Stochastic Dual Dynamic Programming Algorithm for Multistage Stochastic Programming

Final presentation – ISyE 8813 Fall 2011

Guido Lagos   Wajdi Tekaya

Georgia Institute of Technology

November 30, 2011

Multistage Stochastic Linear Programming

Analysis of Stochastic Dual Dynamic Programming Method
Alex Shapiro (2010)

$$\min_{\substack{A_1 x_1 = b_1 \\ x_1 \geq 0}} c_1^T x_1 + \mathbb{E}\left[\min_{\substack{B_2 x_1 + A_2 x_2 = b_2 \\ x_2 \geq 0}} c_2^T x_2 + \mathbb{E}\left[\quad \ldots \quad + \mathbb{E}\left[\min_{\substack{B_T x_{T-1} + A_T x_T = b_T \\ x_T \geq 0}} c_T^T x_T \right] \ldots \right]\right]$$

- Applications: planning problems in mining, energy, forestry, etc.

Challenges:

- Tractability of $\mathbb{E}$
- **Stagewise dependence of data process** $\{\xi_t := (c_t, B_t, A_t, b_t)\}_{t=1,\ldots,T}$
- **Curse of dimensionality**

Summary

1. Two-stage stochastic programming
   - Cutting-plane method
   - SDDP algorithm for two-stage SP
2. Multistage stochastic programming
   - SDDP algorithm for multistage SP
   - Convergence and main contributions

1. Two-stage stochastic programming
   - Cutting-plane method
   - SDDP algorithm for two-stage SP

2. Multistage stochastic programming
   - SDDP algorithm for multistage SP
   - Convergence and main contributions

"True" problem:

$$\min_{\substack{A\mathbf{x}=b\\ \mathbf{x}\geq 0}} c^{\mathsf{T}}\mathbf{x} + \mathbb{E}\left[\min_{\substack{T\mathbf{x}+W\mathbf{y}=h\\ \mathbf{y}\geq 0}} q^{\mathsf{T}}\mathbf{y}\right]$$

Take random sample $\xi^1, \ldots, \xi^N$ and approximate $\mathbb{E} \sim \frac{1}{N}\sum_{j=1}^N$
$\Longrightarrow$ **Sample Average Approximation (SAA) problem**:

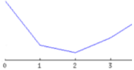
$$\min_{\substack{A\mathbf{x}=b\\ \mathbf{x}\geq 0}} c^{\mathsf{T}}\mathbf{x} + \underbrace{\frac{1}{N}\sum_{j=1}^{N} \underbrace{\left[\min_{\substack{T^j\mathbf{x}+W^j\mathbf{y}=h^j\\ \mathbf{y}\geq 0}} q^{j\mathsf{T}}\mathbf{y}\right]}_{Q(\mathbf{x},\xi^j):=}}_{\widetilde{\mathcal{Q}}(\mathbf{x}):=}$$

Basic Ingredients

$$\min_{\substack{Ax=b\\x\geq 0}} c^{\mathsf T}x + \underbrace{\frac{1}{N}\sum_{j=1}^{N} Q(x,\xi^j)}_{\widetilde{\mathcal{Q}}(x):=} \qquad \text{where} \qquad Q(x,\xi) := \max_{\substack{Tx+Wy=h\\y\geq 0}} q^{\mathsf T}y$$

Assume: relatively complete recourse, i.e. $\forall$ feasible x, $Q(x,\xi) < \infty$ a.s.

$\Rightarrow$ $\frac{1}{N}\sum_{j=1}^{N} Q(\cdot,\xi^j)$ convex piecewise-linear, and problem is

$\min_{\substack{Ax=b\\x\geq 0}} c^{\mathsf T}x +$ 

!!! For f convex: $\partial f(x_0) := \{d \ : \forall x \quad f(x) \geq f(x_0) + d^{\mathsf T}(x - x_0)\}$

$\Rightarrow$ $\partial Q(\cdot,\xi)(x_0) = -T^{\mathsf T}\ \{\pi : \pi \text{ opt. sol. of dual of } Q(x_0,\xi)\}$

