Introduction to Stochastic Combinatorial Optimization

Stefanie Kosuch
PostDok at TCSLab
www.kosuch.eu/stefanie/

Guest Lecture at the CUGS PhD course
"**Heuristic Algorithms for Combinatorial Optimization Problems**"

What is the interest of Stochastic Combinatorial Optimization?

Combinatorial "real world problems" often subject to uncertainties

- Not all parameters known when decision has to be made: market fluctuations, available capacity...
- Own decision depends on future decision of other parties: competition, clients, government...
- Setting of problem might change: weather, location...

What is the interest of Stochastic Combinatorial Optimization?

Combinatorial "real world problems" often subject to uncertainties

- Not all parameters known when decision has to be made: market fluctuations, available capacity...
- Own decision depends on future decision of other parties: competition, clients, government...
- Setting of problem might change: weather, location...

- Random parameters easy to implement $\rightarrow$ Random variables
- "Code" other uncertainties in parameters

Definition

Stochastic Combinatorial Optimization concerns the study and resolution of Combinatorial Optimization problems that involve uncertainties.

Definition

Stochastic Combinatorial Optimization concerns the study and resolution of Combinatorial Optimization problems that involve uncertainties.

Note that there is a slight difference between the usage of the term "Stochastic Optimisation" and "Stochastic Programming".
Stochastic Programming designs the modeling and study of optimization problems that involve uncertainties.
Stochastic Optimization addresses the study of optimization algorithms that are either randomized or created to solve stochastic programming problems.

However, these definitions are not always properly used and of course both fields intersect in a lot of aspects.

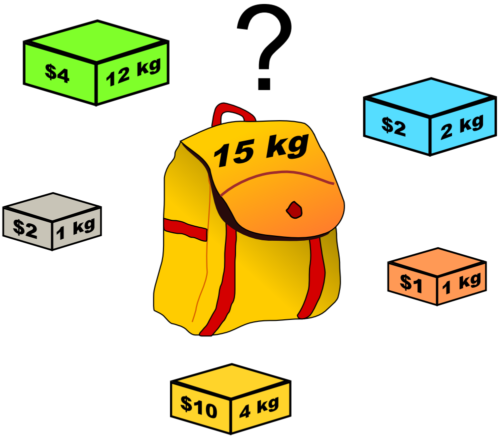
Objectives of this lecture

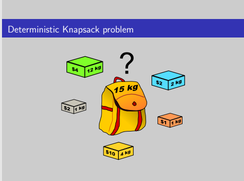
- Give you examples of SCO-problems.
- Give you an idea of how uncertainties can be modeled (most common models).
- Give you an idea of why Stochastic Optimization is hard.
- Give you an idea of how SCO-problems can be solved.
- Give you an idea of why metaheuristics are important tools in SCO.

Outline

1. 2 examples of SCO problems
2. Modeling Stochastic Combinatorial Optimization Problems
   - "Where" does the randomness occur?
     - Randomness occurs in the objective function f.
     - Randomness occurs in the constraint function g.
   - When are the actual parameters revealed?
     - Parameters are revealed after decision has been made.
     - Parameters are revealed before corrective decision is made.
     - Parameters are revealed in several stages.
3. Solving Stochastic Combinatorial Optimization problems
   - Problems/Difficulties
   - Deterministic Reformulation
   - Sample Average Approximation
   - Metaheuristics for SCO problems
4. Conclusion
   - Further Reading

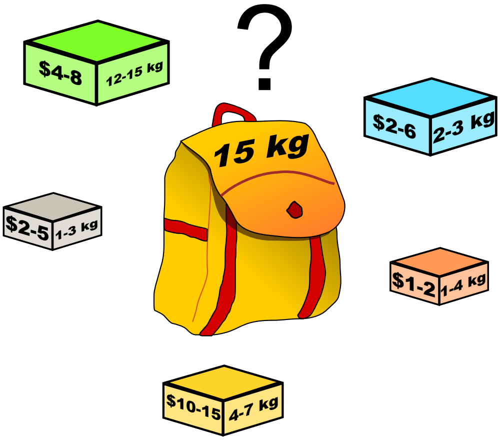
Deterministic Knapsack problem

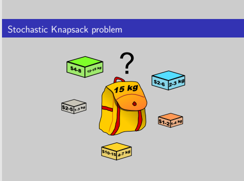




Deterministic knapsack problem: The problem consists in choosing a subset out of a given set of items such that the total weight (or size) of the subset does not exceed a given limit (the capacity of the knapsack) and the total benefit/reward of the subset is maximized.

Stochastic Knapsack problem





What happens if item rewards or weights are random? What is a feasible solution? For example, is it allowed to add all items apart from the green one although they *might* violate the capacity constraint? And what happens if they do?

