

Introduction to algorithms for recourse models

Maarten H. van der Vlerk
University of Groningen



Tucson (AZ), October 9, 2004

Outline

- Introduction
 - recourse model
 - difficulties
- Special cases:
 - large-scale LP, SR, CR, multi-stage
 - motivation
 - properties
 - algorithms
 - example
- Concluding remarks
 - references

Recourse model (two stage)

Motivation:
LP (. . .) with random parameters

$$\begin{array}{ll} \min_x & cx \\ \text{s.t.} & Ax = b \quad m_1 \text{ det. constr.} \\ & T(\omega)x = h(\omega) \quad m_2 \text{ random constr.} \\ & x \geq 0 \quad x \in \mathbb{R}^{n_1} \end{array}$$

with ω random

Decision problem: choose x before ω known

Extend model: allow recourse actions to correct observed infeasibilities $h(\omega) - T(\omega)x$

Minimize costs: for x and ω fixed

$$Q(x, \xi(\omega)) := \min_y q(\omega)y \\ \text{s.t. } Wy = h(\omega) - T(\omega)x \\ y \in \mathbb{R}_+^{n_2}$$

with r -vector $\xi(\omega)$, known distribution
 $\xi(\omega) := (q(\omega), T_1(\omega), \dots, T_{m_2}(\omega), h(\omega))$

All in LP context

For every *first-stage* decision x

$$Q(x) := \mathbb{E}_\xi Q(x, \xi(\omega))$$

gives associated expected recourse costs

Two-stage linear recourse problem (DE)

$$\min \{cx + Q(x) : x \in X\} \quad (\text{RP})$$

where $X := \{x \in \mathbb{R}_+^{n_1} : Ax = b\}$

Choose $x \leftarrow \min$ total expected costs

- direct costs cx
- expected future costs $Q(x)$

Extensions:

- integer variables [Shabbir Ahmed]
- random recourse: $W(\omega)$
- non-linear
- risk: include e.g. $\text{var} Q(x, \xi(\omega))$ in objective [keynote lecture Rockafellar]
- chance constraints [Rene Henrion]
- multi-stage (later)

How to solve recourse problem RP

$$\min \{cx + Q(x) : x \in X\}$$

Observe

- X is convex set
 - Q is convex function (!)
- RP is CONVEX opt. problem (CP)

Theory for CP: KKT conditions, algorithms
→ well-solved ...

Why do we need special algorithms for RP ?

Consider e.g. descent algorithm for CP:

In each iteration t , need

- function evaluation $cx^t + Q(x^t)$
- first-order information “∇” $Q(x^t)$
- Hessian, ...

That is:

calculate $Q(x)$ and “∇” $Q(x)$ MANY TIMES

However

$$\begin{aligned} Q(x) &= \mathbb{E}_\xi Q(x, \xi(\omega)) \\ &= \mathbb{E}_\xi \min_{y \geq 0} \{q(\omega)y : Wy = h(\omega) - T(\omega)x\} \end{aligned}$$

One function evaluation $Q(x)$:

- solve 2nd-stage LP for all realizations ξ
- take expectation

If ξ discrete with

$$p^{i_1 i_2 \dots i_r} := \Pr \{\xi_1 = \xi_1^{i_1}, \xi_2 = \xi_2^{i_2}, \dots, \xi_r = \xi_r^{i_r}\}$$

then

$$Q(x) = \sum_{i_1} \sum_{i_2} \dots \sum_{i_r} p^{i_1 i_2 \dots i_r} Q(x, (\xi_1^{i_1}, \xi_2^{i_2}, \dots, \xi_r^{i_r}))$$

If ξ continuous with joint pdf $f(s)$, $s \in \mathbb{R}^r$

$$Q(x) = \int \dots \int Q(x, s) f(s) ds_1 ds_2 \dots ds_r$$

Remark: no closed form for $Q(x, s)$ in general

HOPELESS in general ...

(e.g. continuous, $r \geq 4$)

Remark: Similar in CC: evaluation $\Pr \{T(\omega)x \geq h(\omega)\}$

Consider SPECIAL CASES

- (small) discrete distribution → \mathbb{E}_ξ easy
- special structure 2nd-stage LP
→ $Q(x, \xi(\omega))$ easy

Use structural / mathematical properties to

- evaluate $Q(x)$ and “∇” $Q(x)$, many times
- optimize

Borrow from det. LP, CP, ...

Finite discrete distribution

Assume $\Xi = \{\xi^1, \xi^2, \dots, \xi^S\}$ with
 $\Pr \{\xi = \xi^s\} = p^s$, $s = 1, \dots, S$

Notation: $\xi^s = (q^s, T_1^s, \dots, T_{m_2}^s, h^s)$

Then \mathbb{E}_ξ easy (?): $Q(x) = \sum_{s=1}^S p^s Q(x, \xi^s)$

In this case

$$\begin{aligned} &\min \{cx + Q(x) : x \in X\} \\ &= \min \left\{ cx + \sum_{s=1}^S p^s Q(x, \xi^s) : Ax = b, x \geq 0 \right\} \\ &= \min \begin{array}{l} cx + p^1 q^1 y^1 + \dots + p^s q^s y^s \\ Ax = b \\ T^1 x + Wy^1 = h^1 \\ \vdots \\ T^S x + Wy^S = h^S \\ x \geq 0 \quad y^1 \geq 0 \quad \dots \quad y^S \geq 0 \end{array} \end{aligned}$$

RP is large-scale LP problem

Idea:

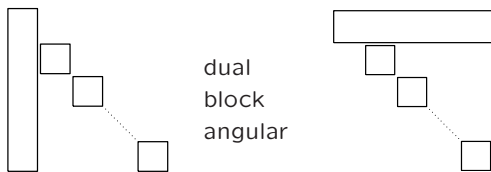
approximate continuous distr. by discrete
 → solve every RP by solving large-scale LP

