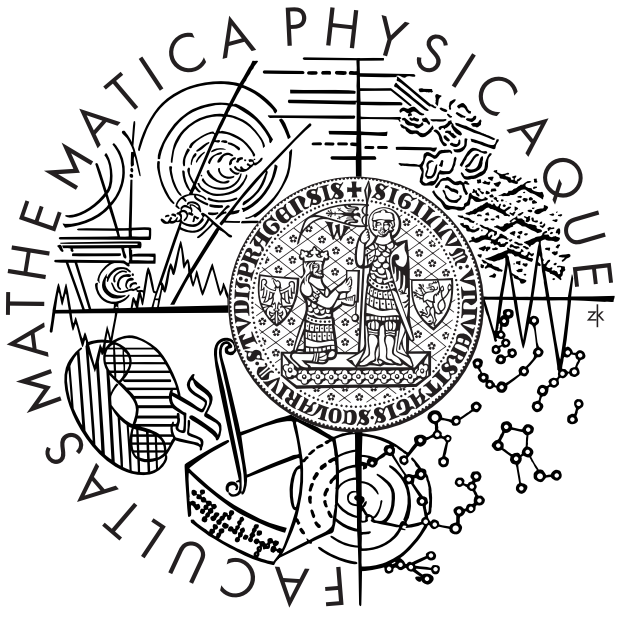
# Benders' decomposition in Stochastic Programming



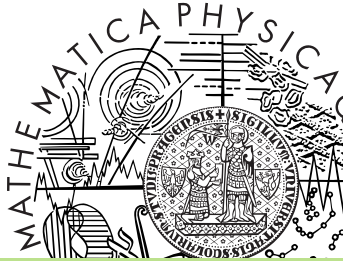

Václav Kozmík

Faculty of Mathematics and Physics
Charles University in Prague

April 28, 2014

# Outline

- Two-stage models
  - Theoretical results
  - Decomposition methods
  - Risk aversion
- Multistage stochastic models
  - Much harder than the two-stage models
  - Complexity explosion when increasing the number of stages
  - Interstage independence can help to reduce complexity
  - Risk measures bring new pitfalls
- Basic results and notation from Shapiro, Dentcheva and Ruszczynski (2009)



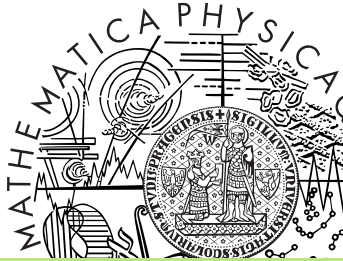
# Two-stage stochastic model

- Data vector $\boldsymbol{\xi} = (\mathbf{q}, \mathbf{h}, \mathbf{T}, \mathbf{W})$
- Decision vectors $\mathbf{x}$, $\mathbf{y}$
- Probability distribution of $\boldsymbol{\xi}$: $\mathbb{P}$
- We consider following stochastic program (SP):

$$\begin{aligned} \min_{\mathbf{x}} \ & \mathbf{c}^\top \mathbf{x} + \mathbb{E}\left[Q(\mathbf{x}, \boldsymbol{\xi})\right] \\ \text{s.t.} \ & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0, \end{aligned}$$

where $Q(\mathbf{x}, \boldsymbol{\xi})$ is given by (REC):

$$\begin{aligned} \min_{\mathbf{y}} \ & \mathbf{q}^\top \mathbf{y} \\ \text{s.t.} \ & \mathbf{T}\mathbf{x} + \mathbf{W}\mathbf{y} = \mathbf{h} \\ & \mathbf{y} \geq 0. \end{aligned}$$



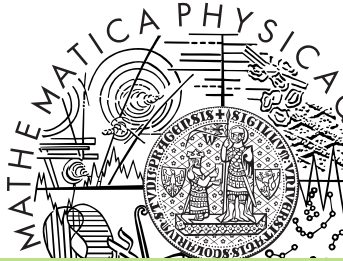
# Two-stage stochastic model

- The program giving a value of $Q(\mathbf{x}, \boldsymbol{\xi})$ is called second-stage problem
- It's dual version (DUAL):

$$\max_{\mathbf{y}} \boldsymbol{\pi}^{\top} (\mathbf{h} - \mathbf{T}\mathbf{x})$$
$$\text{s.t. } \mathbf{W}\boldsymbol{\pi} \leq \mathbf{q}$$

## Proposition

*For any given $\boldsymbol{\xi}$ the function $Q(\cdot, \boldsymbol{\xi})$ is convex. Moreover, if the set of dual feasible solutions is nonempty and problem (REC) is feasible for at least one $\mathbf{x}$, then the function $Q(\cdot, \boldsymbol{\xi})$ is polyhedral.*



# Two-stage stochastic model

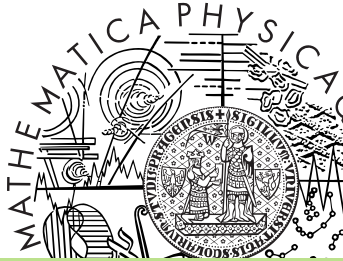
## Proposition

*Suppose that for given* $\mathbf{x} = \mathbf{x}_0$ *and* $\boldsymbol{\xi}$ *the value* $Q(\mathbf{x}_0, \boldsymbol{\xi})$ *is finite. Then* $Q(\cdot, \boldsymbol{\xi})$ *is subdifferentiable at* $\mathbf{x}_0$ *and*

$$\partial Q(\mathbf{x}_0, \boldsymbol{\xi}) = -\mathbf{T}^\top \mathcal{D}(\mathbf{x}_0, \boldsymbol{\xi})$$

*where* $\mathcal{D}(\mathbf{x}_0, \boldsymbol{\xi})$ *is the set of optimal solutions of the dual problem (DUAL).*

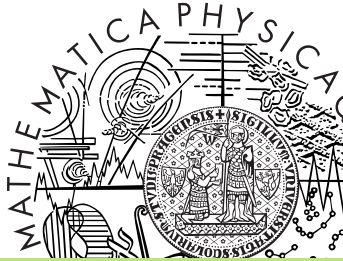
- We have a polyhedral function
- We know how to calculate a subgradient
- This provides a basic block for further development



# Two-stage stochastic model

- Suppose that the distribution $\mathbb{P}$ of $\boldsymbol{\xi}$ is discrete
  - Scenarios $\boldsymbol{\xi}^k = \left(\mathbf{q}^k, \mathbf{h}^k, \mathbf{T}^k, \mathbf{W}^k\right)$ with probabilities $p^k$, $k = 1, \ldots, K$
- If the distribution $\mathbb{P}$ is not discrete, we can obtain it's discrete version $\hat{\mathbb{P}}$ by Monte Carlo sampling
  - But we are not solving the original problem!
  - The quality of solutions has to be controlled, the sampling process repeated
  - See Bayraksan and Morton (2009) for further details
- The expectation is replaced by sum:

$$\mathbb{E}\left[Q(\mathbf{x}, \boldsymbol{\xi})\right] = \sum_{k=1}^{K} p^k Q(\mathbf{x}, \boldsymbol{\xi}^k)$$

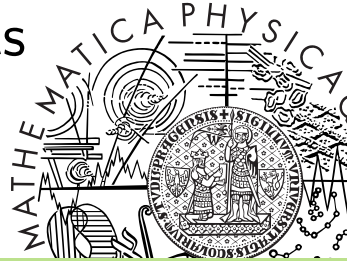


# Two-stage stochastic model

- The whole program can be combined into one large-scale linear programming problem:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{y}^1,\ldots,\mathbf{y}^K} \quad & \mathbf{c}^\top \mathbf{x} + \sum_{k=1}^{K} p^k \mathbf{q}^{k\top} \mathbf{y}^k \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{T}^k \mathbf{x} + \mathbf{W}^k \mathbf{y}^k = \mathbf{h}^k, \;\; k = 1, \ldots, K \\ & \mathbf{x} \geq 0, \mathbf{y}^k \geq 0, \;\; k = 1, \ldots, K. \end{aligned}$$