$\Rightarrow$ $\partial\left[\frac{1}{N}\sum_{j=1}^{N} Q(\cdot,\xi^j)\right](x_0) = -\frac{1}{N}\sum_{j=1}^{N} {T^j}^{\mathsf T}\{\pi :$
. $\pi \text{ opt. sol. of dual of } Q(x_0,\xi^j)\}$

Basic Ingredients

$$\underbrace{\min_{\substack{Ax=b\\x\geq 0}} c^{\top}x + \frac{1}{N}\sum_{j=1}^{N} Q(x,\xi^j)}_{\widetilde{\mathcal{Q}}(x):=} \qquad \text{where} \qquad Q(x,\xi) := \max_{\substack{Tx+Wy=h\\y\geq 0}} q^{\top}y$$

Assume: relatively complete recourse, i.e. $\forall$ feasible x, $Q(x,\xi) < \infty$ a.s.

Conclusion:

$$c^{\top}x + \frac{1}{N}\sum_{j=1}^{N} Q(x,\xi^j) \qquad \text{is:}$$

- easy to compute on given x
- difficult to optimize
- easy to compute subgradient on given x

Cutting Plane Algorithm

Given sample $\left\{\xi^j = (q^j, T^j, W^j, h^j)\right\}_{j=1,\dots,N}$,

0. $k \leftarrow 1$; $LB^1 \leftarrow -\infty$; $UB^1 \leftarrow \infty$; $\mathfrak{Q}_1(x) \leftarrow LB^1 \ \forall x$
1. (**"Forward Step"**) Let x^k be the solution of:

$$LB^k \leftarrow \min_{\substack{Ax=b \\ x \geq 0}} c^\mathsf{T} x + \mathfrak{Q}_k(x)$$

2. (**"Backward Step"**) Compute:

$$\begin{aligned} \widetilde{\mathcal{Q}}(x^k) &\leftarrow \tfrac{1}{N} \textstyle\sum_{j=1}^N Q(x^k, \xi^j) \\ g^k &\leftarrow -\tfrac{1}{N} \textstyle\sum_{j=1}^N T^{j\mathsf{T}} \pi^{j,k} \qquad \text{(subgradient)} \end{aligned}$$

Let $UB^k \leftarrow c^\mathsf{T} x^k + \widetilde{\mathcal{Q}}(x^k)$.

- If $UB^k - LB^k \leq \epsilon$, **END**.
- Else ($LB^k < UB^k$), add plane $\widetilde{\mathcal{Q}}(x^k) + g^{k\mathsf{T}}(x - x^k)$ to $\mathfrak{Q}_k$:

$$\mathfrak{Q}_{k+1}(x) \leftarrow \max\{\mathfrak{Q}_k(x),\ \widetilde{\mathcal{Q}}(x^k) + g^{k\mathsf{T}}(x - x^k)\}$$

3. $k \leftarrow k + 1$, iterate from 1.

SDDP algorithm I

Given sample $\left\{\xi^j = (q^j, T^j, W^j, h^j)\right\}_{j=1,\dots,N}$,

0. $k \leftarrow 1$; $\quad UB^1 \leftarrow \infty$; $\quad LB^1 \leftarrow -\infty$; $\quad \mathfrak{Q}_1(x) \leftarrow LB^1 \ \forall x$
1. **Forward Step**
   1.1 Let x^k be the solution of:

$$LB^k \leftarrow \min_{\substack{Ax=b \\ x\geq 0}} c^\top x + \mathfrak{Q}_k(x)$$

   1.2 Take subsample $\{\xi^{(j)}\}_{j=1}^M$ of $\{\xi^j\}_{j=1}^N$ ($N >> M$), and with values $\{\vartheta_j := c^\top x_k + Q(x_k, \xi^{(i)})\}_{j=1}^M$ compute $(1-\alpha)$ confidence upper bound of "true" problem opt. value ϑ^*:

$$UB^k \leftarrow \overline{\vartheta} + z_{\alpha/2}\widehat{\sigma}_\vartheta/\sqrt{M}$$

   where $\overline{\vartheta} := \frac{1}{M}\sum_{j=1}^M \vartheta_j$ and $\widehat{\sigma}_\vartheta^2 := \frac{1}{M-1}\sum_{j=1}^M (\vartheta_j - \overline{\vartheta})^2$.