Possible ways to handle capacity constraint

- knapsack constraint violated $\Rightarrow$ penalty
- probability of capacity violation restricted
- decision can be corrected later (add. costs/reduced rewards)

Possible ways to handle capacity constraint

- knapsack constraint violated ⇒ penalty
- probability of capacity violation restricted
- decision can be corrected later (add. costs/reduced rewards)

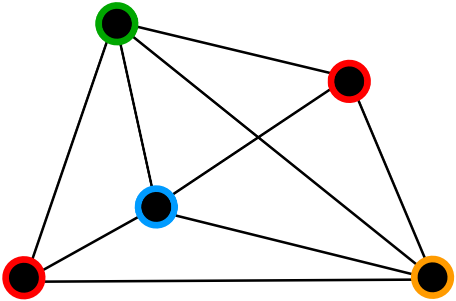
In the first two examples violation was acceptable. But what if a violation is not allowed, in any case? Well, we could force our solution to *always* respect the knapsack constraint. In this case at most 3 items could be chosen, at a much lower reward.

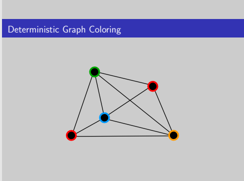
Or we could see, if a correction might be possible later (3rd example), i.e. we chose the 4 items (not the green one) and then, if their total weight exceeds the capacity, we reject one item.

Possible ways to handle capacity constraint

- knapsack constraint violated $\Rightarrow$ penalty
- probability of capacity violation restricted
- decision can be corrected later (add. costs/reduced rewards)

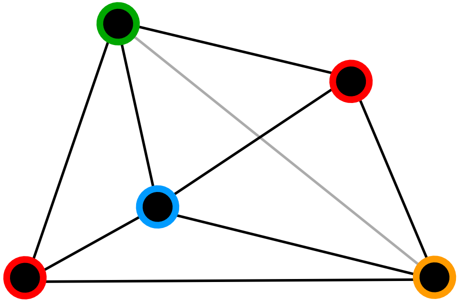
Deterministic Graph Coloring



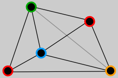


Deterministic Graph Coloring: Color a graph such that no two adjacent vertices are colored in the same color and such that a minimum number of colors is used.
In this example: Use of 4 colors is optimal as graph contains complete graph with 4 vertices.
Application: Assignment problems. The vertices could represent university courses, two courses are linked iff there is at least one student that wants to attend both courses. Coloring the obtained modelgraph with the minimum number of colors tells you how many time slots you need to schedule these courses.

Stochastic Graph Coloring

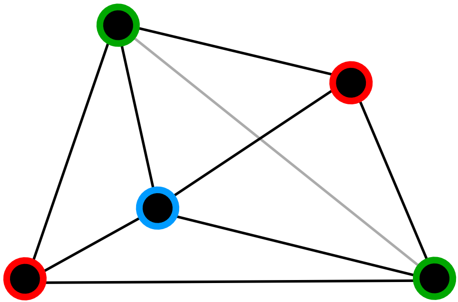


Stochastic Graph Coloring



What if, at the moment where you have to create the schedule, you do not know the decision of the students yet? And what if there are two courses with a very low probability that a student wants to take both of them? And what if you are running out of time slots? You might consider coloring both vertices with the same color and reduce the number of used colors:

Stochastic Graph Coloring



Changing settings

- set of edges random
- set of vertices random

Changing parameters

- allowed number of colors random
- "cost" of colors random

Changing settings

- set of edges random
- set of vertices random

Changing parameters

- allowed number of colors random
- "cost" of colors random

Vertex set random: You do not know which courses will take place in the end. For example a course might be cancelled due to lack of students.
Number of allowed colors random: The university might assign you a restricted number of time slots, that might change in the future due to changings in other programs.

Outline

1 2 examples of SCO problems
2 Modeling Stochastic Combinatorial Optimization Problems
- "Where" does the randomness occur?
  - Randomness occurs in the objective function f.
  - Randomness occurs in the constraint function g.
- When are the actual parameters revealed?
  - Parameters are revealed after decision has been made.
  - Parameters are revealed before corrective decision is made.
  - Parameters are revealed in several stages.

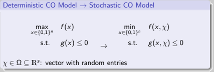
3 Solving Stochastic Combinatorial Optimization problems
- Problems/Difficulties
- Deterministic Reformulation
- Sample Average Approximation
- Metaheuristics for SCO problems

4 Conclusion
- Further Reading

Deterministic CO Model → Stochastic CO Model

$$\begin{aligned} \max_{x\in\{0,1\}^n} \quad & f(x) \\ \text{s.t.} \quad & g(x) \le 0 \end{aligned} \quad \rightarrow \quad \begin{aligned} \min_{x\in\{0,1\}^n} \quad & f(x,\chi) \\ \text{s.t.} \quad & g(x,\chi) \le 0 \end{aligned}$$

$\chi \in \Omega \subseteq \mathbb{R}^s$: vector with random entries



If you have an SCO optimization with random parameters and you fix these parameters, you get a deterministic CO problem.
The other way round, if you have a deterministic CO problem and you assume some of the parameters to be ranom your problem gets stochastic. Question: Where does the randomness occurs? Only in the objective, only in the constraint, in both?

Outline

1 2 examples of SCO problems

2 Modeling Stochastic Combinatorial Optimization Problems
- "Where" does the randomness occur?
  - Randomness occurs in the objective function f.
  - Randomness occurs in the constraint function g.
- When are the actual parameters revealed?
  - Parameters are revealed after decision has been made.
  - Parameters are revealed before corrective decision is made.
  - Parameters are revealed in several stages.