- This program can be solved by a standard solver like CPLEX
- However, the computation times and memory requirements are extensive for a large number of scenarios



# Two-stage stochastic model

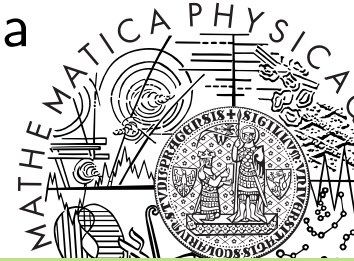
- Denote $\phi(x) = \mathbb{E}\left[Q(\mathbf{x}, \boldsymbol{\xi})\right]$

## Proposition

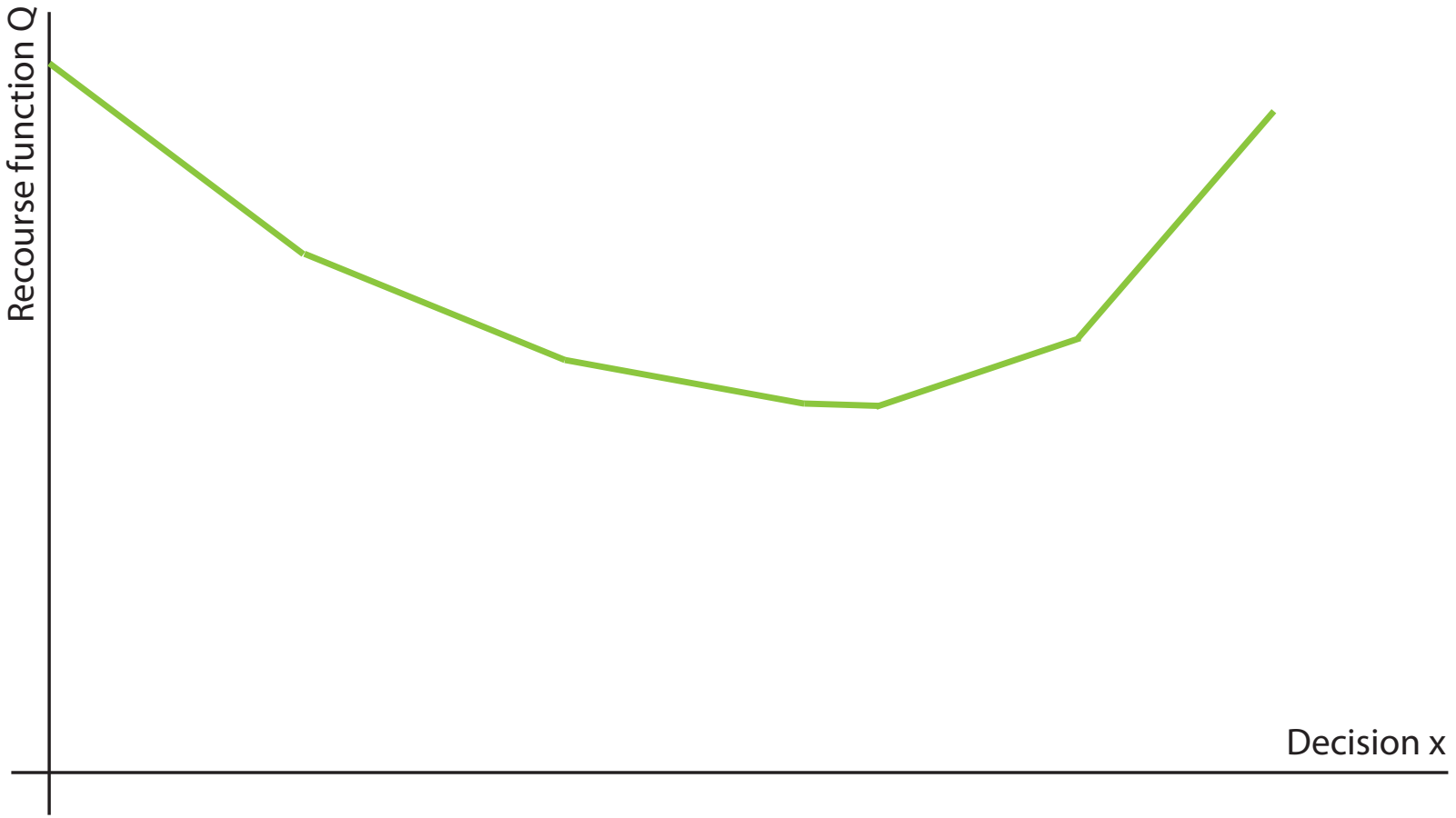
*Suppose that the probability distribution of $\boldsymbol{\xi}$ has finite support with scenarios $\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^K$. Suppose that the expected recourse cost $\phi(\cdot)$ has a finite value in at least one point $\bar{\mathbf{x}} \in \mathbb{R}^n$. Then the function $\phi(\cdot)$ is polyhedral and for any $\mathbf{x}_0 \in$ dom $\phi$:*

$$\partial\phi(\mathbf{x}_0) = \sum_{k=1}^{K} p^k \partial Q(\mathbf{x}_0, \boldsymbol{\xi}^k)$$

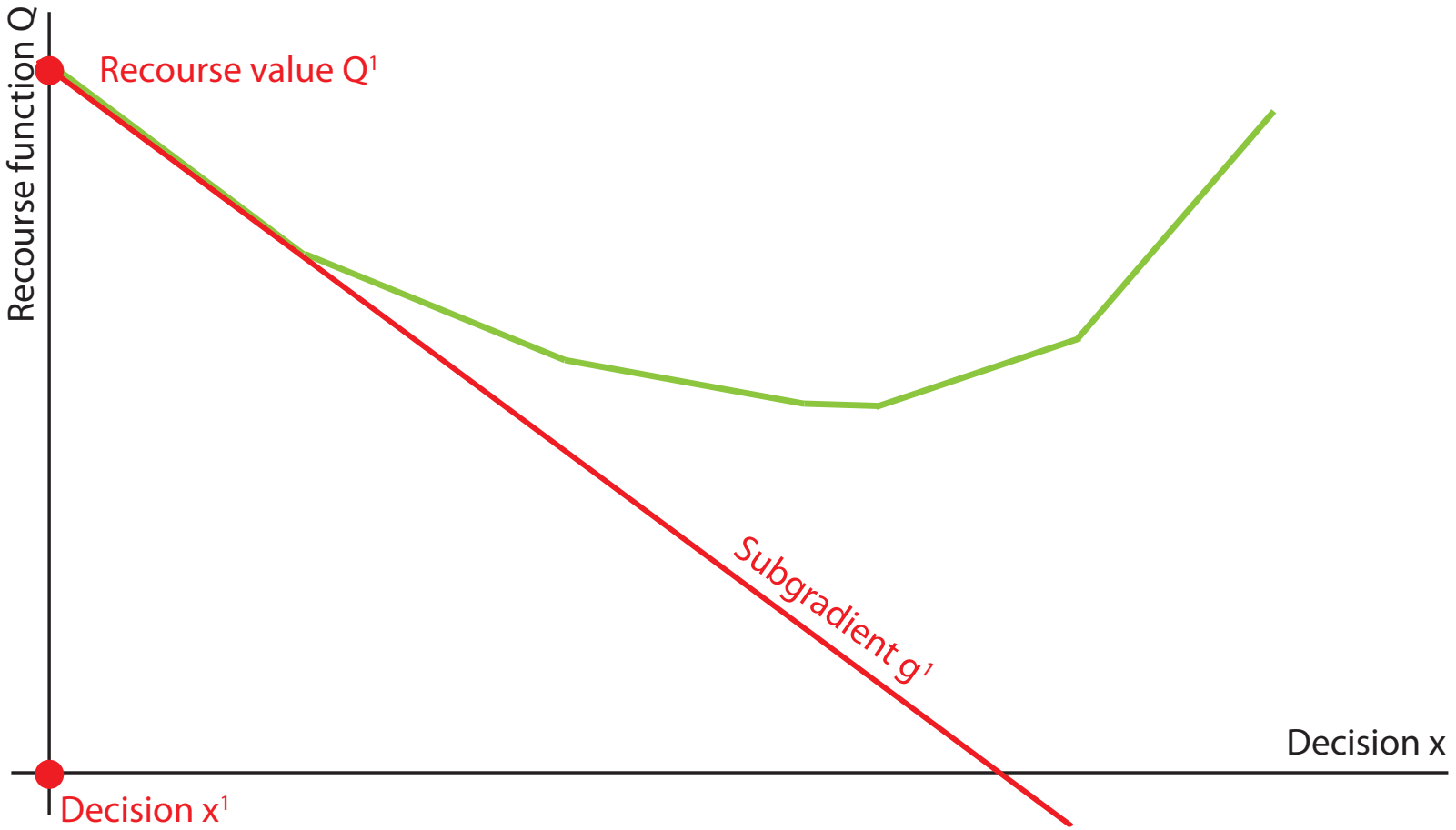
- The function is polyhedral and we know how to calculate a subgradient
- These properties are important for algorithm development