   1.3 If $UB^k - LB^k \leq \epsilon$, **END**.

SDDP algorithm II

2. **Backward Step**

   2.1 Compute:

$$
\begin{aligned}
\widetilde{\mathcal{Q}}(x^k) \quad &\leftarrow \quad \tfrac{1}{N}\textstyle\sum_{j=1}^{N} Q(x^k, \xi^j) = \tfrac{1}{N}\sum_{j=1}^{N} \left[\max_{\substack{T^j x^k + W^j y = h^j \\ y \geq 0}} q^{j\mathsf{T}} y \right] \\
g^k \quad &\leftarrow \quad -\tfrac{1}{N}\textstyle\sum_{j=1}^{N} T^{j\mathsf{T}} \pi^{j,k} \qquad \text{(subgradient)}
\end{aligned}
$$

   2.2 Add plane $\widetilde{\mathcal{Q}}(x^k) + g^{k\mathsf{T}}(\cdot - x^k)$ to $\mathfrak{Q}_k(\cdot)$:

$$
\mathfrak{Q}_{k+1}(x) \leftarrow \max\{\mathfrak{Q}_k(x),\ \widetilde{\mathcal{Q}}(x^k) + g^{k\mathsf{T}}(x - x^k)\}
$$

3. $k \leftarrow k + 1$, iterate from 1.

1. Two-stage stochastic programming
   - Cutting-plane method
   - SDDP algorithm for two-stage SP

2. Multistage stochastic programming
   - SDDP algorithm for multistage SP
   - Convergence and main contributions

"True" problem

$$\min_{\substack{A_1 x_1 = b_1 \\ x_1 \geq 0}} c_1^\top x_1 + \mathbb{E}\left[\min_{\substack{B_2 x_1 + A_2 x_2 = b_2 \\ x_2 \geq 0}} c_2^\top x_2 + \mathbb{E}\left[\quad \dots \quad + \mathbb{E}\left[\min_{\substack{B_T x_{T-1} + A_T x_T = b_T \\ x_T \geq 0}} c_T^\top x_T \right]\right]\right]$$

Equivalently (**Dynamic Programming equations**):

$$\min_{\substack{A_1 x_1 = b_1 \\ x_1 \geq 0}} c_1^\top x_1 + \underbrace{\mathbb{E}\left[Q_2(x_1, \xi_2) \right]}_{\mathcal{Q}_2(x_1):=}$$

where

$$\begin{aligned} Q_t(x_{t-1}, \xi_t) &:= \inf_{\substack{B_t x_{t-1} + A_t x_t = b_t \\ x_t \geq 0}} c_t^\top x_t + \underbrace{\mathbb{E}\left[Q_{t+1}(x_t, \xi_{t+1}) \right]}_{\mathcal{Q}_{t+1}(x_t)} \quad t = 2, \dots, T-1 \\ Q_T(x_{T-1}, \xi_T) &:= \inf_{\substack{B_T x_{T-1} + A_T x_T = b_T \\ x_T \geq 0}} c_T^\top x_T \end{aligned}$$

Assumptions

1. Process is stagewise independent, i.e. ξ_{t+1} indep. of $\xi_1, \dots, \xi_t$.
2. Problem has *relatively complete recourse*

SAA problem

Take random sample $\left\{\tilde{\xi}_t^j = (\tilde{c}_{tj}, \tilde{A}_{tj}, \tilde{B}_{tj}, \tilde{b}_{tj})\right\}_{j=1,\dots,N_t}$ for each stage $t = 2, \dots, T$. SAA problem is:

$$\min_{\substack{A_1 \mathbf{x_1} = b_1 \\ \mathbf{x_1} \geq 0}} c_1^{\mathsf{T}} \mathbf{x_1} + \frac{1}{N_2} \sum_{j=1}^{N_2} \left[\min_{\substack{\tilde{B}_{j2} \mathbf{x_1} + \tilde{A}_{j2} \mathbf{x_2} = \tilde{b}_{j2} \\ \mathbf{x_2} \geq 0}} \tilde{c}_{j2}^{\mathsf{T}} \mathbf{x_2} + \frac{1}{N_3} \sum_{j=1}^{N_3} \left[\quad \cdots \quad + \frac{1}{N_T} \sum_{j=1}^{N_T} \left[\min_{\substack{\tilde{B}_{jT} \mathbf{x_{T-1}} + \tilde{A}_{jT} \mathbf{x_T} = \tilde{b}_{jT} \\ \mathbf{x_T} \geq 0}} \tilde{c}_{jT}^{\mathsf{T}} \mathbf{x_T} \right] \right] \right]$$