3 Solving Stochastic Combinatorial Optimization problems
- Problems/Difficulties
- Deterministic Reformulation
- Sample Average Approximation
- Metaheuristics for SCO problems

4 Conclusion
- Further Reading

Minimize an expectation

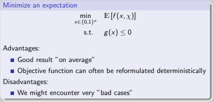
$$\min_{x \in \{0,1\}^n} \quad \mathbb{E}\left[f(x,\chi)\right]$$
$$\text{s.t.} \quad g(x) \leq 0$$

Advantages:

- Good result "on average"
- Objective function can often be reformulated deterministically

Disadvantages:

- We might encounter very "bad cases"



- $\mathbb{E}[X]$: expectation of random variable X

Minimize variance

$$\min_{x \in \{0,1\}^n} \quad Var\left[f(x,\chi)\right]$$
$$\text{s.t.} \quad g(x) \leq 0$$

Advantages:

- Outcome more concentrated around mean
- Possibility to reduce risk

Disadvantages:

- Makes not much sense without benchmark for expected costs

Minimize variance

$$\begin{aligned} \min_{x \in \{0,1\}^n} \quad & Var\left[f(x, \chi)\right] \\ \text{s.t.} \quad & g(x) \leq 0 \end{aligned}$$

Advantages:

- Outcome more concentrated around mean
- Possibility to reduce risk

Disadvantages:

- Makes not much sense without benchmark for expected costs

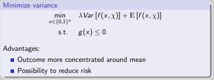
- $Var[X]$: variance of random variable X

Minimize variance

$$\min_{x \in \{0,1\}^n} \quad \lambda Var\left[f(x,\chi)\right] + \mathbb{E}\left[f(x,\chi)\right]$$
$$\text{s.t.} \quad g(x) \leq 0$$

Advantages:

- Outcome more concentrated around mean
- Possibility to reduce risk



Role of λ: Control relative importance of expectation and variance in your model.

Minimize variance

$$\min_{x \in \{0,1\}^n} \quad Var\left[f(x,\chi)\right]^{\lambda} \cdot \mathbb{E}\left[f(x,\chi)\right]$$
$$\text{s.t.} \quad g(x) \leq 0$$

Advantages:

- Outcome more concentrated around mean
- Possibility to reduce risk

Robust optimization

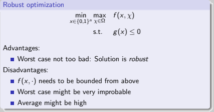
$$\min_{x\in\{0,1\}^n} \max_{\chi\in\Omega} \quad f(x,\chi)$$
$$\text{s.t.} \quad g(x) \leq 0$$

Advantages:

- Worst case not too bad: Solution is *robust*

Disadvantages:

- $f(x,\cdot)$ needs to be bounded from above
- Worst case might be very improbable
- Average might be high



- Robust Optmization generally not considered as being part of Stochastic Optmization, e.g. as the solution algorithms and approaches are generally quite different.
- However, I think the presented worst case model can be of good use in many cases, at least as a subproblem.
- Most common assumed distribution: χ_i (uniformly) distributed over certain interval

Outline

1. 2 examples of SCO problems
2. Modeling Stochastic Combinatorial Optimization Problems
   - "Where" does the randomness occur?
     - Randomness occurs in the objective function f.
     - Randomness occurs in the constraint function g.
   - When are the actual parameters revealed?
     - Parameters are revealed after decision has been made.
     - Parameters are revealed before corrective decision is made.
     - Parameters are revealed in several stages.
3. Solving Stochastic Combinatorial Optimization problems
   - Problems/Difficulties
   - Deterministic Reformulation
   - Sample Average Approximation
   - Metaheuristics for SCO problems
4. Conclusion
   - Further Reading

Worst case model

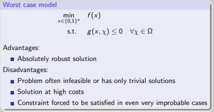
$$\min_{x \in \{0,1\}^n} \quad f(x)$$
$$\text{s.t.} \quad g(x,\chi) \leq 0 \quad \forall \chi \in \Omega$$

Advantages:

- Absolutely robust solution

Disadvantages:

- Problem often infeasible or has only trivial solutions
- Solution at high costs
- Constraint forced to be satisfied in even very improbable cases



- consider a knapsack problem with no upper bound on random item weights: only feasible solution would be to add no item at all.
- Worst case problem considered as robust optmization problem.

Chance-Constrained model

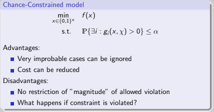
$$\min_{x \in \{0,1\}^n} \quad f(x)$$
$$\text{s.t.} \quad \mathbb{P}\{\exists i : g_i(x, \chi) > 0\} \le \alpha$$

Advantages:

- Very improbable cases can be ignored
- Cost can be reduced

Disadvantages:

- No restriction of "magnitude" of allowed violation
- What happens if constraint is violated?