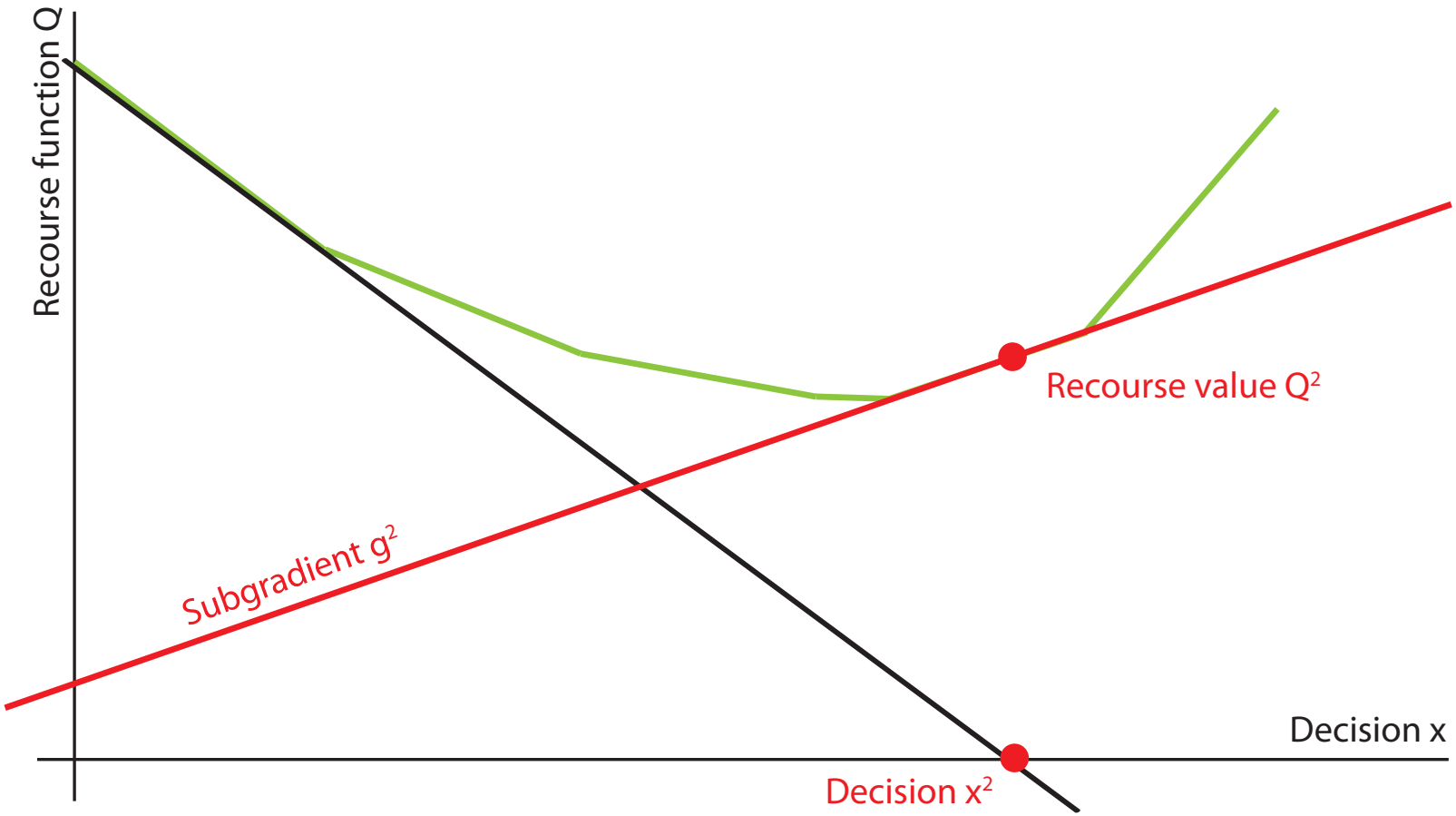
# Benders' decomposition



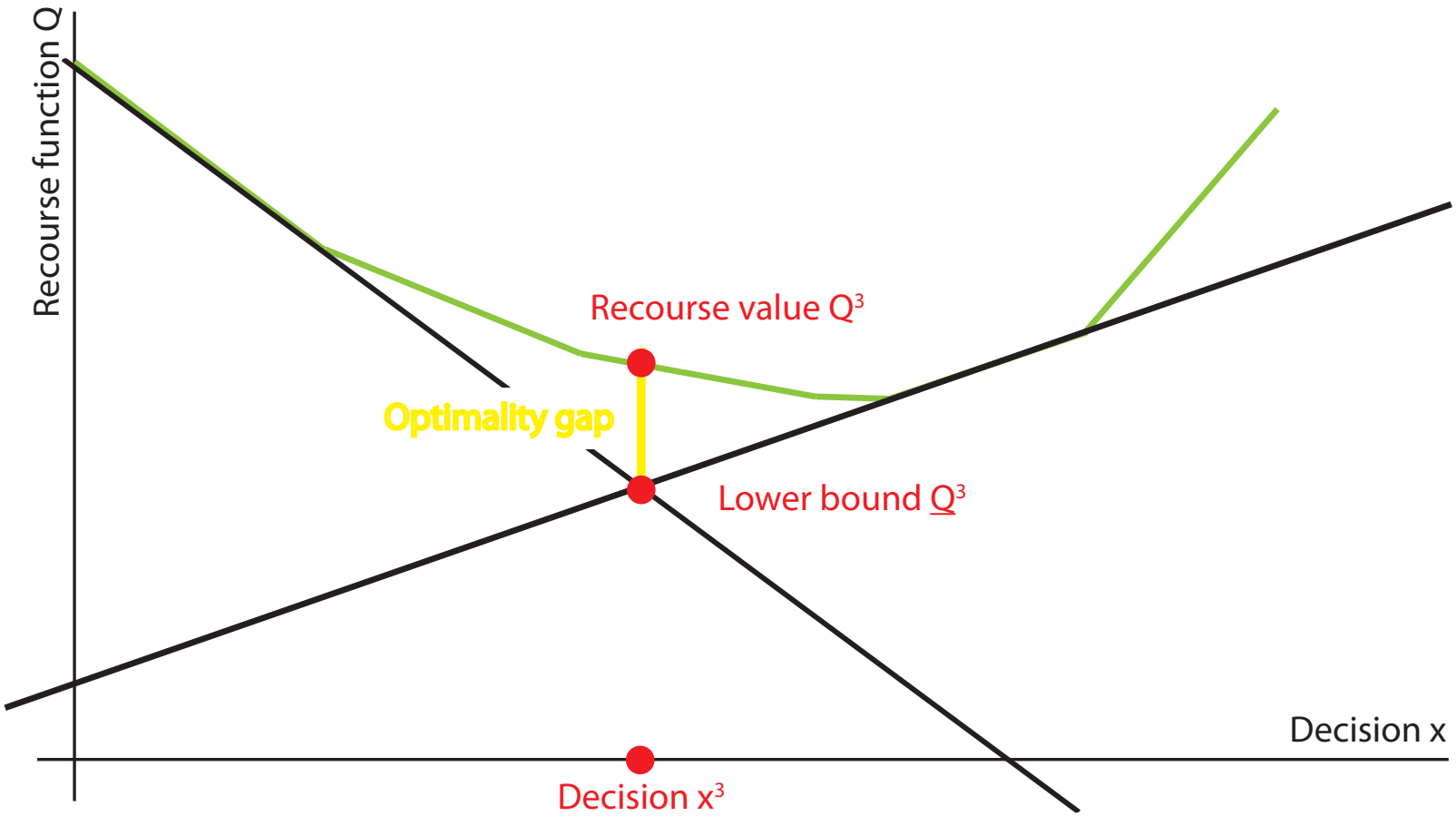
# Benders' decomposition



# Benders' decomposition



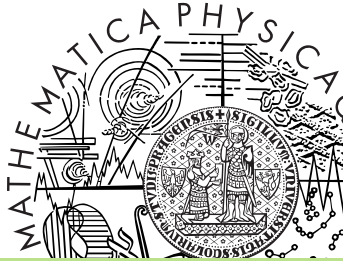
# Benders' decomposition



# Benders' decomposition

- The expectation $\mathbb{E}\left[Q(\mathbf{x}, \boldsymbol{\xi})\right]$ is hard to compute, but since we know it's properties, we replace it with a lower bounding approximation $\theta$
- Assume relatively complete recourse
- As we will solve the master problem and it's subproblems, we will be collecting cuts for the variable $\theta$
- We add some lower bound for the recourse value, e.g. $\theta \geq 0$
- Suppose we have collected $C$ cuts so far, then the master problem is given by:

$$
\begin{aligned}
\min_{\mathbf{x}} \ & \mathbf{c}^\top \mathbf{x} + \theta \\
\text{s.t. } & \mathbf{A}\mathbf{x} = \mathbf{b} \\
& \theta \geq \hat{\mathcal{Q}}_j + \left(\mathbf{g}_j\right)^\top \left(\mathbf{x} - \mathbf{x}_j\right), \quad j = 1, \ldots, C \\
& \mathbf{x} \geq 0, \theta \geq 0
\end{aligned}
$$



# Benders' decomposition

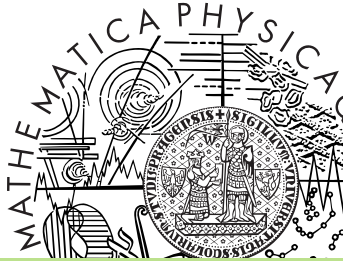
- At iteration $j$, we will have some suboptimal decision $\mathbf{x}_j$ of the master program along with it's optimal value $z_j$
- For each of the scenarios $\boldsymbol{\xi}^1, \ldots, \boldsymbol{\xi}^K$ we will solve following subproblem:

$$\begin{aligned} Q_j^k = \min_{\mathbf{y}} \ & \mathbf{q}^{k\top}\mathbf{y} \\ \text{s.t.} \ & \mathbf{T}^k\mathbf{x}_j + \mathbf{W}^k\mathbf{y} = \mathbf{h}^k \\ & \mathbf{y} \geq 0. \end{aligned}$$

- Along with the optimal value $Q_j^k$ we also calculate subgradient $\mathfrak{g}_j^k$:

$$\mathfrak{g}_j^k = -\mathbf{T}^{k\top}\mathcal{D}(\mathbf{x}_j, \boldsymbol{\xi}^k)$$

where $\mathcal{D}(\mathbf{x}_j, \boldsymbol{\xi}^k)$ is the optimal solution of the problem above

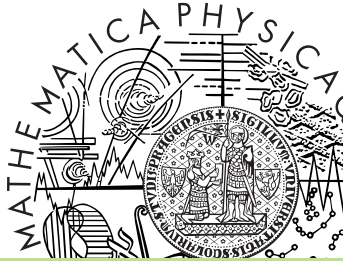


# Benders' decomposition

- We average the optimal values and subgradients:

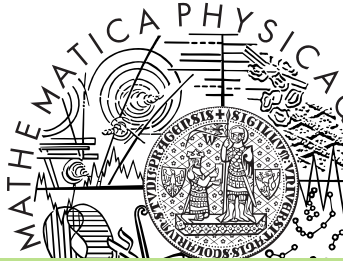