In general too large: this LP has

- $n_1 + Sn_2$ variables
- $m_1 + Sm_2$ constraints

Example: $\xi = (\xi_1, \xi_2, \dots, \xi_9)$ independent,
 10 realizations each → $S = 10^9$

Observe: large-scale LP has special structure



- Exploit structure → decomposition (later)
- Solve by interior-point method
 (no details; see e.g. tutorial Birge (Berlin '01)
 on <http://stoprog.org>)

Easy 2nd-stage problem: Simple Recourse

Motivation: algorithmic

- How to use special structure / properties?
- Introduce solution approaches in easy setting

Simple Recourse (SR):

linear penalty costs for shortage & surplus
 w.r.t. each individual random constraint

Simplify:

- assume q and T fixed
- only rhs random: $h(\omega) =: h \in \mathbb{R}^{m_2}$

$$Q(x, h) = \min_{y \geq 0} q^+ y^+ + q^- y^-$$

$$\text{s.t. } y^+ - y^- = h - Tx$$

$$= \sum_{i=1}^{m_2} \min_{y_i \geq 0} q_i^+ y_i^+ + q_i^- y_i^-$$

$$\text{s.t. } y_i^+ - y_i^- = h_i - T_i x$$

separates: min problem for each constraint

Assuming $q^+ + q^- \geq 0 \Leftrightarrow Q(x, h) > -\infty$
 (sufficiently expensive recourse)

$$Q(x, h) = \sum_{i=1}^{m_2} v_i(z_i, h_i), \quad z := Tx \text{ tender var.}$$

where

$$v_i(z_i, h_i) := q_i^+ (h_i - z_i)^+ + q_i^- (h_i - z_i)^-$$

and $(s)^+ := \max\{0, s\}$, $(s)^- := \max\{0, -s\}$

Indeed, $Q(x, h)$ EASY: separable, closed form

Hence $Q(x) = \Psi(z)$, with $z := Tx$

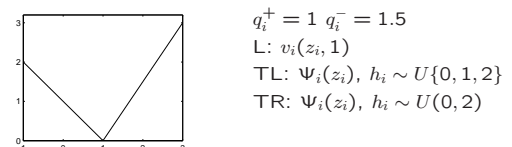
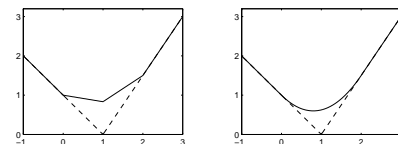
$$\Psi(z) = \sum_{i=1}^{m_2} \Psi_i(z_i)$$

where

$$\Psi_i(z_i) = q_i^+ \mathbb{E}_{h_i}(h_i - z_i)^+ + q_i^- \mathbb{E}_{h_i}(h_i - z_i)^-$$

is separable: each term Ψ_i

- depends only on z_i
- expectation wrt marginal distr. h_i : EASY



Straightforward analysis: properties of Ψ_i
 → properties Ψ and Q

$$\Psi_i(z_i) = q_i^+ (\mu_i - z_i) + (q_i^+ + q_i^-) \int_{-\infty}^{z_i} F_i(t) dt$$

with F_i cdf of $h_i \in H_i$ and $\mu_i = \mathbb{E}_{h_i} h_i$ (!)

Properties Ψ_i immediate from F_i

- convex
- finite iff $\mathbb{E}_{h_i} |h_i| < \infty$
- subdifferential $-q_i^+ + (q_i^+ + q_i^-) [\Pr\{h_i < z_i\}, F_i(z_i)]$
- piecewise linear if h_i discrete
- differentiable if h_i continuous
- asymptotes $v_i(z_i, \mu_i)$
- $z_i \notin \text{conv } H_i$: equal to asymptotes

SR problem

$$\min \left\{ cx + \sum_{i=1}^{m_2} \Psi_i(z_i) : x \in X, z = Tx \right\}$$

Algorithms for SR: use knowledge of $\mathcal{Q}(\Psi)$

- closed form
- properties

(A) Specific distribution \rightarrow formula for Ψ

Example: suppose each $h_i \sim \mathcal{E}(\lambda_i)$

$$\Psi_i(z_i) = \begin{cases} q_i^+(1/\lambda_i - z_i), & z_i \leq 0 \\ q_i^+(1/\lambda_i - z_i) + \\ (q_i^+ + q_i^-) \left(\frac{e^{-\lambda_i z_i} - 1}{\lambda_i} + z_i \right), & z_i \geq 0 \end{cases}$$

- convex
 - differentiable
- \rightarrow solve SR by favorite CP algorithm

(B) Finite discrete distribution [Wets]

Properties / formula:

\rightarrow SR equivalent to SMALL LP

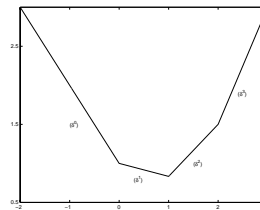
$h_i \sim F_i$, realizations $h_i^1 < h_i^2 < \dots < h_i^{S_i}$
w.p. p_i^s , $\mathbb{E}_{h_i} h_i = \mu_i$

Then Ψ_i is piecewise linear convex, with slope

$$(\Psi_i)'_{+}(z_i) = -q_i^+ + (q_i^+ + q_i^-) F_i(h_i^k) =: \delta_i^k$$

for $z_i \in [h_i^k, h_i^{k+1})$, $k = 0, 1, \dots, S_i$
(with $h_i^0 := -\infty$ and $h_i^{S_i+1} := +\infty$)