$\mathbb{P}\{A\}$: probability that event A occurs

$$\mathbb{P}\{\exists i : g_i(x,\chi) > 0\} = 1 - \mathbb{P}\{g_i(x,\chi) \leq 0 \quad \forall i\}$$
$$\mathbb{P}\{\exists i : g_i(x,\chi) > 0\} \leq \alpha \Leftrightarrow \mathbb{P}\{g_i(x,\chi) \leq 0 \quad \forall i\} \geq 1 - \alpha$$

Simple-Recourse model

$$\min_{x \in \{0,1\}^{n_1}} \quad f(x) + \sum_{i=1}^{m} d_i \cdot \mathbb{E}\left[[g_i(x,\chi)]^+\right]$$

Advantages:

- Costs in case of violation taken into account
- "Magnitude" of violation can be controlled

Disadvantages:

- Probability of violation not restricted

Simple-Recourse model

$$\min_{x \in \{0,1\}^{n_1}} \quad f(x) + \sum_{i=1}^{m} d_i \cdot \mathbb{E}\left[[g_i(x,\chi)]^+\right]$$

Advantages:

- Costs in case of violation taken into account
- "Magnitude" of violation can be controlled

Disadvantages:

- Probability of violation not restricted

- $d_i > 0$ penalty per "unit of expected amount of violation"
- $[x]^+ = max(0, x)$
- $[g_i(x,\chi)]^+ = 0$ iff constraint i satisfied, $[g_i(x,\chi)]^+$ gives "amount" of violation otherwise
- $\mathbb{E}\left[[g_i(x,\chi)]^+\right]$: expected amount of violation
- Combine Simple Recourse and Chance constraint in order to control both the magnitude and probability of violation

Two-Stage model

$$\begin{aligned} \min_{x\in\{0,1\}^{n_1}} \quad & f(x) + \mathbb{E}\left[Q(x,\chi)\right] \\ \text{s.t.} \quad & Q(x,\chi) = \min_{y\in\{0,1\}^{n_2}} \quad \bar{f}(y) \\ & \qquad\qquad\qquad \text{s.t.} \quad\;\; g(x,y,\chi) \leq 0 \end{aligned}$$

Advantages:

- Violation of constraint not permitted
- Corrections in case of violation taken into account

Disadvantages:

- Problem extremely hard to solve:
  - $\rightarrow$ Non-convex, non-continuous objective function
  - $\rightarrow$ No closed-form expression of objective function
  - $\rightarrow$ Second-stage problem $\mathcal{NP}$-hard

Two-Stage model

$$\begin{aligned} \min_{x \in \{0,1\}^{n_1}} \quad & f(x) + \mathbb{E}\left[Q(x,\chi)\right] \\ \text{s.t.} \quad & Q(x,\chi) = \min_{y \in \{0,1\}^{n_2}} \quad \overline{f}(y) \\ & \qquad\qquad\qquad \text{s.t.} \quad g(x,y,\chi) \le 0 \end{aligned}$$

Advantages:

- Violation of constraint not permitted
- Corrections in case of violation taken into account

Disadvantages:

- Problem extremely hard to solve:
  - $\rightarrow$ Non-convex, non-continuous objective function
  - $\rightarrow$ No closed-form expression of objective function
  - $\rightarrow$ Second-stage problem $\mathcal{NP}$-hard

More general: First stage can of course have additional constraints.

Deterministic Knapsack Problem

$$\max_{x \in \{0,1\}^n} \quad \sum_{i=1}^{n} r_i x_i$$

$$\text{s.t.} \quad \sum_{i=1}^{n} w_i x_i \leq c$$

Simple Recourse Knapsack Problem

$$\max_{x \in \{0,1\}^n} \sum_{i=1}^{n} r_i x_i - d \cdot \mathbb{E}\left[[\sum_{i=1}^{n} \chi_i x_i - c]^+ \right]$$

Deterministic Knapsack Problem

$$\max_{x\in\{0,1\}^n} \quad \sum_{i=1}^{n} r_i x_i$$
$$\text{s.t.} \quad \sum_{i=1}^{n} w_i x_i \le c$$

Simple Recourse Knapsack Problem

$$\max_{x\in\{0,1\}^n} \sum_{i=1}^{n} r_i x_i - d \cdot \mathbb{E}\left[[\sum_{i=1}^{n} \chi_i x_i - c]^+\right]$$

- $[\sum_{i=1}^{n} \chi_i x_i - c]^+ =$ overweight
- $\mathbb{E}\left[[\sum_{i=1}^{n} \chi_i x_i - c]^+\right]$: expected overweight
- $d > 0$ penalty per overweight unit