$$\hat{\mathcal{Q}}_j = \sum_{k=1}^{K} p^k Q_j^k, \quad \mathfrak{g}_j = \sum_{k=1}^{K} p^k \mathfrak{g}_j^k$$

- We append a cut defined by $\hat{\mathcal{Q}}_j$ and $\mathfrak{g}_j$ to the master program and solve it again with new solution $\mathbf{x}_{j+1}$ and optimal solution $z_{j+1}$
- $\hat{\mathcal{Q}}_j$ is the objective value of a suboptimal solution, therefore stands for an upper bound on the true solution
- $z_{j+1}$ is a solution based on lower outer approximation, therefore represents a lower bound for the true solution
- If $\hat{\mathcal{Q}}_j - z_{j+1} < \epsilon$ we stop, otherwise repeat the steps



# Benders' decomposition

- Benders (1962), Van Slyke and Wets (1969) - also called L-Shaped method
- The convergence is guaranteed in a finite number of steps
- We can also deal with the case when the assumption of relatively complete recourse is not fulfilled
  - Feasibility cuts
- Many improvements and extensions exist:
  - Dropping cuts
  - Warm start
  - Multicut method - Birge and Louveaux (1988)
  - Regularized decomposition - Ruszczynski (1986)
- Other important algorithms:
  - Stochastic decomposition - Higle and Sen (1991)
  - Progressive hedging - Rockafellar and Wets (1991)



# Risk measures

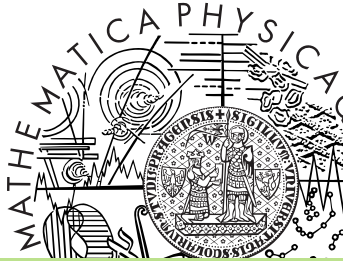
- Similar approach can be applied to more complicated functionals, for example convex risk measures
- Consider following master program:

$$\begin{aligned} \min_{\mathbf{x}} \ & \mathbf{c}^\top \mathbf{x} + (1-\lambda)\mathbb{E}\left[Q(\mathbf{x},\boldsymbol{\xi})\right] + \lambda\, \mathrm{CVaR}_\alpha\left[Q(\mathbf{x},\boldsymbol{\xi})\right] \\ \text{s.t. } & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0. \end{aligned}$$

- The subproblem remains the same
- The formula of Rockafellar and Uryasev (2002) can be used to simplify the problem:

$$\mathrm{CVaR}_\alpha\left[Z\right] = \min_u \left( u + \frac{1}{\alpha}\mathbb{E}\left[Z-u\right]_+ \right),$$

where $[\,\cdot\,]_+ \equiv \max\{\,\cdot\,, 0\}$.