Observe: $-q_i^+ = \delta_i^0 < \delta_i^1 < \dots < \delta_i^{S_i} = q_i^-$



$q_i^+ = 1, q_i^- = 1.5$
 $h_i \sim \mathcal{U}\{0, 1, 2\}$

Hence (LP representation of PL function)

$$\begin{aligned} \Psi_i(z_i) = \min_u \quad & \delta_i u_i \equiv \sum_{k=0}^{S_i} \delta_i^k u_i^k \\ \text{s.t.} \quad & \sum_{k=0}^{S_i} u_i^k = z_i - \mu_i \\ & u_i \in U_i \begin{cases} u_i^0 \leq h_i^1 - \mu_i \\ 0 \leq u_i^k \leq h_i^{k+1} - h_i^k, \quad k = 1, \dots, S_i \end{cases} \end{aligned}$$

\rightarrow SR equivalent to SMALL LP

$$\begin{aligned} \min_{x,u} \quad & cx + \sum_{i=1}^{m_2} \delta_i u_i \\ \text{s.t.} \quad & Ax = b \\ & T_i x - \sum_{k=0}^{S_i} u_i^k = \mu_i, \quad i = 1, \dots, m_2 \\ & u_i \in U_i, \quad i = 1, \dots, m_2 \end{aligned}$$

- $n_1 + \sum_{i=1}^{m_2} (S_i + 1)$ variables
- $m_1 + m_2$ constraints
- $\sum_{i=1}^{m_2} S_i$ simple bounds

Indeed, small compared to "naive" LP

- $n_1 + 2m_2 \prod_{i=1}^{m_2} S_i$ variables
 - $m_1 + m_2 \prod_{i=1}^{m_2} S_i$ constraints
- assuming independence

Example: $h \in \mathbb{R}^9$ (independent),
10 realizations each

$$(n_1 + 99) \times (m_1 + 9) + 99 \text{ simple bounds}$$

versus

$$(n_1 + 18 \cdot 10^9) \times (m_1 + 9 \cdot 10^9)$$

Conclusion: exploit structure !

Other example: 2nd-stage network flow
[Wallace]

(C) SR continuous distribution (only rhs h)

Idea:

- construct discrete approximation of distribution
- solve as small LP
- approximate solution

Same approach for more general recourse
Introduce concepts / details in simple setting

Developing idea:

Would like “small” discrete approximation
→ “good” approximation of

- true optimal solution
- true optimal value

i.e. not necessarily of original distribution

Construct lower and upper bounds,
satisfied if gap is small enough . . .

Use properties of $\mathcal{Q} \leftarrow \Psi_i(z_i) := \mathbb{E}_{h_i} v_i(z_i, h_i)$

$$v_i(z_i, h_i) := q_i^+(h_i - z_i)^+ + q_i^-(h_i - z_i)^-$$

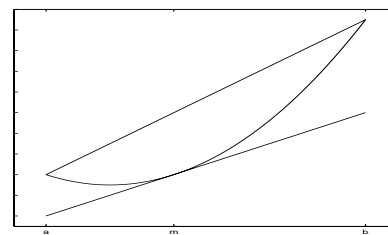
is finite and convex in h_i for fixed z_i

For any convex function $\varphi : \mathbb{R} \mapsto \mathbb{R}$
and random variable ω with

- support in $[a, b]$
- mean value μ

$$\begin{aligned} \varphi(\omega) &\geq \varphi(\mu) + u(\omega - \mu) && \text{(subgradient inequality)} \\ \varphi(\omega) &\leq \text{chord}(a, b), \quad \omega \in [a, b] && \text{(E-M ineq.)} \end{aligned}$$

with u a subgradient of φ at μ



Hence

$$\mathbb{E}_{\omega} \varphi(\omega) \geq \varphi(\mu) \quad \text{(Jensen ineq.)}$$

and

$$\mathbb{E}_{\omega} \varphi(\omega) \leq \frac{b - \mu}{b - a} \varphi(a) + \frac{\mu - a}{b - a} \varphi(b) \quad \text{(E-M ineq.)}$$

Apply to $\Psi_i(z_i) = \mathbb{E}_{h_i} v_i(z_i, h_i)$ assuming

- support h_i in $H_i := [a_i, b_i]$ (truncate)
- mean value \bar{h}_i

For any z_i , to obtain

- lower bound for $\Psi_i(z_i)$: replace h_i by η_i

$$\Pr \{ \eta_i = \bar{h}_i \} = 1$$

- upper bound for $\Psi_i(z_i)$: replace h_i by v_i

$$\Pr \{ v_i = a_i \} = \frac{b_i - \bar{h}_i}{b_i - a_i} \quad \Pr \{ v_i = b_i \} = \frac{\bar{h}_i - a_i}{b_i - a_i}$$

Indeed, “small” discrete approximations

However, bad approximation of Ψ_i . . .

Improve approximations:

- partition support $H_i = \bigcup_{j=1}^{n_i} H_i^j$
 $H_i^j := (s_i^{j-1}, s_i^j], a_i = s_i^0 < s_i^1 < \dots < s_i^{n_i} = b_i$
- apply approach to conditional distributions on each H_i^j

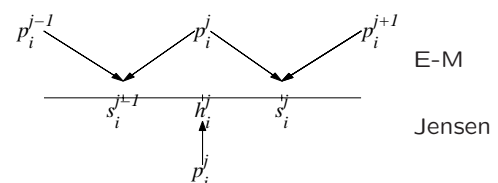
Result:

LB (Jensen): n_i -point discrete distribution

- on conditional means $\bar{h}_i^j \in H_i^j$
- $\Pr \{ \eta_i = \bar{h}_i^j \} = \Pr \{ h_i \in H_i^j \} =: p_i^j$

UB (E-M): $n_i + 1$ -point discrete distribution

- on endpoints $a_i = s_i^0, s_i^1, \dots, s_i^{n_i} = b_i$
- $\Pr \{ v_i = s_i^j \} = \frac{\bar{h}_i^j - s_i^{j-1}}{s_i^j - s_i^{j-1}} p_i^j + \frac{s_i^{j+1} - \bar{h}_i^{j+1}}{s_i^{j+1} - s_i^j} p_i^{j+1}$



Conceptual algorithm SR, continuous distr.