Two-Stage Knapsack Problem

$$(TSKP) \quad \max_{x\in\{0,1\}^n} \quad \sum_{i=1}^{n} r_i x_i + \mathbb{E}[\mathcal{Q}(x,\chi)]$$

$$\text{s.t.} \quad \mathcal{Q}(x,\chi) = \max_{y^+,y^-\in\{0,1\}^n} \sum_{i=1}^{n} \overline{r}_i y_i^+ - \sum_{i=1}^{n} d_i y_i^-,$$

$$\text{s.t.} \quad y_j^+ \leq 1 - x_j, \quad j = 1,\dots,n,$$

$$y_j^- \leq x_j, \quad j = 1,\dots,n,$$

$$\sum_{i=1}^{n} (x_i + y_i^+ - y_i^-)\chi_i \leq c.$$

Two-Stage Knapsack Problem

$$
\begin{aligned}
(TSKP) \quad & \max_{x \in \{0,1\}^n} \quad \sum_{i=1}^{n} r_i x_i + \mathbb{E}[\mathcal{Q}(x,\chi)] \\
& \text{s.t.} \quad \mathcal{Q}(x,\chi) = \max_{y^+, y^- \in \{0,1\}^n} \sum_{i=1}^{n} \overline{r}_i y_i^+ - \sum_{i=1}^{n} d_i y_i^-, \\
& \qquad \text{s.t.} \; y_j^+ \leq 1 - x_j, \quad j = 1, \ldots, n, \\
& \qquad \quad y_j^- \leq x_j, \quad j = 1, \ldots, n, \\
& \qquad \quad \sum_{i=1}^{n} (x_i + y_i^+ - y_i^-) \chi_i \leq c.
\end{aligned}
$$

- Items can be added and/or removed in the second stage
- In the end remaining items need to respect knapsack capacity
- x: decision vector of 1st stage
- y^+, y^-: decision vectors of 2nd stage (recourse action)
- $\overline{r}_i < r_i$, $d_i > r_i$
- If $\overline{r}_i \geq r_i$: Add item i in 2. stage
- If $d_i \leq r_i$: Add item in 1. stage (removal is without cost or one even gains)

Outline

1 2 examples of SCO problems
2 Modeling Stochastic Combinatorial Optimization Problems
- "Where" does the randomness occur?
  - Randomness occurs in the objective function f.
  - Randomness occurs in the constraint function g.
- When are the actual parameters revealed?
  - Parameters are revealed after decision has been made.
  - Parameters are revealed before corrective decision is made.
  - Parameters are revealed in several stages.

3 Solving Stochastic Combinatorial Optimization problems
- Problems/Difficulties
- Deterministic Reformulation
- Sample Average Approximation
- Metaheuristics for SCO problems

4 Conclusion
- Further Reading

Static Stochastic Optimization problems

- Random parameters revealed after decision has been made.
- For decision maker parameters are revealed "once for all".
- No corrective decision can be made.

- Minimize expectation and/or variance
- Robust/Worst Case Optimization
- Chance-Constrained Optimization
- Simple Recourse Model

Static Stochastic Optimization problems

- Random parameters revealed after decision has been made.
- For decision maker parameters are revealed "once for all".
- No corrective decision can be made.

- Minimize expectation and/or variance
- Robust/Worst Case Optimization
- Chance-Constrained Optimization
- Simple Recourse Model

- Simple Recourse model: is paying a penalty a corrective decision? No.

Two-Stage Optimization problems

- For decision maker parameters are revealed "once for all".
- Random parameters revealed after first-stage decision has been made.
- Corrective decision can be made once the parameters are known.

- Two-Stage Model
- Simple Recourse Model

- Simple Recourse model: can be reformulated as Two-Stage decision model
- Continuous second stage decision variables y_i serve to "correct constraints"
- One variable for each constraint
- Second stage constraints: $g_i(x, \chi) \leq 0 + y_i$
- Optimal second-stage decision: $y_i = [g_i(x, \chi)]^+$

Multi-Stage Optimization problems

- Parameters are revealed in several stages.
- Corrective decision can be made in each stage.
- Which parameters are revealed in which stage generally defined.
- Decisions do only depend on already revealed parameters.

- Two-Stage Model
- Multi-Stage Model

Multi-Stage Optimization problems

- Parameters are revealed in several stages.
- Corrective decision can be made in each stage.
- Which parameters are revealed in which stage generally defined.
- Decisions do only depend on already revealed parameters.

- Two-Stage Model
- Multi-Stage Model

- Attention: In deterministic **multi-period** decision problems: future parameters all known
  decisions based on current parameters and future changings
  $\Rightarrow$ multi-period decision problems problems are deterministic problems

Outline

1 2 examples of SCO problems
2 Modeling Stochastic Combinatorial Optimization Problems
- "Where" does the randomness occur?
  - Randomness occurs in the objective function f.
  - Randomness occurs in the constraint function g.
- When are the actual parameters revealed?
  - Parameters are revealed after decision has been made.
  - Parameters are revealed before corrective decision is made.
  - Parameters are revealed in several stages.

3 Solving Stochastic Combinatorial Optimization problems
- Problems/Difficulties
- Deterministic Reformulation
- Sample Average Approximation
- Metaheuristics for SCO problems

4 Conclusion
- Further Reading

Outline

1 2 examples of SCO problems

2 Modeling Stochastic Combinatorial Optimization Problems
- "Where" does the randomness occur?
  - Randomness occurs in the objective function f.
  - Randomness occurs in the constraint function g.
- When are the actual parameters revealed?
  - Parameters are revealed after decision has been made.
  - Parameters are revealed before corrective decision is made.
  - Parameters are revealed in several stages.