Equivalently (**Dynamic Programming equations**):

$$\min_{\substack{A_1 \mathbf{x_1} = b_1 \\ \mathbf{x_1} \geq 0}} c_1^{\mathsf{T}} \mathbf{x_1} + \underbrace{\frac{1}{N_2} \sum_{j=1}^{N_2} \tilde{Q}_{2,j}(\mathbf{x_1})}_{\tilde{\mathcal{Q}}_2(\mathbf{x_1})}$$

where

$$\tilde{Q}_{t,j}(\mathbf{x_{t-1}}) := \min_{\substack{\tilde{B}_{tj} \mathbf{x_{t-1}} + \tilde{A}_{tj} \mathbf{x_t} = \tilde{b}_{tj} \\ \mathbf{x_t} \geq 0}} \tilde{c}_{tj}^{\mathsf{T}} \mathbf{x_t} + \underbrace{\frac{1}{N_{t+1}} \sum_{j=1}^{N_{t+1}} \tilde{Q}_{t+1,j}(\mathbf{x_t})}_{\tilde{\mathcal{Q}}_{t+1}(\mathbf{x_t})} \quad t = 2, \dots, T-1$$

$$\tilde{Q}_{T,j}(\mathbf{x_{T-1}}) := \min_{\substack{\tilde{B}_{Tj} \mathbf{x_{T-1}} + \tilde{A}_{Tj} \mathbf{x_T} = \tilde{b}_{Tj} \\ \mathbf{x_T} \geq 0}} \tilde{c}_{Tj}^{\mathsf{T}} \mathbf{x_T}$$

SDDP method: **the idea**

!!! Cost-to-go functions

$$\widetilde{\mathcal{Q}}_t(x_{t-1}) = \frac{1}{N_t}\sum_{j=1}^{N_t} \widetilde{Q}_{t,j}(x_{t-1}) \qquad t = 2, \ldots, T$$

are convex piecewise-linear

- Approximate

$$\min_{\substack{A_1 \mathbf{x_1} = b_1 \\ \mathbf{x_1} \geq 0}} \quad c_1^{\mathsf{T}} \mathbf{x_1} + \widetilde{\mathcal{Q}}_2(\mathbf{x_1}) \quad \text{by} \quad \min_{\substack{A_1 \mathbf{x_1} = b_1 \\ \mathbf{x_1} \geq 0}} \quad c_1^{\mathsf{T}} \mathbf{x_1} + \mathfrak{Q}_2(\mathbf{x_1})$$

$$\min_{\substack{\widetilde{B}_{tj}\mathbf{x_{t-1}} + \widetilde{A}_{tj}\mathbf{x_t} = \widetilde{b}_{tj} \\ \mathbf{x_t} \geq 0}} \quad \widetilde{c}_{tj}^{\mathsf{T}} \mathbf{x_t} + \widetilde{\mathcal{Q}}_{t+1}(\mathbf{x_t}) \quad \text{by} \quad \min_{\substack{\widetilde{B}_{tj}\mathbf{x_{t-1}} + \widetilde{A}_{tj}\mathbf{x_t} = \widetilde{b}_{tj} \\ \mathbf{x_t} \geq 0}} \quad \widetilde{c}_{tj}^{\mathsf{T}} \mathbf{x_t} + \mathfrak{Q}_{t+1}(\mathbf{x_t})$$

where $\mathfrak{Q}_2(\cdot), \ldots, \mathfrak{Q}_T(\cdot)$ are convex, piecewise-linear and

$$\mathfrak{Q}_t(\cdot) \leq \widetilde{\mathcal{Q}}_t(\cdot) \qquad t = 2, \ldots, T$$