# Risk measures

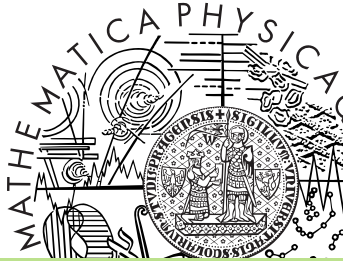
- We combine the objectives into one program:

$$\begin{aligned}\min_{\mathbf{x},u}\ & \mathbf{c}^\top \mathbf{x} + (1-\lambda)\mathbb{E}\left[Q(\mathbf{x},\boldsymbol{\xi})\right] + \lambda u + \lambda\frac{1}{\alpha}\mathbb{E}\left[Q(\mathbf{x},\boldsymbol{\xi}) - u\right]_+ \\ \text{s.t. } & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0.\end{aligned}$$

- Equivalently

$$\begin{aligned}\min_{\mathbf{x},u}\ & \mathbf{c}^\top \mathbf{x} + \lambda u + \mathcal{Q}(\mathbf{x}) \\ \text{s.t. } & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0.\end{aligned}$$

with $\mathcal{Q}(\mathbf{x}) = (1-\lambda)\mathbb{E}\left[Q(\mathbf{x},\boldsymbol{\xi})\right] + \lambda\frac{1}{\alpha}\mathbb{E}\left[Q(\mathbf{x},\boldsymbol{\xi}) - u\right]_+$



# Risk measures

- It remains to show how to calculate subgradient of this functional
- Following Shapiro (2011):

$$\mathfrak{g}_j = \left[ \sum_{k=1}^{K} p^k (1-\lambda) \mathfrak{g}_j^k + \frac{\lambda}{\alpha} \sum_{k \in K_*} p^k \mathfrak{g}_j^k \ , \ -\frac{\lambda}{\alpha} |K_*| \right],$$

where the index set

$$K_* = \left\{ k : Q_j^k > u, \ k = 1, \ldots, K) \right\}.$$

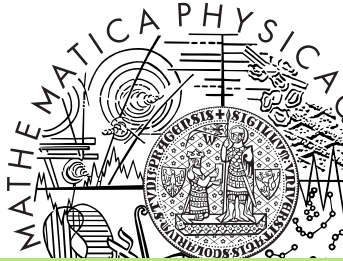
- The algorithm will require slightly more iterations due to the additional variable $u$

# Multistage stochastic optimization

- Consider $T$ stage stochastic program:
  - Data process $\boldsymbol{\xi} = (\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_T)$
  - Decision process $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$
  - Filtration $\mathcal{F}_t$ generated by the projection $\Pi_t \boldsymbol{\xi} = \boldsymbol{\xi}_{[t]} := (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_t)$
  - Sequence of decisions at stages $1, \ldots, t$: $\Pi_t \mathbf{x} = \mathbf{x}_{[t]} := (\mathbf{x}_1, \ldots, \mathbf{x}_t)$
  - Probability distribution of $\boldsymbol{\xi}$: $\mathbb{P}$
  - $\mathbb{P}_t$ denotes the marginal probability distribution of $\boldsymbol{\xi}_t$
  - $\mathbb{P}_t \left[ \cdot | \boldsymbol{\xi}_{[t-1]} \right]$ denotes the conditional probability distribution
- The decision process is *nonanticipative*:
  - Decisions taken at any stage of the process do neither depend on future *realizations* of stochastic data nor on future decisions
  - $\mathbf{x}_t$ is $\mathcal{F}_t$-measurable
  - The sequence of decisions and observations is:

$$\mathbf{x}_1, \boldsymbol{\xi}_2, \mathbf{x}_2(\mathbf{x}_1, \boldsymbol{\xi}_2), \ldots, \mathbf{x}_T(\mathbf{x}_{T-1}, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_T)$$

    - its random outcome $f(\mathbf{x}, \boldsymbol{\xi})$



# Multistage stochastic optimization

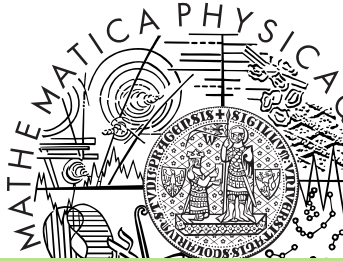
- Nested form of multistage stochastic linear program (MSLP):

$$\min_{\mathbf{x}_1 \in \mathcal{X}_1} \mathbf{c}_1^\top \mathbf{x}_1 + \mathbb{E}_{\mathbb{P}} \left[ Q_2(\mathbf{x}_1, \boldsymbol{\xi}_2) \right] \text{ with } \mathcal{X}_1 := \{ \mathbf{x}_1 | \, \mathbf{A}_1 \mathbf{x}_1 = \mathbf{b}_1, \, \mathbf{x}_1 \geq 0 \}$$

  - With $Q_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_{[t]})$, $t = 2, \ldots, T$, defined recursively as

$$Q_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_{[t]}) = \min_{\mathbf{x}_t} \mathbf{c}_t(\boldsymbol{\xi}_{[t]})^\top \mathbf{x}_t + \mathbb{E}_{\mathbb{P}_{t+1}[\cdot | \boldsymbol{\xi}_{[t]}]} \left[ Q_{t+1}(\mathbf{x}_t, \boldsymbol{\xi}_{[t+1]}) \right]$$

- In the case of *stagewise independence* the conditional distributions boil down to marginal distributions $\mathbb{P}_t$ of $\boldsymbol{\xi}_t$
- We assume:
  - Constraints involving random elements hold almost surely
  - *All infima are attained*, which is related with the *relatively complete recourse*
  - *All conditional expectations exist*



# Risk-averse multistage programs

- In the risk-neutral programs possible risks are not reflected
- Risk measure is a functional which assigns a real value to the random outcome $f(\mathbf{x}, \boldsymbol{\xi})$
- Risk measures depend on decisions and probability distribution $\mathbb{P}$.
  - They should also incorporate the filtration $\mathcal{F}_1 \subset \cdots \subset \mathcal{F}_t \cdots \subseteq \mathcal{F}$
- Risk monitoring in individual stages should be incorporated

$$\min_{\mathbf{x}_1} \mathbf{c}_1^\top \mathbf{x}_1 + \rho_2 \left( \min_{\mathbf{x}_2} \mathbf{c}_2(\boldsymbol{\xi}_{[1]})^\top \mathbf{x}_2 + \cdots + \rho_T \left( \min_{\mathbf{x}_T} \mathbf{c}_T(\boldsymbol{\xi}_{[T-1]})^\top \mathbf{x}_T \right) \right)$$

  - Different risk measures $\rho_t$ can be applied in each stage
- Coherence of $\rho$ is mostly expected [Artzner et al., 2007]

