) is piece-wise linear

Expected Recourse Function

The expected recourse function Q(x) is convex and, if there are a finite number of scenarios, is also piece-wise linear

"Extensive-Form" Two-stage SLPs with Recourse

- We could solve a 2SLPwR as a (really big) linear program, called the *extensive form*

$$\text{Min } z = c^T x + E_\omega[\min q(\omega)^T y(\omega)]$$

s.t.

$$Ax = b$$

$$T(\omega)\, x + W(\omega)\, y(\omega) = h(\omega) \text{ for all } \omega$$

$$x \geq 0,\ y(\omega) \geq 0$$

Extensive Form of a Two-stage Stochastic Linear Program

$$
\begin{array}{llllllll}
\text{Min} & c^T x & + \; p_1 q_1^T y_1 & + \; p_2 q_2^T y_2 & + \cdots & & + \; p_s q_s^T y_s & \\
\text{s.t.} & Ax & & & & & & = b \\
& T_1 x & + \; W_1 y_1 & & & & & = h_1 \\
& T_2 x & + & W_2 y_2 & & & & = h_2 \\
& \vdots & + & & . & & & = \vdots \\
& \vdots & + & & & . & & = \vdots \\
& \vdots & + & & & & . & = \vdots \\
& T_K x & + & & & & W_K y_K & = h_K \\
& x \geq 0, & y_1 \geq 0, & y_2 \geq 0, & \ldots\ldots\ldots, & & y_K \geq 0 &
\end{array}
$$

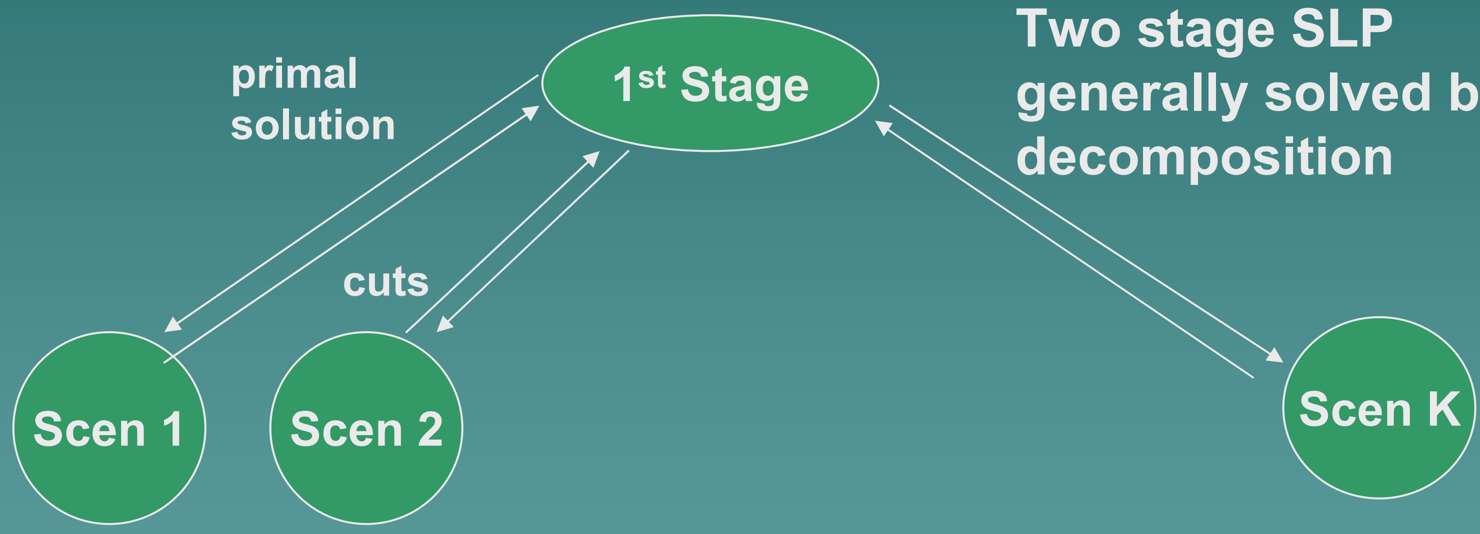
“Staircase Structure”

- This “dual block angular” or “staircase” structure is critical for solving stochastic linear programs

SP Applications - Examples

- Investment decisions in portfolio planning:
  - Portfolio planning Problems must be implemented before stock performance can be observed
- Power Generation:
  - Utilities must plan power generation before the demand for electricity is realized
- Supply chain management:
  - Uncertainty in demand and supply data is considered together with capacity limits and service level requirements
- Telecommunications network planning

How can I solve Two-Stage Stochastic Linear Programs?