Iteration t :

current partitions $P_i^t, i = 1, \dots, m_2$

→ η^t (Jensen) and v^t (E-M)

- solve SR with rhs η^t (small LP: fast!)
→ solution x^t , LB^t on opt. value SR
- evaluate $cx^t + Q^t(x^t)$ with rhs v^t
→ UB^t on opt. value SR
- if $UB^t - LB^t \leq \epsilon^*$ STOP; $x^* = x^t$
- refine partitions → P_i^{t+1} ; REPEAT

Converges to optimal value / solution (...)

[Stability: see plenary lecture Römisch]

How to refine partition?

Several options:

(1) Brute force:

Split every cell in current partition

e.g. equal length or on conditional mean

→ # realizations η_{t+1} and v_{t+1} doubled

i.e. exponential growth # realizations !

(2) Adapt to problem:

Split cells which contribute most to gap
 $UB^t(x^t) - LB^t(x^t)$

e.g. split one cell in $H_{i,t}, i = 1, \dots, m_2$

→ linear growth # realizations

(3) Specific for SR with random rhs h

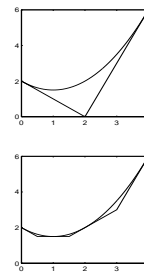
(uses $\Psi_i(z_i) = \text{asymptotes for } z_i \notin \text{conv } H_i$)

For $i = 1, \dots, m_2$

- split cell $\subset \text{conv } H_i$ containing $z_i^t = T_i x^t$ at z_i^t
- if no such cell exists $LB_i^t(x^t) = \Psi_i(T_i x^t)$
→ no update needed

Remark: (3) implies $LB^{t+1}(x^t) = Q(x^t)$

→ no need for UB for SR [Kall & Stoyan]



Function Ψ_i

$h_i \sim \mathcal{U}(0, 4)$

$q_i^+ = 1, q_i^- = 3$

“General” two-stage recourse problems

Recall two-stage recourse problem is

$$\min \{ cx + Q(x) : x \in X \}$$

where $X := \{x \geq 0 : Ax = b\}$,

$$Q(x) := \mathbb{E}_\xi Q(x, \xi(\omega))$$

and

$$Q(x, \xi(\omega)) := \min_{y \geq 0} \{ q(\omega)y : Wy = h(\omega) - T(\omega)x \}$$

Second-stage LP problem specified by

recourse structure $(q(\omega), W)$

e.g. SR: $W = (I_m - I_m)$ and $q(\omega) = (q^+ \ q^-)$

Next: algorithms for “general” recourse

→ conditions on $(q(\omega), W)$ such that
problem is “nice” for algorithms

(1) Simplify presentation: $q(\omega) = q$ fixed

(2) Avoid $Q(x, \xi(\omega)) = \pm\infty$

$Q(x, \xi(\omega)) = -\infty$: infinite reward for recourse

→ always exclude:

Dual feasible set $\{ \lambda \in \mathbb{R}^{m_2} : \lambda W \leq q \} \neq \emptyset$

Called *sufficiently expensive recourse*

$Q(x, \xi(\omega)) = +\infty$: no recourse action possible

Interpretation OK; inconvenient in algorithms

→ exclude (for now):

$\text{pos } W := \{ t : \exists y \geq 0, Wy = t \} = \mathbb{R}^{m_2}$

Called *complete recourse*

(3) $Q(x)$ finite:

Given SER & CR → $Q(x, \xi(\omega))$ finite,
depends on distribution $\xi(\omega)$.

Since q is fixed, sufficient condition is

$$\mathbb{E}_\xi |\xi(\omega)| < \infty \text{ (componentwise)}$$

Algorithms tailored to properties RP
 → study properties of $Q(x, \xi(\omega))$ and $Q(x)$

SER & CR → $Q(x, \xi(\omega))$ finite.
 Hence, by LP duality

$$Q(x, \xi(\omega)) := \min_{y \geq 0} \{qy : Wy = h(\omega) - T(\omega)x\}$$

$$= \max_{\lambda \in \mathbb{R}^{m/2}} \{\lambda (h(\omega) - T(\omega)x) : \lambda W \leq q\}$$

Dual set is polyhedral, non-empty, bounded
 → finitely many extreme points $\lambda^1, \lambda^2, \dots, \lambda^K$

$$Q(x, \xi(\omega)) = \max_{k=1, \dots, K} \lambda^k (h(\omega) - T(\omega)x)$$

$Q(x, \xi)$ is pointwise maximum of
 finitely many linear functions

→ for fixed ξ ,
 $Q(x, \xi)$ is polyhedral convex function of x

Hence $Q(x)$ is convex

- polyhedral if ξ finite discrete
- differentiable if ξ continuous (...)

Algorithms for CR model, ξ discrete

Assume ξ discrete: for $s = 1, \dots, S$
 $\xi^s = (h^s, T^s)$ with $p^s := \Pr\{\xi = \xi^s\}$

Equivalent large-scale LP

$$\min cx + \sum_{s=1}^S p^s qy^s : Ax = b$$

$$T^s x + Wy^s = h^s \quad \forall s$$

$$x \geq 0, \quad y^s \geq 0 \quad \forall s$$

Special structure: L-shaped, ...
 → decomposition, algorithms

$Q(x)$ polyhedral: pointwise max of
 finitely many linear functions:
 supporting hyperplanes of $\text{epi } Q$

$$\text{RP} = \min_{x \in X, \theta \in \mathbb{R}} \{cx + \theta : \theta \geq E^j x + e^j, j \in J\}$$

Algorithmic perspective:

- How to construct *optimality cuts*
 $\theta \geq E^j x + e^j$?
- $|J|$ too large → don't add all ...