- In successive iterations, refine lower approximations $\mathfrak{Q}_t(\cdot)$ using subgradient of $\widetilde{\mathcal{Q}}_t(\cdot)$:

$$\mathfrak{Q}_t^k(\cdot) \leq \mathfrak{Q}_t^{k+1}(\cdot) \leq \mathfrak{Q}_t^{k+2}(\cdot) \leq \ldots \leq \widetilde{\mathcal{Q}}_t(\cdot)$$

SDDP method: Forward step

At iteration $k \geq 1$, we have lower approximations $\mathfrak{Q}_2, \ldots, \mathfrak{Q}_T$

- Take subsample $\{(\tilde{\xi}_2^{(j)}, \ldots, \tilde{\xi}_T^{(j)})\}_{j=1}^M$ of original sample
- For $j = 1, \ldots, M$, take sampled process $(\tilde{\xi}_2^{(j)}, \ldots, \tilde{\xi}_T^{(j)})$ and solve

$$\min_{\substack{A_1 x_1 = b_1 \\ x_1 \geq 0}} \; c_1^{\mathsf{T}} x_1 + \mathfrak{Q}_2(x_1) \quad \Rightarrow x_{1j}$$

$$\tilde{\xi}_t^{(j)},\; x_{t-1,j} \Rightarrow \min_{\substack{\tilde{B}_{t(j)} x_{t-1,j} + \tilde{A}_{t(j)} x_t = \tilde{b}_{t(j)} \\ x_t \geq 0}} \; \tilde{c}_{t(j)}^{\mathsf{T}} x_t + \mathfrak{Q}_{t+1}(x_t) \quad \Rightarrow x_{tj}(\tilde{\xi}_{[t]}^{(j)})$$

$$t = 2, \ldots, T-1$$

$$\tilde{\xi}_T^{(j)},\; x_{T-1,j} \Rightarrow \min_{\substack{\tilde{B}_{T(j)} x_{T-1,j} + \tilde{A}_{T(j)} x_T = \tilde{b}_{T(j)} \\ x_T \geq 0}} \; \tilde{c}_{T(j)}^{\mathsf{T}} x_T \quad \Rightarrow x_{Tj}(\tilde{\xi}_{[T]}^{(j)})$$

obtaining candidate policy values $x_{1j}, x_{2j}, \ldots, x_{Tj}$ with cost

$$\vartheta_j \leftarrow \sum_{t=1}^{T} c_{t(j)}^{\mathsf{T}} x_{tj}$$

- It's a $(1-\alpha)$ confidence upper bound of "true" opt. value ϑ^*:

$$UB^k \leftarrow \overline{\vartheta} + z_\alpha \hat{\sigma}_\vartheta / \sqrt{M}$$

where $\overline{\vartheta} := \frac{1}{M} \sum_{j=1}^M \vartheta_j$ and $\hat{\sigma}_\vartheta^2 := \frac{1}{M-1} \sum_{j=1}^M (\vartheta_j - \overline{\vartheta})^2$.

SDDP method: Backward step

We have candidate values $\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_T$.

- At stage $t = T$:
  - For $x_{T-1} = \overline{x}_{T-1}$ and for $j = 1, \ldots, N_T$, solve:

$$\widetilde{Q}_{T,j}(x_{T-1}) := \min_{\substack{\widetilde{B}_{T,j}\mathbf{x_{T-1}} + \widetilde{A}_{T,j}\mathbf{x_T} = \widetilde{b}_{T,j} \\ \mathbf{x_T} \geq 0}} \widetilde{c}_{T,j}^{\mathrm{T}} \mathbf{x_T}$$

    and let $\widetilde{\pi}_{T,j}$ be opt. dual solution.
  - Let

$$\begin{aligned} \widetilde{\mathcal{Q}}_T(\overline{x}_{T-1}) &:= \frac{1}{N_T} \sum_{j=1}^{N_T} \widetilde{Q}_{T,j}(x_{T-1}) \\ \widetilde{g}_T &:= -\frac{1}{N_T} \sum_{j=1}^{N_T} \widetilde{B}_{T,j}^{\mathrm{T}} \widetilde{\pi}_{T,j} \ . \end{aligned}$$