Two stage SLP generally solved by decomposition

How do we solve the simplest form of Stochastic Programs?

- Linear Approaches
  - The L-Shaped Method (most common)
    - (Single cut & Multicut version)
    - Numerous computational enhancements
  - Inner Linearization Methods (won't discuss)
  - Basis Factorization Methods (won't discuss)

The L-Shaped Method

- The most commonly used technique.
- Basic idea: To approximate the convex term in the objective function.
- Recourse function involves a solution of all second stage recourse linear programs, we want to avoid numerous function evaluations for it.
- Therefore, divide the problem into two stage:
  - Master Problem
  - Sub-problems
- Converges to the optimal solution in finite steps with adding 2 new type of constraints to the master problem called feasibility cuts and optimality cuts.
- Feasibility cuts – ensure that the second-stage problems are feasible
- Optimality cuts – relate the second-stage costs to first-stage constraints

L-shaped Restricted Master Problem

Master problem is as follows:

$$
\begin{aligned}
\text{min} \quad & z = c^T x + \theta \\
\text{s.t} \quad & Ax = b \\
& D_\ell x \geq d_\ell \qquad \ell = 1, \ldots\ldots, s \rightarrow \textbf{Feasibility cuts} \\
& E_\ell x + \theta \geq e_\ell \qquad \ell = 1, \ldots\ldots, r \rightarrow \textbf{Optimality cuts} \\
& x \geq 0 \qquad \theta \in \Re
\end{aligned}
$$

L-shaped Optimality Subproblem

Recourse problem for scenario k given first-stage x is as follows:

$$\min \quad w = q_k^T y$$

$$\text{s.t} \quad Wy = h_k - T_k x$$

$$y \geq 0$$

Let π denote the duals. Then $E_l = \pi^T T_k$ and $e_l = \pi^T h_k$

Algorithm

At each iteration:

Step 1: Solve the master problem

Step 2: If the solution to the master problem (x*) leads to feasible recourse problems for all scenarios,

- Go to step 3
- Else add a *FEASIBILITY CUT* and go to step 1 and solve master problem again.

Step 3: If the expected value for the optimal values of the recourse problems (w*) is no greater than Θ obtained in step 1

- Stop the current solution is optimal,
- else add an *OPTIMALITY CUT* and go to step 1 and resolve master problem.

Expected Recourse Function

The expected recourse function Q(x) is convex and, if there are a finite number of scenarios, is also piece-wise linear

L-shaped optimality cuts support Q(x) from below. If there are a finite number of scenarios, there are a finite number of optimality cuts

Multicut Version

- Instead of using Θ to represent Q(x), we use Θ^k to represent $Q(x,\omega^k)$
- We may add as many as K cuts per iteration, but we get more information
- In general, multicut works well if there aren't "too many" scenarios
- For integer first stage, multicut is not competitive with single cut

Stochastic Integer Programs

- When recourse problem is an MIP, Q(x) is (in general) non-convex and discontinuous
- The absence of general efficient methods reflects this
- Some techniques have been proposed that address specific problems or use a particular property
- Much work needs to be done to solve SIPs efficiently
- No industrial-sized problems with integer recourse have been solved thus far
- The field is expected to evolve a great deal in the future

How to solve SIPs?