## Definition (Multiperiod risk measures, Artzner et al., 2007)

*A functional* $\rho$ *on* $\times_{t=1}^{T}\mathcal{L}_p(\Omega, \mathcal{F}_t, \mathbb{P})$ *is called a multi-period coherent risk measure if it satisfies following:*

1. *if* $Z_t \geq \tilde{Z}_t$ *a.s.,* $t = 1, \ldots, T$*, then* $\rho(Z_1, \ldots, Z_T) \geq \rho\left(\tilde{Z}_1, \ldots, \tilde{Z}_T\right)$ *(monotonicity);*
2. *for each* $r \in \mathbb{R}$ *we have* $\rho(Z_1 + r, \ldots, Z_T + r) = \rho(\mathbf{Z}) + r$ *(translation invariance);*
3. $\rho\left(\mu Z_1 + (1-\mu)\tilde{Z}_1, \ldots, \mu Z_T + (1-\mu)\tilde{Z}_T\right) \leq$ $\mu\rho(Z_1, \ldots, Z_T) + (1-\mu)\rho\left(\tilde{Z}_1, \ldots, \tilde{Z}_T\right)$ *for* $\mu \in [0, 1]$ *(convexity);*
4. *for* $\mu \geq 0$ *we have* $\rho(\mu Z_1, \ldots, \mu Z_T) = \mu\rho(Z_1, \ldots, Z_T)$ *(positive homogeneity).*

# Time consistency

- At every state of the system, optimality of our decisions should not depend on scenarios which we already know cannot happen in the future.
- Risk neutral stochastic programs are time constistent
- Risk averse stochastic programs don't have to be time consistent
- Many slightly different definitions

## Definition (Time consistency)

*If* $1 \leq t_1 < t_2 \leq T$ *and* $\mathbf{x}_t^* \left( \boldsymbol{\xi}_{[t]} \right)$, $t = t_1, \ldots, T$, *is an optimal solution for* $t = t_1$, *conditional on a realization* $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{t_1}$ *of the process, then* $\mathbf{x}_t^* \left( \boldsymbol{\xi}_{[t]} \right)$, $t = t_2, \ldots, T$, *is an optimal solution for* $t = t_2$, *conditional on a realization* $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{t_1}, \boldsymbol{\xi}_{t_1+1}, \ldots, \boldsymbol{\xi}_{t_2}$ *of the process.*

## Definition (Conditional risk mappings, Ruszcz., Shapiro, 2006)

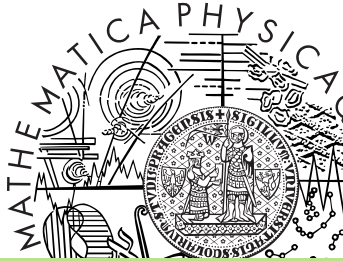
*Let $\mathcal{F} \subset \mathcal{F}'$ be $\sigma$-fields of subsets of $\Omega$ and $\mathcal{Z}$ and $\mathcal{Z}'$ be linear spaces of real-valued functions $f(\omega)$, $\omega \in \Omega$ measurable with respect to $\mathcal{F}$ and $\mathcal{F}'$, respectively. We say that mapping $\rho : \mathcal{Z}' \to \mathcal{Z}$ is a conditional risk mapping if the following properties hold:*

1. *Convexity. If $\alpha \in [0,1]$ and $X, Y \in \mathcal{Z}'$, then*

$$\alpha\rho(X) + (1-\alpha)\rho(Y) \succeq \rho(\alpha X + (1-\alpha)Y).$$

2. *Monotonicity. If $Y \succeq X$, then $\rho(Y) \succeq \rho(X)$.*
3. *Predictable Translation Equivariance. If $Y \in \mathcal{Z}$ and $X \in \mathcal{Z}'$, then*

$$\rho(X+Y) = \rho(X) + Y.$$



# Composite risk measures

Consider conditional risk mappings $\rho_2, \ldots, \rho_T$ and a risk function $\rho : \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_T \to \mathbb{R}$ given by:

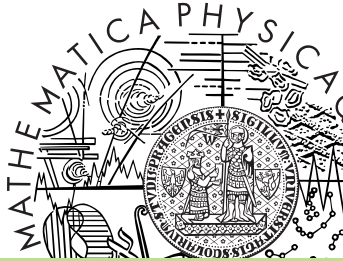
$$\rho\left(Z_1, \ldots, Z_T\right) = Z_1 + \rho_2\left(Z_2 + \cdots \rho_{T-1}\left(Z_{T-1} + \rho_T\left(Z_T\right)\right)\right).$$

Using Predictable Translation Equivariance we get

$$\rho_{T-1}\left(Z_{T-1} + \rho_T\left(Z_T\right)\right) = \rho_{T-1} \circ \rho_T\left(Z_{T-1} + Z_T\right).$$

By continuing this process we end up with a composite risk measure $\bar{\rho} := \rho_2 \circ \cdots \circ \rho_T$. It holds

$$\bar{\rho}(Z_1 + \cdots + Z_T) = \rho\left(Z_1, \ldots, Z_T\right).$$

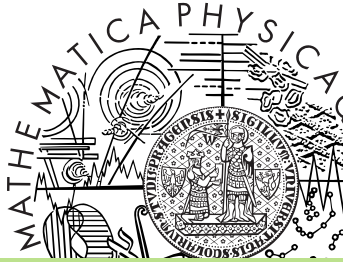


# Nested CVaR risk measure

- Given by following equation:

$$\rho^n(\mathbf{Z}) = \text{CVaR}_\alpha[\cdot|\mathcal{F}_1] \circ \cdots \circ \text{CVaR}_\alpha[\cdot|\mathcal{F}_{T-1}]\left(\sum_{t=1}^{T} Z_t\right)$$

- The interpretation is not straightforward
  - can be viewed as the cost we would be willing to pay at the first stage instead of incurring the sequence of random costs $Z_1, \ldots, Z_T$
  - cf. Ruszczynski [2010]
- Satisfies the time consistency property by construction



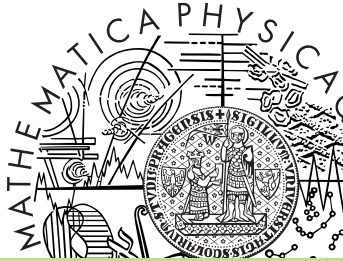
# Nested CVaR model

- Given risk coefficients $\lambda_t$ and random loss variable $Z$ we define:

$$\rho_{t,\boldsymbol{\xi}_{[t-1]}}[Z] = (1-\lambda_t)\,\mathbb{E}\left[Z|\boldsymbol{\xi}_{[t-1]}\right] + \lambda_t\,\mathrm{CVaR}_{\alpha_t}\left[Z|\boldsymbol{\xi}_{[t-1]}\right]$$

- Nested model can be written:

$$\min_{\mathbf{A}_1\mathbf{x}_1=\mathbf{b}_1,\mathbf{x}_1\geq 0} \mathbf{c}_1^\top\mathbf{x}_1 + \rho_{2,\boldsymbol{\xi}_{[1]}}\left[\min_{\mathbf{B}_2\mathbf{x}_1+\mathbf{A}_2\mathbf{x}_2=\mathbf{b}_2,\mathbf{x}_2\geq 0} \mathbf{c}_2^\top\mathbf{x}_2 + \cdots\right.$$
$$\left.\cdots + \rho_{T,\boldsymbol{\xi}_{[T-1]}}\left[\min_{\mathbf{B}_T\mathbf{x}_{T-1}+\mathbf{A}_T\mathbf{x}_T=\mathbf{b}_T,\mathbf{x}_T\geq 0} \mathbf{c}_T^\top\mathbf{x}_T\right]\right]$$