Optimality cuts: outer linearization of Q

Recall

$$Q(\bar{x}, \xi^s) = \max_{k=1, \dots, K} \lambda^k (h^s - T^s \bar{x})$$

→ opt. dual solution $\lambda^{k(\bar{x}, s)}$:
 $-\lambda^{k(\bar{x}, s)} T^s$ subgradient of $Q(\cdot, \xi^s)$ at \bar{x}

Hence

$$Q(\bar{x}) = \sum_{s=1}^S p^s \lambda^{k(\bar{x}, s)} (h^s - T^s \bar{x})$$

$$= E(\bar{x}) \bar{x} + e(\bar{x})$$

where

$$E(\bar{x}) := - \sum_{s=1}^S p^s \lambda^{k(\bar{x}, s)} T^s \in \partial Q(\bar{x})$$

$$e(\bar{x}) := \sum_{s=1}^S p^s \lambda^{k(\bar{x}, s)} h^s$$

so that

$$\theta \geq E(\bar{x})x + e(\bar{x})$$

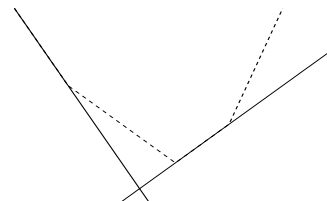
is an optimality cut at \bar{x}

That is: Compute $Q(\bar{x})$ by solving
 DUAL of 2nd-stage problems, $s = 1, \dots, S$
 → $Q(\bar{x})$ and optimality cut, sharp at \bar{x}

Remark: dual feasible set is $\{\lambda : \lambda W \leq q\}$
 → hot starts, bunching, ...

Opt. cuts → outer linearization of Q
 Hence

- subset of all cuts → lower bound of Q
- add cut at \bar{x} → approximation sharp at \bar{x}



L-shaped algorithm for RP [Van Slyke & Wets]:
add cuts iteratively

Iteration ν :

(i) Solve MASTER LP problem

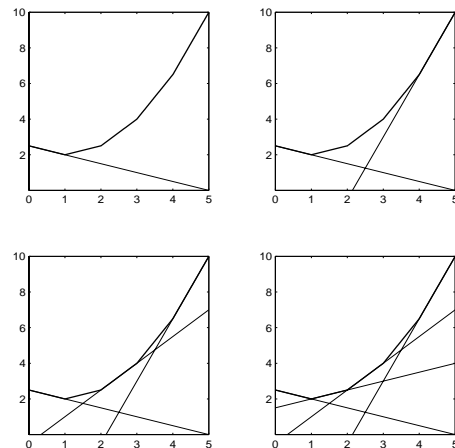
$$\min_{\substack{x \in X \\ \theta \in \mathbb{R}}} \{cx + \theta : \theta \geq E^j x + e^j, j = 1, \dots, \nu - 1\}$$

→ current optimal solution (x^ν, θ^ν)

(ii) Solve S dual 2nd-stage problems
with x^ν and $\xi^s, s = 1, \dots, S$

→ $Q(x^\nu) = E^\nu x^\nu + e^\nu$ and opt. cut at x^ν

(iii) If $\theta^\nu \geq Q(x^\nu)$ STOP: x^ν is optimal (!)
Otherwise: iteration $\nu + 1$



L-shaped algorithm applied to SR problem

$$\min\{x/2 + Q(x) : 0 \leq x \leq 5\}$$

with $q^+ = 1, q^- = 3,$

$T(\omega) = 1$ fixed, and $h(\omega) \sim U\{1, 2, 3, 4\}$

What if recourse is not complete,
i.e. $Q(x, \xi) = +\infty$ is possible?

Define

$$K := \{x \in \mathbb{R}^{n_1} : Q(x, \xi^s) < +\infty, s = 1, \dots, S\}$$

$$= \{x \in \mathbb{R}^{n_1} : Q(x) < +\infty\}$$

K is a polyhedral convex set (...)

Very many facets in general

Remark: If $X \subset K$ don't bother...

Known as *relatively complete recourse*
(hard to verify)

Obviously, if $\bar{x} \notin K$ then \bar{x} not optimal
→ algorithmic approach: iterate,
cut off such \bar{x} by a linear *feasibility cut*

In L-shaped algorithm: iteration ν

(i) Solve master problem

$$\min_{\substack{x \in X, \theta \in \mathbb{R}}} cx + \theta$$

$$\text{s.t. } D^l x \geq d^l, \quad l = 1, \dots, m$$

$$\theta \geq E^j x + e^j, \quad j = 1, \dots, n$$

→ optimal solution (x^ν, θ^ν)

(ii') Solve auxiliary problems (CR):

for $s = 1, \dots, S$, with $e := (1 \ 1 \dots 1)$

$$w(x^\nu, \xi^s) := \min e(y^+ + y^-)$$

$$\text{s.t. } Wy + y^+ - y^- = h^s - T^s x^\nu$$

$$y \geq 0, y^+ \geq 0, y^- \geq 0$$

$$= \max\{\sigma(h^s - T^s x^\nu) : \sigma W \leq 0, |\sigma| \leq e\}$$

→ optimal dual solution $\sigma^{\nu s}$

$x^\nu \notin K$ iff $w(x^\nu, \xi^s) > 0$ for some s

In that case

$$\sigma^{\nu s}(h^s - T^s x) \begin{cases} \leq 0 & \forall x : Q(x) < +\infty \\ > 0, & x = x^\nu \end{cases}$$

→ feasibility cut $D^{m+1} x \geq d^{m+1}$

Return to step (i) with $m \leftarrow m + 1$

Otherwise proceed with (ii):

generate optimality cut

Variants

Multi-cut L-shaped [Birge & Louveaux]

Generate opt. cuts for each $Q(x, \xi^s)$ instead of $Q(x)$ (aggregated)

- more detailed information to master
→ less iterations expected
- larger master problem
- Numerical evidence:
faster if $S \gg m_1$ (1st-stage constr.)