  - Add cut

$$L_T(x_{T-1}) := \widetilde{\mathcal{Q}}_T(\overline{x}_{T-1}) + \widetilde{g}_T^{\mathrm{T}}(x_{T-1} - \overline{x}_{T-1})$$

    to lower approx. $\mathfrak{Q}_{\mathbf{T}}$ used in stage $t = \mathbf{T-1}$:

$$\mathfrak{Q}_{\mathbf{T}}(\cdot) := \max\{\mathfrak{Q}_T(\cdot), L_T(\cdot)\}.$$

SDDP method: Backward step

- At stage $t = T - 1$:
  - For $x_{T-2} = \bar{x}_{T-2}$ and for $j = 1, \ldots, N_{T-1}$, solve:

$$\widetilde{Q}_{T-1,j}(x_{T-2}) := \min_{\substack{\widetilde{B}_{T-1,j}\mathbf{x_{T-2}} + \widetilde{A}_{T-1,j}\mathbf{x_{T-1}} = \widetilde{b}_{T-1,j} \\ \mathbf{x_{T-1}} \geq 0}} \widetilde{c}_{T-1,j}^{\mathsf{T}} \mathbf{x_{T-1}} + \mathfrak{Q}_T(\mathbf{x_{T-1}})$$

    and let $\widetilde{\pi}_{T-1,j}$ be opt. dual solution.
  - Let

$$\begin{array}{lcl} \widetilde{\mathcal{Q}}_{T-1}(\bar{x}_{T-2}) & := & \frac{1}{N_{T-1}} \sum_{j=1}^{N_{T-1}} \widetilde{Q}_{T-1,j}(x_{T-2}) \\ \widetilde{g}_{T-1} & := & -\frac{1}{N_{T-1}} \sum_{j=1}^{N_{T-1}} \widetilde{B}_{T-1,j}^{\mathsf{T}} \widetilde{\pi}_{T-1,j} \; . \end{array}$$

  - Add cut

$$L_{T-1}(x_{T-2}) := \widetilde{\mathcal{Q}}_{T-1}(\bar{x}_{T-2}) + \widetilde{g}_{T-1}^{\mathsf{T}}(x_{T-2} - \bar{x}_{T-2})$$

    to lower approx. $\mathfrak{Q}_{\mathbf{T-1}}$ used in stage $t = \mathbf{T} - \mathbf{2}$:

$$\mathfrak{Q}_{\mathbf{T-1}}(\cdot) := \max\{\mathfrak{Q}_{T-1}(\cdot), L_{T-1}(\cdot)\}.$$

SDDP method: Backward step

- At stage $t = T - 2$:

  ...

  ...

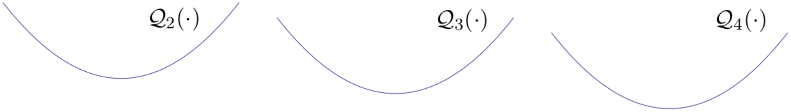
  ...

- At stage $t = 1$: solve

$$LB^k \leftarrow \min_{\substack{A_1 x_1 = b_1 \\ x_1 \geq 0}} \; c_1^\mathsf{T} x_1 + \mathfrak{Q}_2(x_1)$$

LB^k is, on average, lower bound to ϑ^* "true" optimal value

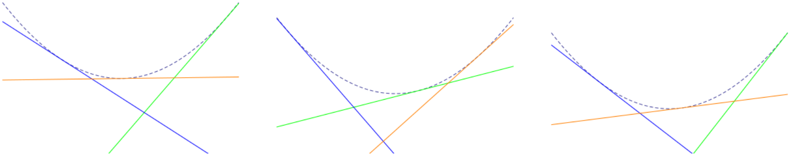
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: "True" problem

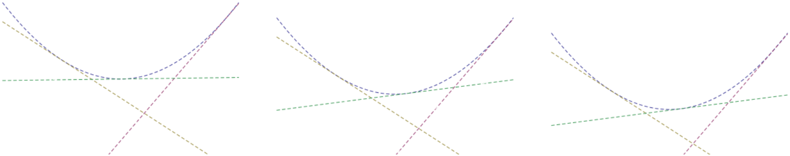
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SAA problem