# Nested CVaR model

- Allows to develop dynamic programming equations, using:

$$\text{CVaR}_\alpha [Z] = \min_u \left[ u + \frac{1}{\alpha} \mathbb{E} [Z - u]_+ \right]$$

- Denote $Q_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_{[t]})$, $t = 2, \ldots, T$ as the optimal value of:

$$\begin{aligned} Q_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_{[t]}) = \min_{\mathbf{x}_t, u_t} \; & \mathbf{c}_t^\top \mathbf{x}_t + \lambda_{t+1} u_t + \mathcal{Q}_{t+1}(\mathbf{x}_t, u_t, \boldsymbol{\xi}_{[t]}) \\ \text{s.t.} \; & \mathbf{A}_t \mathbf{x}_t = \mathbf{b}_t - \mathbf{B}_t \mathbf{x}_{t-1} \\ & \mathbf{x}_t \geq 0, \end{aligned}$$

- Recourse function $\mathcal{Q}_{t+1}(\mathbf{x}_t, u_t, \boldsymbol{\xi}_{[t]})$ is given by ($\mathcal{Q}_{T+1}(\cdot) \equiv 0$):

$$\mathbb{E}_{\mathbb{P}_{t+1}[\cdot|\boldsymbol{\xi}_{[t]}]} \left[ (1 - \lambda_{t+1}) Q_{t+1}(\mathbf{x}_t, \boldsymbol{\xi}_{[t+1]}) + \frac{\lambda_{t+1}}{\alpha_{t+1}} \left[ Q_{t+1}(\mathbf{x}_t, \boldsymbol{\xi}_{[t+1]}) - u_t \right]_+ \right.$$

# Comparison with risk-neutral model

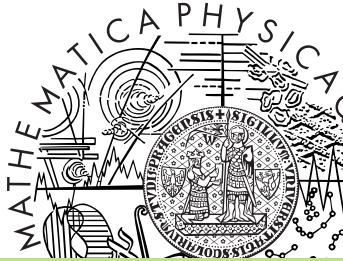
- Consider an additive utility with contribution $u_t(\cdot)$
- For expectation we have:

$$\mathbb{E}\left[u_2(Z_2) + \mathbb{E}\left[u_3(Z_3)|\boldsymbol{\xi}_{[2]}\right]\right] = \mathbb{E}\left[u_2(Z_2)\right] + \mathbb{E}\left[u_3(Z_3)\right]$$

- However, this additive form does not hold for CVaR
- We only have:

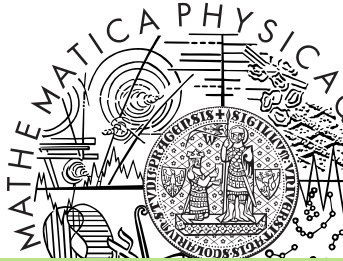
$$\mathrm{CVaR}_\alpha\left[\mathrm{CVaR}_\alpha\left[Z_2 + Z_3|\boldsymbol{\xi}_{[2]}\right]\right] = \mathrm{CVaR}_\alpha\left[Z_2 + \mathrm{CVaR}_\alpha\left[Z_3|\boldsymbol{\xi}_{[2]}\right]\right]$$

$$\leq \mathrm{CVaR}_\alpha\left[Z_2\right] + \mathrm{CVaR}_\alpha\left[\mathrm{CVaR}_\alpha\left[Z_3|\boldsymbol{\xi}_{[2]}\right]\right]$$

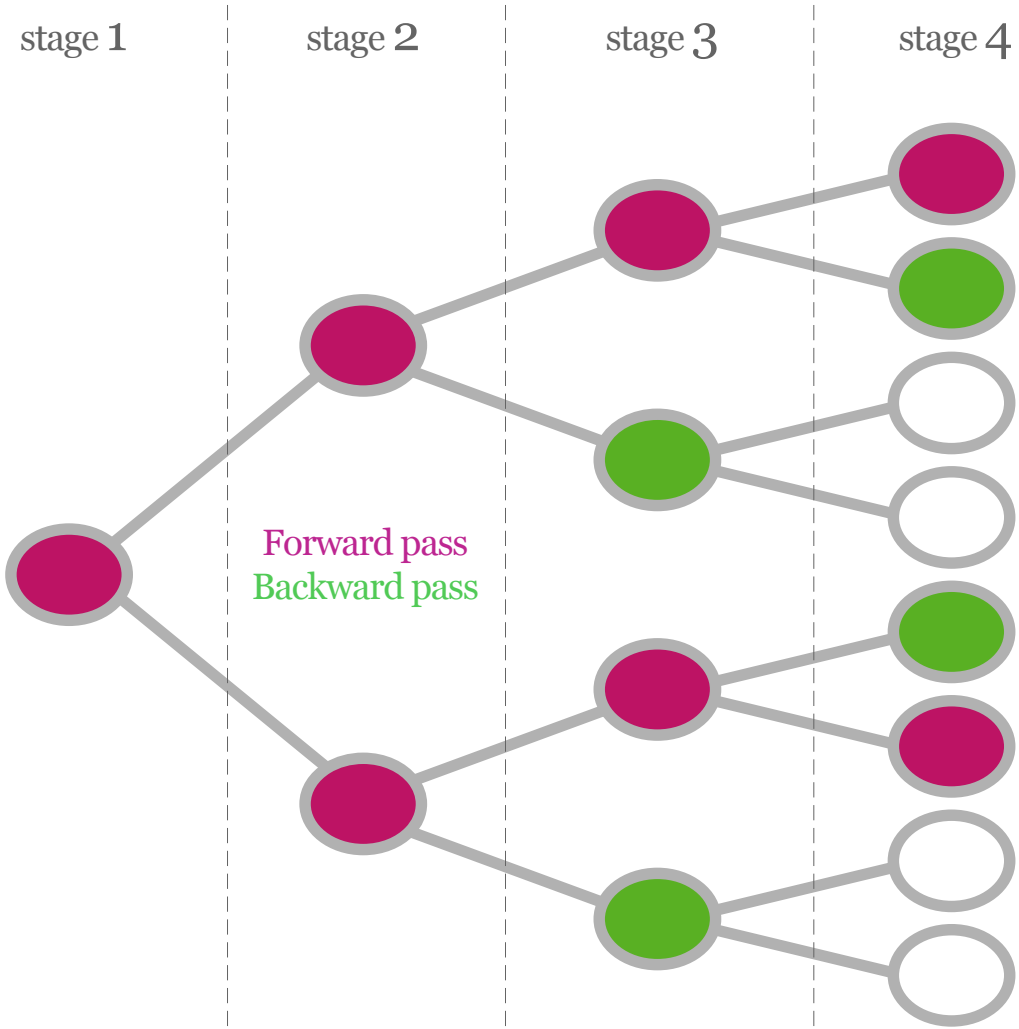


# SDDP algorithm properties

- First designed to solve hydro-scheduling problems
- Relies on the stage-independence assumption
- Each iteration runs with linear complexity
- Provides approximate solution using Benders' cuts
  - Cuts provide polyhedral approximation of the recourse function
  - LP duality - subgradient computed from the dual variables
  - Lower bound
- Policy evaluation procedure
  - Upper bound
- Upper bound requires estimation
  - Precise calculation is impossible for large number of stages
  - Algorithm stops if lower bound is close enough to confidence interval for the upper bound
    - rarely done in a statistically rigorous manner