3 Solving Stochastic Combinatorial Optimization problems
- **Problems/Difficulties**
- Deterministic Reformulation
- Sample Average Approximation
- Metaheuristics for SCO problems

4 Conclusion
- Further Reading

Structural Difficulties

- Non-convexity
- Non-continuous objective functions
- No closed-form expression for objective function

Two-Stage model

$$\begin{aligned} \min_{x \in \{0,1\}^{n_1}} \quad & f(x) + \mathbb{E}\left[Q(x,\chi)\right] \\ \text{s.t.} \quad & Q(x,\chi) = \min_{y \in \{0,1\}^{n_2}} \quad \bar{f}(y) \\ & \qquad\qquad\qquad \text{s.t.} \quad g(x,y,\chi) \leq 0 \end{aligned}$$

Structural Difficulties

- Non-convexity
- Non-continuous objective functions
- No closed-form expression for objective function

Two-Stage model

$$\begin{aligned} \min_{x \in \{0,1\}^{n_1}} \quad & f(x) + \mathbb{E}\left[Q(x,\chi)\right] \\ \text{s.t.} \quad & Q(x,\chi) = \min_{y \in \{0,1\}^{n_2}} \quad \bar{f}(y) \\ & \qquad\qquad\qquad \text{s.t.} \quad g(x,y,\chi) \leq 0 \end{aligned}$$

The Two-Stage model is an extrem example for the structural difficulties as generally all of them are present (at least in case of second stage integer or binary decision variables)

Computational Difficulties

- Expectations and probabilities = (multi-dimensional) integrals!
- Evaluating objective function might be $\mathcal{NP}$-hard
- High number of binary decision variables and constraints

Two-Stage Knapsack Problem

$$
\begin{aligned}
(TSKP) \quad & \max_{x\in\{0,1\}^n} \sum_{i=1}^{n} r_i x_i + \mathbb{E}[\mathcal{Q}(x,\chi)] \\
\text{s.t.} \quad & \mathcal{Q}(x,\chi) = \max_{y^+,y^-\in\{0,1\}^n} \sum_{i=1}^{n} \bar{r}_i y_i^+ - \sum_{i=1}^{n} d_i y_i^-, \\
& \text{s.t. } y_j^+ \le 1 - x_j, \quad j = 1,\dots,n, \\
& \qquad y_j^- \le x_j, \quad j = 1,\dots,n, \\
& \qquad \sum_{i=1}^{n} (x_i + y_i^+ - y_i^-)\chi_i \le c.
\end{aligned}
$$

Two-Stage Knapsack Problem

$$
\begin{aligned}
(TSKP) \quad & \max_{x\in\{0,1\}^n} \quad \sum_{i=1}^{n} r_i x_i + \mathbb{E}[\mathcal{Q}(x,\chi)] \\
& \text{s.t.} \quad \mathcal{Q}(x,\chi) = \max_{y^+,y^-\in\{0,1\}^n} \sum_{i=1}^{n} \bar{r}_i y_i^+ - \sum_{i=1}^{n} d_i y_i^-, \\
& \qquad \text{s.t.} \quad y_j^+ \leq 1 - x_j, \quad j = 1,\ldots,n, \\
& \qquad\qquad y_j^- \leq x_j, \quad j = 1,\ldots,n, \\
& \qquad\qquad \sum_{i=1}^{n} (x_i + y_i^+ - y_i^-)\chi_i \leq c.
\end{aligned}
$$

An example for $\mathcal{NP}$-hardness of the second-stage problem is the Two-Stage Knapsack problem, as the second stage problem can be shown to be a "simple" knapsack problem.

Outline

1 2 examples of SCO problems
2 Modeling Stochastic Combinatorial Optimization Problems
- "Where" does the randomness occur?
  - Randomness occurs in the objective function f.
  - Randomness occurs in the constraint function g.
- When are the actual parameters revealed?
  - Parameters are revealed after decision has been made.
  - Parameters are revealed before corrective decision is made.
  - Parameters are revealed in several stages.

3 Solving Stochastic Combinatorial Optimization problems
- Problems/Difficulties
- Deterministic Reformulation
- Sample Average Approximation
- Metaheuristics for SCO problems

4 Conclusion
- Further Reading

Idea

$\rightarrow$ Reformulate problem as a deterministic optimization problem

$\rightarrow$ Use already existing solvers to solve obtained problem

$\rightarrow$ Adapt existing algorithms to the special structure of the obtained problem

Problem

Generally only possible under assumption of special distributions!

Idea

- → Reformulate problem as a deterministic optimization problem
- → Use already existing solvers to solve obtained problem
- → Adapt existing algorithms to the special structure of the obtained problem

Problem

Generally only possible under assumption of special distributions!

Currently the most practiced approach to solve stochastic compbinatorial optimization problem. Unfortunately, as I think that if we tried to be a bit more innovative concerning the creation of special algorithms for SCO, we might advance faster.