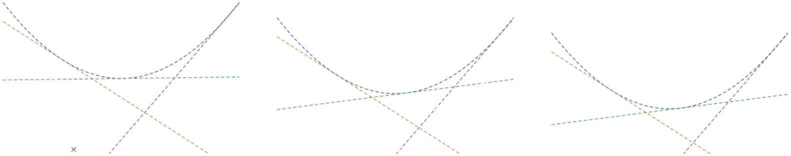
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 1: Forward step

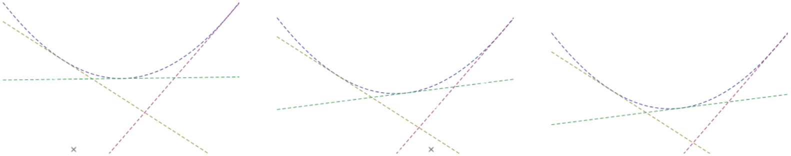
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 1: Forward step

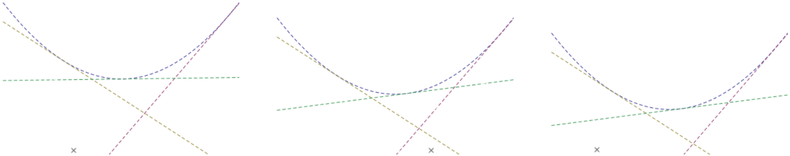
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 1: Forward step

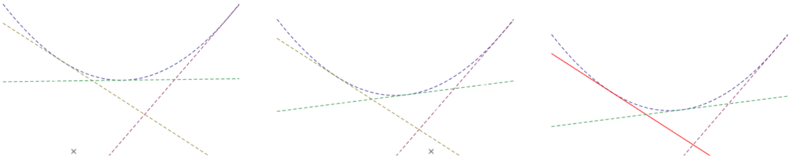
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 1: Forward step

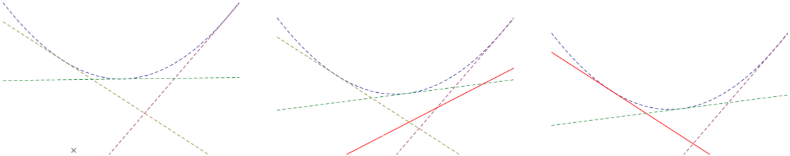
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 1: Backward step

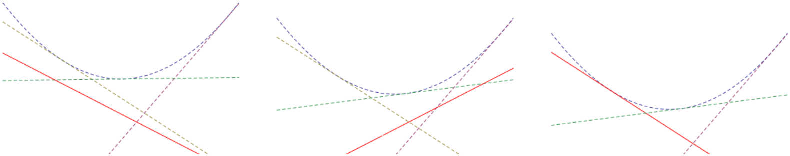
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 1: Backward step

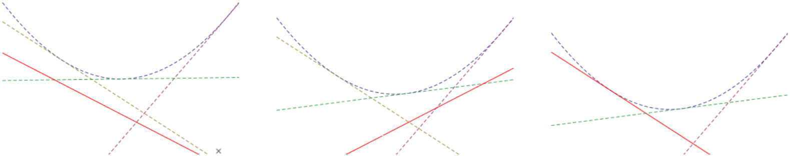
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 1: Backward step

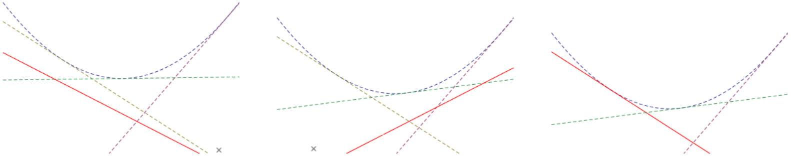
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 2: Forward step

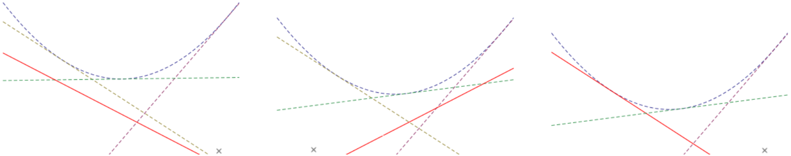
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 2: Forward step