- A set of valid feasibility cuts and optimality cuts, which are based on duality theory of linear programming, is known to exist in the continuous case and forms the basis of the classical L-shaped method
- They can also be used in the case where only the first-stage variables contain some integrality restrictions
- The most common method to solve SIPs is the so-called *Integer L-Shaped Method*
  - General method, but often ineffective

Multistage SP with Recourse

- Involves a sequence of decisions that react to outcomes that evolve over time
- The extensive form of an N-stage fixed-recourse problem:

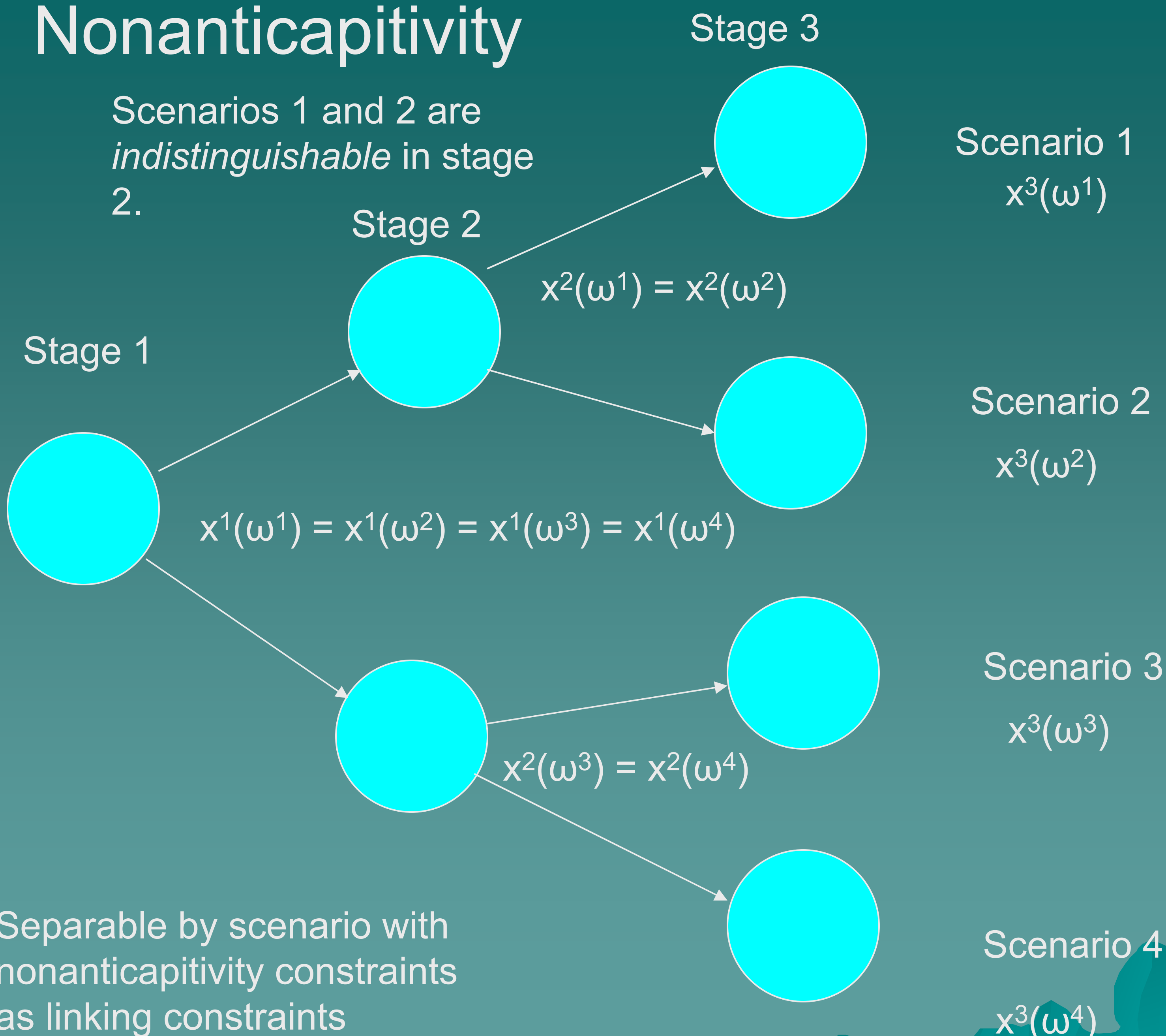
min $$z = c^1 x^1 + E_{\xi^2}[c^2(\omega) x^2(\omega^2) + \ldots + E_{\xi^N}[c^N(\omega) x^N(\omega^N)]\ldots]$$

s.t.
$$W^1 x^1 = h^1$$
$$T^1(\omega) x^1 + W^2 x^2(\omega^2) = h^2(\omega)$$
$$\vdots$$
$$T^{N-1}(\omega) x^{N-1}(\omega^{N-1}) + W^N x^N(\omega^N) = h^N(\omega)$$
$$x^1 \geq 0, x^t(\omega^t) \geq 0, t = 2, \ldots, N$$

where ω^t denotes the history up to time *t*

Nonanticapitivity

Scenarios 1 and 2 are *indistinguishable* in stage 2.