Regularized decomposition [Ruszczynski]

Multi-cut

NLP: Regularizing term $\|x - a^v\|^2$, with a^v incumbent solution, in objective to overcome

- inefficient initial iterations
- degeneracy during later iterations
→ keep at most $n_1 + S$ constraints

(Implementation: QDECOM)

- Numerical evidence:
outperforms L-shaped and Multi-cut

L-shaped: *time* decomposition large-scale LP

$$\min cx + \sum_{s=1}^S p^s qy^s : \begin{array}{l} Ax = b \\ T^s x + W y^s = h^s \quad \forall s \\ x \geq 0, \quad y^s \geq 0 \quad \forall s \end{array}$$

Indeed: 1st-stage variables x fixed
→ $Q(x)$: decomposed in S small LPs
→ LB model of $Q(x)$ by cutting planes

Similar approach for MIP with $x \in \mathbb{Z}^n$, $y \in \mathbb{R}^m$
known as Benders Decomposition

Alternative: decomposition by *scenarios* ξ^s :
Introduce copies x^s of x , $s = 1, \dots, S$

$$\min \sum_{s=1}^S p^s (cx^s + qy^s) : \left. \begin{array}{l} Ax^s = b \\ T^s x^s + W y^s = h^s \\ x^s \geq 0, \quad y^s \geq 0 \end{array} \right\} \forall s \\ x^1 = x^2 = \dots = x^S$$

Relax *non-anticipativity constraints*
→ decomposition in S *scenario problems*

More details: multi-stage problems (later)

Algorithms for CR with continuous ξ

(A) Conceptual algorithm:

- approximate using discrete distributions
- solve by e.g. L-shaped
- stop if approximation good enough,
else: refine discrete approximations, repeat

Iteration t :

Current partition P^t of $\Xi \subset \mathbb{R}^r$
(rectangular cells)

- Jensen → η^t on conditional means
E-M → v^t on 2^r vertices of cells
- Solve RP with η^t → x^t and LB^t
Evaluate x^t using v^t → UB^t
- If $UB^t - LB^t \leq \epsilon^*$ STOP: solution x^t
else: update partition P^t ; iteration $t + 1$

How to update partition ?

- E-M: evaluate $Q(x^t, \cdot)$ at 2^r vertices of each cell
- “Bad” choice never undone:
extra work in iterations $t + 1, \dots$
→ Invest (CPU) time in “good” update ...

Current partition P^t : cells Ξ_1, \dots, Ξ_K

Which cell(s) to partition ?

Choose Ξ_k with largest gap $UB_k - LB_k$

$$LB_k := p_k Q(x^t, \mu_k) \quad \begin{array}{c} s^{k4} \quad s^{k3} \\ \square \\ \mu_k \\ s^{k1} \quad s^{k2} \end{array}$$

$$UB_k := \sum_{i=1}^{2^r} p_{ki} Q(x^t, s^{ki})$$

or all Ξ_k with $UB_k - LB_k > \epsilon^t$

(# cells ↔ # iterations: solve by L-shaped)

How to partition a cell Ξ_k ?

- Choice of coordinate ξ_j , $j = 1, \dots, r$
- Where ?
e.g. equal size, at $\mu_{kj} := \mathbb{E}_{\xi} \xi_j | \xi \in \Xi_k$

Many heuristic choices possible ...

The following works well in practice:

- Try each partition at μ_{kj} , $j = 1, \dots, r$
- Apply Jensen & E-M → gap $UB_{kj} - LB_{kj}$
- Keep partition which minimizes gap

Lot of work !

Each iteration of “discrete approximation” expensive

- update discrete approximations
- solve RP^t by L-shaped
→ exact LB^t and UB^t (also for t small)

Hopefully few iterations needed ...

Alternative: use “cheap iterations”

- sample from distribution ξ
- fast approximation $Q(x^t)$
→ opt. cut, LB in expectation
(better cuts for t large)

Willing to use many such iterations ...

Stochastic Decomposition [Higle & Sen]

Assume CR, q fixed, $Q(x) \geq 0$ (known LB)

Iteration t of SD:

- solve master LP → opt. solution (x^t, θ^t)
- sample $\xi^t = (T^t, h^t)$
- update optimality cuts:

(i) Generate new cut at x^t :

Calculate $Q(x^t, \xi^t)$, i.e. solve 1 dual problem

$$\max \{ \lambda (h^t - T^t x) : \lambda W \leq q \} \rightarrow \lambda_t^t$$

Set $\Lambda^t = \Lambda^{t-1} \cup \{ \lambda_t^t \}$ (known dual vertices)

Calculate lower bound for $Q(x^t, \xi^s)$,

$s = 1, \dots, t-1$

$$\max \{ \lambda (h^s - T^s x) : \lambda \in \Lambda^t \} \rightarrow \lambda_s^t \quad \text{FAST!}$$

New cut at x^t :

$$\theta \geq \frac{1}{t} \sum_{s=1}^t \lambda_s^t (h^s - T^s x) =: E_t^t x + e_t^t$$