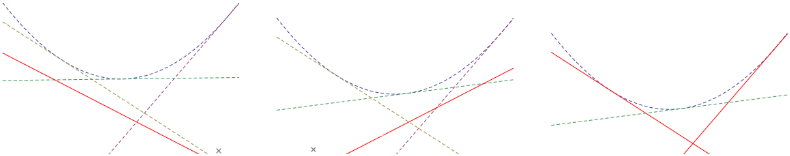
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 2: Forward step

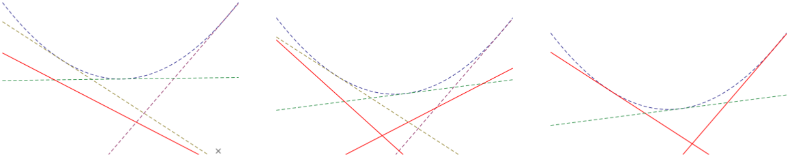
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 2: Backward step

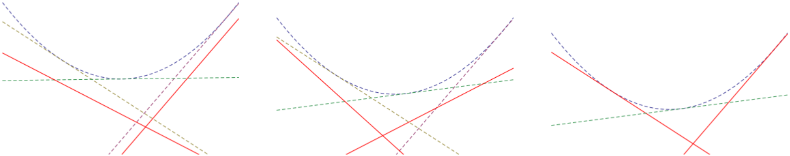
SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 2: Backward step

SDDP method: Illustration



(a) Stage 1 (b) Stage 2 (c) Stage 3

Figure: SDDP iteration 2: Backward step...

Summary: **KEY IDEAS** for SDDP algorithm

- **Forward step**:
  - sample process $\xi_1, \ldots, \xi_T \;\Rightarrow\;$ implementable policy $x_1, \ldots, x_T$
  - Repetitions $\Rightarrow$ Upper bound on true optimal value
- **Backward step**:
  - Cost-to-go functions

$$\widetilde{\mathcal{Q}}_t(x_{t-1}) = \frac{1}{N_t} \sum_{j=1}^{N_t} \left[\max_{\substack{\widetilde{B}_{t,j} x_{t-1} + \widetilde{A}_{t,j} x_t = \widetilde{b}_{t,j} \\ x_t \geq 0}} \widetilde{c}_{t,j}^{\mathsf{T}} x_t + \widetilde{\mathcal{Q}}_{t+1}(x_t) \right]$$

are convex piecewise-linear functions of x_{t-1}

  - Refine lower approximations $\mathfrak{Q}_t(\cdot)$ using subgradient

$$\partial \left[\widetilde{\mathcal{Q}}_t(\cdot) \right] (\overline{x}_{t-1}) = -\frac{1}{N_t} \sum_{j=1}^{N_t} \widetilde{B}_{t,j}^{\mathsf{T}} \left\{ \pi_{t,j} \;:\; \text{opt. sol. of dual} \ldots \right\}$$

  - Lower bound on true optimal value

Proposition (Convergence)

Assume

i. *At the forward step, process subsamples are taken independently of each other*

ii. *At all iterations, approximated problems*

$$\min_{\substack{A_1 x_1 = b_1 \\ x_1 \geq 0}} \; c_1^{\mathsf{T}} x_1 + \mathfrak{Q}_2(x_1) \quad \textit{and} \quad \min_{\substack{\widetilde{B}_{t,j} x_{t-1} + \widetilde{A}_{t,j} x_t = \widetilde{b}_{t,j} \\ x_t \geq 0}} \; \widetilde{c}_{t,j}^{\mathsf{T}} x_t + \mathfrak{Q}_{t+1}(x_t)$$

have finite optimal value

iii. *In the backward step, basical solutions are used*

Then, after a sufficiently but finite large amount of iterations of the SDDP algorithm, the forward procedure defines an optimal policy for the SAA problem.

Tractability?

- **Issue**: total number of scenarios $\Pi_{t=2}^{T} N_t$
- Cutting plane for two-stage: bad
- SDDP algorithm: generalization of cutting plane method $\Rightarrow$ even worse?
  - One run of Backward step: solve $1 + N_2 + \ldots + N_T$ LP's
  - One run of Forward step: solve $1 + M(T-1)$ LP's
- """"Tractability"""":

SDDP method $\Rightarrow$ construct **feasible** policy

THE END