Separable by scenario with nonanticapitivity constraints as linking constraints

Solving SPs with Nonanticipativity

- Such a model is decomposable by scenario, where nonanticipativity constraints are linking constraints
- Lagrangian relaxation of linking constraints
- For reasonably large scenario trees, the number of possible nonanticipativity constraints is enormous

Nested Benders' Decomposition

- Another method generalizes the L-shaped method
- Candidate solutions are passed from stage t to stage $t+1$
- L-shaped cuts (feasibility or optimality) are passed from stage $t+1$ to stage t

Continuous Distributions

- How can stochastic programming handle continuous distributions?
- With a continuous distribution, there are infinitely many scenarios (Σ becomes $\int$)
- The extensive form formulation is infinite-dimensional
- However, there are ways to overcome these difficulties

Sample Average Approximation (SAA)

- One approach is to sample from the continuous distribution ($\omega^1, \ldots, \omega^N$)
- Assign each scenario probability 1/N
- How big should N be?
  - Use statistical properties to estimate convergence to optimal solution

SAA Advantages

- The two key sources of difficulty in solving stochastic programs:
  - Exact evaluation of the expected recourse function.
  - Optimizing the expected recourse function over the first stage decisions.
- SAA addresses the above difficulties.

SAA Main Idea

- The main idea of the SAA method is as follows:
  - A sample $\xi^1, \ldots, \xi^N$ of N realizations of the random vector $\xi(\omega)$ is generated.
  - Q(x) is approximated by

$$\frac{1}{N}\sum_{n=1}^{N} Q(x, \xi^n)$$

SAA Main Idea

- Sample average approximation problem

$$\min_{x \in X} \quad c^T x + \frac{1}{N}\sum_{n=1}^{N} Q(x, \xi^n)$$

  is then solved by a deterministic optimization algorithm.
- Let ζ^N and x^N denote the optimal value and the optimal solution of the SAA problem, respectively.
- Let ζ^* and x^* denote the optimal value and the optimal solution of the true problem, respectively.

The crucial issues to address

- Whether ζ^N and x^N converges to their true counterparts ζ^* and x^*.
- If so, can we analyze the rate of convergence?
- Can we estimate the required sample size to obtain a true optimal solution with certain confidence?
- Is there an efficient optimization approach for solving the SAA problem for the required sample size?
- Can we take advantage of *variance reduction techniques?* (e.g. Antithetic Variables, Stratified Sampling, Conditional Sampling, etc…)

SAA Results

- A solution to the SAA problem converges to a solution of the true problem as N $\rightarrow \infty$. (Schultz, 1996).

- For SLPs with discrete distributions, an optimal solution of the SAA problem provides an exact optimal solution of the true problem with probability approaching one exponentially fast as *N* increases. (Shapiro and Homem-de-Mello, 98)