(ii) Update old cuts:

- expected to be loose (based on Λ^{t-1})
- possibly invalid (sample)

For $s = 1, \dots, t-1$

$$\theta \geq \frac{t-1}{t} (E_s^{t-1} x + e_s^{t-1}) =: E_s^t x + e_s^t$$

Old cuts → 0 as # iterations t increases

Problem: $\{x^t\} \not\rightarrow x^*$

Solution: track incumbent solutions $\{\bar{x}_s^t\}$

- new incumbent only if x^t “really better” than \bar{x}_s^t
- update cut at incumbent \bar{x}_s^t using (i)

Identifiable subsequence of $\{\bar{x}_s^t\} \rightarrow x^*$

Stopping criterion: if for some time

- change in objective small AND
- no new dual vertices AND
- incumbent not changed

Other sampling methods

- Importance sampling [Dantzig, Infanger]
- External sampling [Shapiro et al]

Multi-stage recourse

Usual H -stage LP structure:

$$\begin{array}{llll} \min & c^1 x^1 + c^2 x^2 + & \dots & c^{H-1} x^{H-1} + c^H x^H \\ & W^1 x^1 & & = h^1 \\ & T^2 x^1 + W^2 x^2 & & = h^2 \\ & & & T^3 x^2 + W^3 x^3 & = h^2 \\ & \vdots & \ddots & & \\ & & & T^{H-1} x^{H-2} + W^{H-1} x^{H-1} & = h^{H-1} \\ & & & & T^H x^{H-1} + W^H x^H & = h^H \\ & x^1 \geq 0 & x^2 \geq 0 & \dots & & x^H \geq 0 \end{array}$$

Stage t :

- decision vector x^t
- parameters: W^t and $\xi^t = (c^t, T^t, h^t)$
- separated constraints: $x^t \geq 0$ (...)

Constraint matrix has STAIRCASE structure:

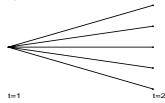
coupling between neighboring stages only

Multi-stage recourse: information structure

Recall: two-stage recourse

Decision – Observation – Decision

Represent discrete distribution by "tree":



Decisions: solve (N)LP in each node of tree

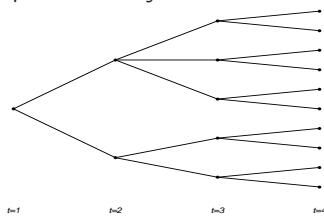
Multi-stage recourse:

Uncertainty revealed at different moments

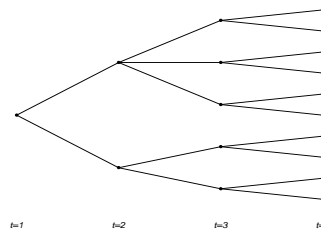
D – O – D – ... – O – D

Assume discrete distributions at $t = 2, \dots, H$

Represented by *scenario tree*



Again: decision x_k^t in every node (t, k) ,
 $t = 1, \dots, H, k = 1, \dots, K^t$



At every node (t, k) ($1 < t < H$)

- observation $\xi_k^t = (c_k^t, T_k^t, h_k^t)$
- solve NLP

$$\min c_k^t x_k^t + Q_k^t(x_k^t)$$

$$\text{s.t. } W^t x_k^t = h_k^t - T_k^t x_{a(k)}^{t-1}$$

$$x_k^t \geq 0$$

with $Q_k^t(x_k^t)$ expected future costs

and $(t-1, a(k))$ unique *ancestor* node of (t, k)

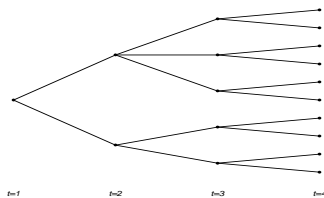
→ connections between nodes !

Observations (cf. two-stage):

- MSRP is a large-scale LP (HUGE)
- each Q_k^t is a polyhedral function

Solvability: size of scenario tree

- Scenario generation [Georg Pflug]



Consider a node (t, k) ($1 < t < H$)
 "myopic view"

- for ancestor node $(t-1, a(k))$
 (t, k) is a second-stage problem
- for every *child* node $(t+1, j), j \in D^t(k)$
 (t, k) is the first-stage problem

→ MSRP consists of
 nested two-stage problems

Algorithm: Nested L-shaped/Benders [Birge]

Master problem LP(t, k) at every node

$$\min_{x_k^t \geq 0, \theta_k^t} c_k^t x_k^t + \theta_k^t$$

$$\text{s.t. } W^t x_k^t = h_k^t - T_k^t x_{a(k)}^{t-1}$$

$$E_{k,j}^t x_k^t + \theta_k^t \geq e_{k,j}^t, \quad j = 1, \dots, n_k^t$$

(feasibility cuts)

send information through tree (\longleftrightarrow)

Iteration of Nested L-shaped / Benders

FORWARD:

for $t = 1$ to H

for $k = 1$ to K^t

solve master LP(t, k)

→ x_k^t, θ_k^t and multipliers λ_k^t : recourse constr.
 σ_k^t : opt. cuts

end

end

BACK:

for $t = H-1$ to 1

for $k = 1$ to K^t

$$E_k^t = E_k^t(\lambda_j^{t+1}, j \in D^{t+1}(k))$$

$$e_k^t = e_k^t(\lambda_j^{t+1}, \sigma_j^{t+1}, j \in D^{t+1}(k))$$

$$\bar{\theta}_k^t = e_k^t - E_k^t x_k^t \quad (\text{current approx. } Q_k^t(x_k^t))$$

if $\bar{\theta}_k^t > \theta_k^t$ then

add opt. cut (E_k^t, e_k^t) to LP(t, k)

else

if $t = 1$ STOP: x_1^1 optimal

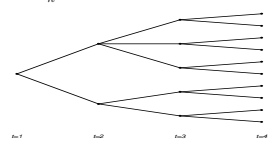
end

end

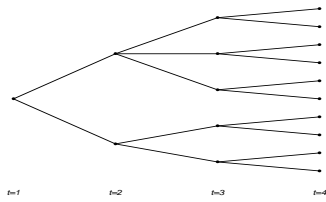
end

Variants:

- feasibility cuts for non-CR
- various strategies Forward/Back



Alternative approach: scenario decomposition



For each scenario

$$\xi_s := (\xi_s^1, \xi_s^2, \dots, \xi_s^H), \quad s = 1, \dots, S := K^H$$

define the scenario problem

$$\min f_s(x)$$

with $x := (x^1, x^2, \dots, x^H)$ and

$$f_s(x) := \begin{cases} \sum_{t=1}^H c_s^t x^t, & x \in X(\xi_s) \\ +\infty, & \text{otherwise} \end{cases}$$

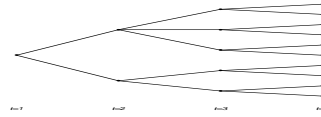
Observe: LP problem (relatively small)

Then MSRP is

$$\min_x \sum_{s=1}^S p_s f_s(x_s) : \text{non-anticipativity constr.}$$

NAC: For $t = 1, \dots, H - 1$

$$x_s^t = x_\sigma^t \text{ for all } s, \sigma: \xi_s^{[1,t]} = \xi_\sigma^{[1,t]}$$



Various representations of NAC, e.g.

- $Gx = 0$ with G sparse
- for all s and t : $x_s^{[1,t]} = \mathbb{E}\{x^{[1,t]} | \xi_s^{[1,t]}\}$

Lagrangian Methods

(Progressive Hedging [Rockafellar & Wets])

Relax non-anticipativity constraints

Iterate:

- solve Lagrangian scenario subproblems (including regularizing term)
- update multipliers

Advantage: problem structure intact

Concluding remarks

SP algorithms for recourse problems

- are necessary
- use structure / properties
- allow solving realistic problems

This tutorial

- outline of main algorithmic concepts
- many advanced ideas not covered

Challenges: many, including

- multi-stage
- mixed-integer

Textbooks:

J.R. Birge and F.V. Louveaux. *Introduction to Stochastic Programming*. Springer, 1997.

Yu. Ermoliev and R.J-B. Wets. *Numerical Techniques for Stochastic Optimization*. Springer, Berlin, 1988.

P. Kall and S.W. Wallace. *Stochastic Programming*. Wiley, 1994. Also available as PDF file, e.g. via <http://stoprog.org>.

A. Prékopa. *Stochastic Programming*. Kluwer, 1995.

A. Ruszczyński and A. Shapiro, editors. *Stochastic Programming*, vol. 10 of *Handbooks in OR and MS*. North-Holland, 2003.

References:

J.F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4:238–252, 1962.

J.R. Birge. Decomposition and partitioning methods for multistage stochastic linear programs. *Oper. Res.*, 33(5):989–1007, 1985.

J.R. Birge. Stochastic programming computation and applications, *INFORMS Journal on Computing* 9(2): 111–133, 1997. (PDF via <http://stoprog.org>)

J.R. Birge and F.V. Louveaux. A multicut algorithm for two-stage stochastic linear programs. *European Journal of Operations Research*, 34:384–392, 1988.

G.B. Dantzig and G. Infanger. Large-scale stochastic linear programs—importance sampling and Benders decomposition. In C. Brezinski and U. Kulisch, editors, *Computational and applied mathematics, I (Dublin, 1991)*, pages 111–120. North-Holland, Amsterdam, 1992.

K. Frauendorfer. *Stochastic two-stage programming*, volume 392 of *Lecture Notes in Economics and Mathematical Systems*. Springer-Verlag, Berlin, 1992.

J.L. Hight and S. Sen. Stochastic decomposition: An algorithm for two-stage linear programs with recourse. *Math. Oper. Res.*, 16:650–669, 1991.

G. Infanger. Monte Carlo (importance) sampling within a Benders decomposition algorithm for stochastic linear programs. *Ann. Oper. Res.*, 39(1-4):69–95 (1993), 1992.

P. Kall and D. Stoyan. Solving stochastic programming problems with recourse including error bounds. *Math. Operationsforsch. Statist. Ser. Optim.*, 13(3):431–447, 1982.

J. T. Linderoth, A. Shapiro, and S. J. Wright. The empirical behavior of sampling methods for stochastic programming. Optimization Online, <http://www.optimization-online.org>, 2002.

J. Mayer. *Stochastic Linear Programming Algorithms: A Comparison Based on a Model Management System*. Gordon and Breach, Amsterdam, 1998.

M.V.F. Pereira and L.M.V.G Pinto. Stochastic optimization of a multireservoir hydroelectric system—a decomposition approach. *Water Resources Research*, 21(6):779–792, 1985.

A. Ruszczyński. A regularized decomposition method for minimizing a sum of polyhedral functions. *Math. Program.*, 35:309–333, 1986.

A. Shapiro. Statistical inference of multistage stochastic programming problems. Optimization Online, <http://www.optimization-online.org>, 2002.

R. Van Slyke and R.J-B. Wets. L-shaped linear programs with applications to control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17:638–663, 1969.

S.W. Wallace. Solving stochastic programs with network recourse. *Networks*, 16(3):295–317, 1986.

R.J-B. Wets. Solving stochastic programs with simple recourse. *Stochastics*, 10:219–242, 1984.

Stochastic Programming Bibliography,
<http://mally.eco.rug.nl/spbib.html>