Ex. 1: Chance-constrained Knapsack prob. (normal distribution)

$$\max_{x \in \{0,1\}^n} \quad \sum_{i=1}^{n} r_i x_i$$

$$\text{s.t.} \quad \mathbb{P}\{\sum_{i=1}^{n} \chi_i x_i > c\} \leq \alpha$$

Assume:

$\chi \sim \mathcal{N}(\mu, \Sigma)$ and $\alpha < 0.5 \Rightarrow$

$$\mathbb{P}\{\sum_{i=1}^{n} \chi_i x_i > c\} \leq \alpha \Leftrightarrow \underbrace{\sum_{i=1}^{n} x_i \mu^i + \Phi^{-1}(1-\alpha)\|\Sigma^{1/2} x\| \leq c}_{\text{Second Order Cone Constraint!}}$$

Ex. 1: Chance-constrained Knapsack prob. (normal distribution)

$$\max_{x \in \{0,1\}^n} \quad \sum_{i=1}^{n} r_i x_i$$
$$\text{s.t.} \quad \mathbb{P}\{\sum_{i=1}^{n} \chi_i x_i > c\} \leq \alpha$$

Assume:

$\chi \sim \mathcal{N}(\mu, \Sigma)$ and $\alpha < 0.5 \Rightarrow$

$$\mathbb{P}\{\sum_{i=1}^{n} \chi_i x_i > c\} \leq \alpha \Leftrightarrow \underbrace{\sum_{i=1}^{n} x_i \mu^i + \Phi^{-1}(1-\alpha)\|\Sigma^{1/2} x\| \leq c}_{\text{Second Order Cone Constraint!}}$$

- $\mathcal{N}(\mu, \Sigma)$: joint probability distribution for random vector χ
- $\mathcal{N}(\cdot, \cdot)$: (joint) normal distribution
- μ: vector of expectations of components of χ: $\mu_i = \mathbb{E}[\chi_i]$
- Σ: $n \times n$ covariance matrix
- here: Σ diagonal as weights assumed independently distributed
- Φ: cumulative distribution function of standard normal distribution
- values of Φ^{-1} can be looked up in tables
- $\alpha < 0.5$ needed for $\Phi^{-1}(1-\alpha)$ to be positive
- otherwise constraint not convex
- obtained constraint can be evaluated (one can check easily for feasibility)
- obtained problem type (Second Order Cone Problem) has been studied a lot and special algorithms have been proposed $\rightarrow$ no more "miracle" about how to solve the chance-constraint knapsack problem

Ex. 2: Chance-constrained Knapsack prob. (discrete distribution)

$$\max_{x\in\{0,1\}^n} \quad \sum_{i=1}^{n} r_i x_i \qquad \text{s.t.} \quad \mathbb{P}\{\sum_{i=1}^{n} \chi_i x_i > c\} \leq \alpha$$

Assume:

K outcomes $\chi^1, \dots, \chi^K$ with prob.'s $p^1, \dots, p^K$

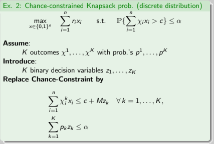
Introduce:

K binary decision variables $z_1, \dots, z_K$

Replace Chance-Constraint by

$$\sum_{i=1}^{n} \chi_i^k x_i \leq c + M z_k \quad \forall\, k = 1, \dots, K,$$

$$\sum_{k=1}^{K} p_k z_k \leq \alpha$$



- $M \geq \max_k (\sum_{i=1}^{n} \chi_i^k - c)$
- $z_k = 1$: scenario k is "ignored"
  corresponding constraint $\sum_{i=1}^{n} \chi_i^k x_i \leq c + M z_k$ always satisfied
- total probability of "ignored" scenarios must not exceed α

Outline

1 2 examples of SCO problems

2 Modeling Stochastic Combinatorial Optimization Problems
- "Where" does the randomness occur?
  - Randomness occurs in the objective function f.
  - Randomness occurs in the constraint function g.
- When are the actual parameters revealed?
  - Parameters are revealed after decision has been made.
  - Parameters are revealed before corrective decision is made.
  - Parameters are revealed in several stages.

3 Solving Stochastic Combinatorial Optimization problems
- Problems/Difficulties
- Deterministic Reformulation
- Sample Average Approximation
- Metaheuristics for SCO problems

4 Conclusion
- Further Reading

Idea of the SAA

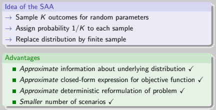
→ Sample K outcomes for random parameters

→ Assign probability $1/K$ to each sample

→ Replace distribution by finite sample

Advantages

- *Approximate* information about underlying distribution ✓
- *Approximate* closed-form expression for objective function ✓
- *Approximate* deterministic reformulation of problem ✓
- *Smaller* number of scenarios ✓



Note that SAA methods are not usable for robust optimization as in robust optimization we are concerned about the worst case. This cannot be reflected by working only on a sample.

Concerning "Smaller number of scenarios": Of course sample average approximation can also be used in case of a discrete probability distribution with a huge number of scenarios. If an approximation of the solution is all we need, a SAA with smaller sample might be solved instead.