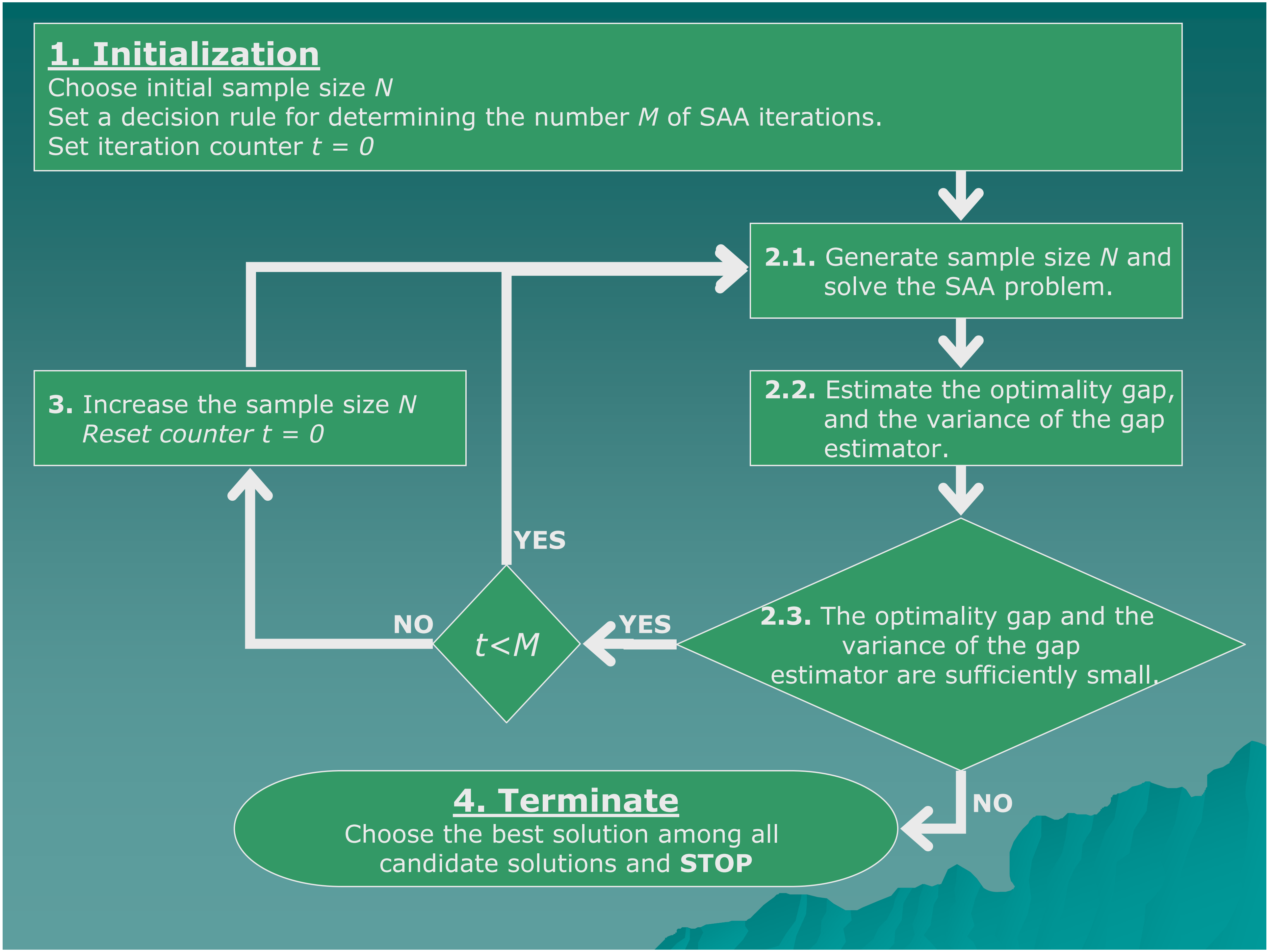
SAA Results cntd…

- Kleywegt et al.(2001) extended the convergence of the SAA approach to SPs where the set of first-stage decisions is finite.

- Ahmed and Shapiro (2002) extended these results to two-stage stochastic programs with integer recourse where the space of feasible first-stage decisions is infinite.

SAA Algorithm Design

- Selection Of the Sample Size *N*
  - Theoretical bounds exist on the sample size required to find an ε-optimal solution with probability at least $1-\alpha$.
  - They are usually hard to compute and far too conservative to obtain a practical estimate.
  - The choice of sample size *N* may be adjusted dynamically, depending on the results of preliminary computations.
  - The trade-off between the quality of an optimal solution of the SAA problem, and the bounds on the optimality gap on the one hand, and computational effort on the other hand, should be taken into account.


1. Initialization
Choose initial sample size N
Set a decision rule for determining the number M of SAA iterations.
Set iteration counter t = 0
2.1. Generate sample size N and solve the SAA problem.
2.2. Estimate the optimality gap, and the variance of the gap estimator.
2.3. The optimality gap and the variance of the gap estimator are sufficiently small.
YES
NO
t<M
YES
NO
3. Increase the sample size N
Reset counter t = 0
4. Terminate
Choose the best solution among all candidate solutions and STOP

Using Sampling in the L-Shaped Algorithm

- Importance sampling
  - To reduce variance in deriving each cut based on a large sample
- Stochastic decomposition
  - Uses a single sample stream to derive many cuts that eventually phase out as iteration numbers increase

Importance Sampling

- Importance sampling
  - Techniques to reduce the variance
- Dantzig and Glynn (1990), Infanger (1992)
  - Sample Q in the L-shaped method instead of actually computing it
  - Importance sampling can be used to achieve converging results

Importance Sampling

- $Q(x)$ is an expectation of random variables $Q(x,\xi)$.
- Estimate $Q(x)$ instead of actually computing it:
  - n independent random vectors: $\xi^1, \xi^2, \ldots, \xi^n$
  - Crude Monte Carlo Sampling

$$\bar{Q}(x) = \frac{1}{n}\sum_{i=1}^{n} Q(x,\xi^i)$$

  - Importance Sampling

$$\bar{Q}(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{Q(x,\xi^i)p(\xi^i)}{q(\xi^i)}$$

Probability of observing ξ^i

A probability mass function introduced to reduce the variance

Importance Sampling

- Variance of $\bar{Q}(x)$

$$\operatorname{var}(\bar{Q}(x)) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{Q(x,\xi^i)p(\xi^i)}{q(\xi^i)} - Q(x)\right)^2 q(\xi^i)$$

  – Choosing a good probability mass function q help to reduce the variance!

- By using importance sampling within the L-shaped algorithm, we calculate estimates for the coefficients and RHS values of the cuts.

Stochastic Decomposition

- Works for 2SLP with complete recourse (Higle & Sen)
- Sample to create cuts on each iteration of the L-shaped algorithm
- These cuts fall off (are given less weight) as the algorithm progresses

Stochastic Decomposition

- Step 1. Set $v = 0, \xi^0 = \bar{\xi}$ and let x^1 solve

$$\min_{Ax=b, x\geq 0} \{c^T x + Q(x, \xi^0)\}.$$

- Step 2. Let $v = v + 1$ and ξ^v be an independent sample generated from ξ.

  Find $Q^v(x^v) = \frac{1}{v}\sum_{s=1}^{v} Q(x^v, \xi^s) = \frac{1}{v}\sum_{s=1}^{v} (\pi_s^v)^T (\xi^s - Tx^v)$

  Let $E_v = \frac{1}{v}\sum_{s=1}^{v} (\pi_s^v)^T T$ and $e_v = \frac{1}{v}\sum_{s=1}^{v} (\pi_s^v)^T \xi^s$

Stochastic Decomposition

- Step 3. Update all previous cuts by $E_s \leftarrow \frac{v-1}{v} E_s$ and $e_s \leftarrow \frac{v-1}{v} e_s$ for $s = 1, \ldots, v-1.$
- Step 4. After adding a new cut by Step 2 and updating the existing ones with Step 3, solve the updated L-shaped master problem to obtain x^{v+1} and go to Step 2.

Convergence and Extensions

- Higle and Sen have shown that stochastic decomposition can converge under certain conditions
- In the last ten years Sen has extended this and other ideas to stochastic integer programs

Scenario Reduction

- **Number of scenarios** is a key parameter determining the computational effort for solving stochastic optimization models.
- **Main Idea:** Scenario-based approximations of the random data process.
- Small number of scenarios represent reasonably well approximations.

Optimal Scenario Reduction

- For a stochastic program given by a probability distribution P with finitely many scenarios and their probabilities, ***the optimal scenario reduction*** consists of:

  - determining a scenario subset (of prescribed cardinality or accuracy) and
  - assign new probabilities to the preserved scenarios such that
  - the corresponding reduced probability measure Q is the closest to the original measure P in terms of a certain probability distance between P and Q.

Scenario Reduction Results

- Optimal Scenario Reduction problem is NP-Hard.
- **Heitsch and Römisch, 2003** and **Dupačová et al., 2003** developed forward and backward type algorithms for approximately computing reduced probability measures.

Conclusions

- Stochastic programming is an emerging field that extends classical optimization to stochastic and dynamic settings
- SP has significant potential in EWO
- Stochastic programs are difficult to solve, so modeling becomes challenging
- Despite their inherent difficulty, industrial-sized stochastic programs can now be solved