Idea of the SAA

→ Sample K outcomes for random parameters

→ Assign probability $1/K$ to each sample

→ Replace distribution by finite sample

Disadvantages

- Solution of SAA might be infeasible for original problem
- Solution of SAA might be non-optimal for original problem
- To approximate original problem high number of samples might be needed

Simple Recourse Knapsack Problem

$$\max_{x\in\{0,1\}^n} \sum_{i=1}^{n} r_i x_i - d \cdot \mathbb{E}\left[[\sum_{i=1}^{n} \chi_i x_i - c]^+\right]$$

Sample Average Approximation:

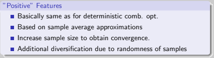
$$\max_{x\in\{0,1\}^n} \sum_{i=1}^{n} r_i x_i - d \cdot \sum_{k=1}^{K} \frac{1}{K}[\sum_{i=1}^{n} \chi_i^k x_i - c]^+$$

Outline

1. 2 examples of SCO problems
2. Modeling Stochastic Combinatorial Optimization Problems
   - "Where" does the randomness occur?
     - Randomness occurs in the objective function f.
     - Randomness occurs in the constraint function g.
   - When are the actual parameters revealed?
     - Parameters are revealed after decision has been made.
     - Parameters are revealed before corrective decision is made.
     - Parameters are revealed in several stages.
3. Solving Stochastic Combinatorial Optimization problems
   - Problems/Difficulties
   - Deterministic Reformulation
   - Sample Average Approximation
   - Metaheuristics for SCO problems
4. Conclusion
   - Further Reading

"Positive" Features

- Basically same as for deterministic comb. opt.
- Based on sample average approximations
- Increase sample size to obtain convergence.
- Additional diversification due to randomness of samples



Most metaheuristics for stochastic combinatorial optimisation work as follows: At the beginning of each iteration you draw a sample of the random parameters and create the corresponding SAA. The rest of the iteration this SAA is used to create new solutions, to compare their quality etc. In the next iteration, a new, slightly bigger sample is drawn etc..

Once more this approach is not possible for robust optimization (see remark in the SAA subsection).

Difficulties

- Evaluation of objective function expensive
- Comparison of quality of solutions difficult
- SCO problems generally have a lot of local optima
- Values of local optima can be close

Difficulties

- Evaluation of objective function expensive
- Comparison of quality of solutions difficult
- SCO problems generally have a lot of local optima
- Values of local optima can be close

In general you have to evaluate the objective function many times when applying a metaheuristic, in order to compare the quality of found solutions.

Outline

1 2 examples of SCO problems
2 Modeling Stochastic Combinatorial Optimization Problems
- "Where" does the randomness occur?
  - Randomness occurs in the objective function f.
  - Randomness occurs in the constraint function g.
- When are the actual parameters revealed?
  - Parameters are revealed after decision has been made.
  - Parameters are revealed before corrective decision is made.
  - Parameters are revealed in several stages.

3 Solving Stochastic Combinatorial Optimization problems
- Problems/Difficulties
- Deterministic Reformulation
- Sample Average Approximation
- Metaheuristics for SCO problems

4 Conclusion
- Further Reading

Summary

- Modeling SCO problems:
  - $\rightarrow$ Randomness in objective function?
  - $\rightarrow$ Randomness in constraint?
  - $\rightarrow$ Violation of constraint possible? Penalty?
  - $\rightarrow$ Correction of decision possible? How often?
- Solving SCO problems:
  - $\rightarrow$ Deterministic equivalent formulation
  - $\rightarrow$ Approximation of problem using sampling
  - $\rightarrow$ Meta-Heuristics

Greatest Challenges in Stochastic Combinatorial Optimization

- Modeling Combinatorial Real World problems with uncertainties as SCO problems
- Solve "realistic" sized problems in reasonable time
- Find more adapted solution techniques (for general distributions)
- Find efficient (Meta)Heuristics

Outline

1. 2 examples of SCO problems
2. Modeling Stochastic Combinatorial Optimization Problems
   - "Where" does the randomness occur?
     - Randomness occurs in the objective function f.
     - Randomness occurs in the constraint function g.
   - When are the actual parameters revealed?
     - Parameters are revealed after decision has been made.
     - Parameters are revealed before corrective decision is made.
     - Parameters are revealed in several stages.
3. Solving Stochastic Combinatorial Optimization problems
   - Problems/Difficulties
   - Deterministic Reformulation
   - Sample Average Approximation
   - Metaheuristics for SCO problems
4. Conclusion
   - Further Reading

Alexander Shapiro, Darinka Dentcheva, Andrzej Ruszczyński
Lectures on Stochastic Programming (2009)
pdf available online

András Prékopa
Stochastic Programming (1995).
Springer.

Anton J. Kleywegt, Alexander Shapiro, Tito Homem-de-Mello
The Sample Average Approximation Method for Stochastic Discrete Optimization (2002)
SIAM Journal on Optimization

Peter Kall, Stein W. Wallace
Stochastic Programming (1994)
pdf available online

Stochastic Programming Community
Stochastic Programming Community Home Page
`http://stoprog.org/`

Leonora Bianchi, Marco Dorigo, Luca Maria Gambardella, Walter J. Gutjahr
Metaheuristics in Stochastic Combinatorial Optimization: a Survey (2006)
pdf available online