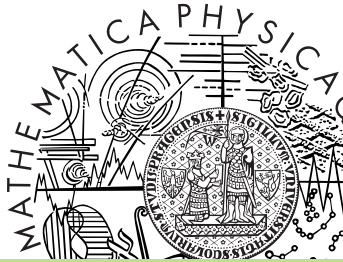


# SDDP scheme



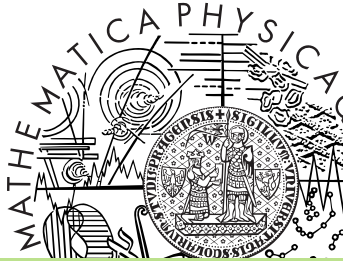
# SDDP algorithm outline

- Because of the stage independence, cuts collected at any node from the stage $t$ are valid for all nodes from that stage
- Algorithm consists of forward and backward iterations
- Forward iteration
  - Samples $\xi^1, \ldots, \xi^J$ sample paths
  - Policy is evaluated using all the cuts collected so far
  - Value of the policy gives the upper bound
- Backward iteration
  - Subset of the scenarios from the forward iteration is chosen
  - For every chosen node the Benders' cut is calculated
    - Using all of its immediate descendants (not just scenarios from the forward pass)
  - Optimal value of the root problem gives the lower bound
- The bounds are compared and the process is repeated



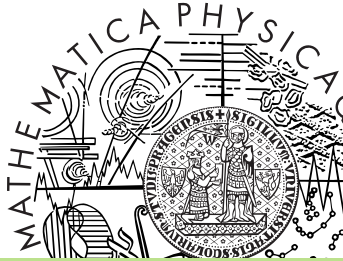
# Our SDDP implementation

- Using the nested CVaR model
- Using own software developed in C++
- CPLEX and COIN-OR used as solvers for the LPs
- Stock assets allocation problem used as the example
- SDDP applied to a sampled tree from the continuous problem
- The algorithm can be implemented for parallel processing
  - We have not done so
- Testing data from Czech Stock Market
- Log-normal distribution of returns is assumed
- Risk aversion coefficients set to $\lambda_t = \frac{t-1}{T}$, $\lambda_t = \frac{1}{2}$
- Tail probability for CVaR set to 5% for all stages
  - May be set differently for each stage



# Inter-stage independence

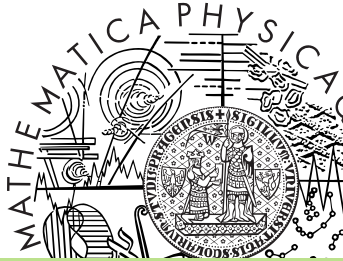
- In order to use SDDP some form of independence is required
  - Efficient algorithms usually rely on an inter-stage independence assumption
  - Otherwise, memory issues arise even for modest number of stages
- This assumption can be weakened
  - One extension is to incorporate an additive dependence model
    - See Infanger & Morton [1996]
  - Another approach to bring dependence into the model is the use of a Markov chain in the model
    - See Philpott & Matos [2012]
  - Yet another approach couples a "small" scenario tree with general dependence structure with a second tree that SDDP can handle
    - See Rebennack et al. [2012]



# Asset allocation model

- At stage $t$ we observe the price ratio between the new price and the old price $\mathbf{p}_t$
- $\mathbf{x}_t$ contains the optimal allocation (in USD, say)
- The total portfolio value is tracked as a multiple of the initial value
- Dynamic programming equations are very simple:

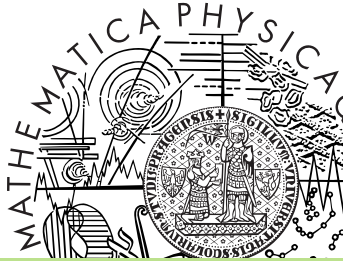
$$\begin{aligned} \min_{\mathbf{x}_t, u_t} \; & -\mathbf{1}^\top \mathbf{x}_t + \lambda_{t+1} u_t + \mathcal{Q}_{t+1}(\mathbf{x}_t, u_t) \\ \text{s.t. } & \mathbf{p}_t^\top \mathbf{x}_{t-1} - \mathbf{1}^\top \mathbf{x}_t = 0 \\ & \mathbf{x}_t \geq 0 \end{aligned}$$



# Asset allocation model with transaction costs

- Relative fee $f_t$ (e.g., 0.3% of the asset price)
- Transaction costs of $f_t \mathbf{1}^\top |\mathbf{x}_t - \mathbf{x}_{t-1}|$
- Linearizing we obtain the following model:

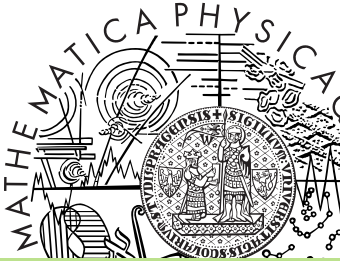
$$
\begin{aligned}
Q_t(\mathbf{x}_{t-1}, \boldsymbol{\xi}_t) = \min_{\mathbf{x}_t, \mathbf{z}_t, u_t} \quad & -\mathbf{1}^\top \mathbf{x}_t + \lambda_{t+1} u_t + \mathcal{Q}_{t+1}(\mathbf{x}_t, u_t) \\
\text{s.t.} \quad & \mathbf{1}^\top \mathbf{x}_t + f_t \mathbf{1}^\top \mathbf{z}_t = \mathbf{p}_t^\top \mathbf{x}_{t-1} \\
& \mathbf{z}_t - \mathbf{x}_t \geq -\mathbf{x}_{t-1} \\
& \mathbf{z}_t + \mathbf{x}_t \geq \mathbf{x}_{t-1} \\
& \mathbf{x}_t \geq 0
\end{aligned}
$$



# Empirical study

- Week-to-week ratios from Prague Stock Exchange, November 2007 to March 2012

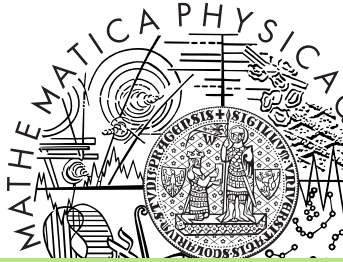
| **asset** | **mean** | **std. deviation** |
|---|---|---|
| AAA | 0.9980 | 0.0716 |
| CETV | 0.9929 | 0.0995 |
| ČEZ | 0.9994 | 0.0406 |
| ERSTE GROUP BANK | 0.9983 | 0.0795 |
| KOMERČNÍ BANKA | 1.0018 | 0.0543 |
| ORCO | 0.9899 | 0.0938 |
| PEGAS NONWOVENS | 0.9995 | 0.0398 |
| PHILIP MORRIS ČR | 1.0035 | 0.0368 |
| TELEFÓNICA C.R. | 1.0004 | 0.0266 |
| UNIPETROL | 0.9986 | 0.0506 |



# Empirical study

- Two different settings of risk coefficients, $\lambda_t = \frac{1}{2}$ and $\lambda_t = \frac{t-1}{T}$
- Both cases, with transaction costs of 0.3% and without transaction costs considered
- In all the testing cases, only three assets play a significant role in our portfolio: ČEZ, PHILIP MORRIS ČR and TELEFÓNICA C.R.

| **stages** | **descendants per node** | **total scenarios** |
|---|---|---|
| 2 | $50,000$ | $50,000$ |
| 3 | $1,000$ | $1,000,000$ |
| 5 | $1,000$ | $10^{12}$ |



# Empirical study

| stages | ČEZ | PHILL | TELE |
|---|---|---|---|
| 2 | 0.0663 (0.0087) | 0.3169 (0.0081) | 0.6168 (0.0092) |
| 3 | 0.0510 (0.0459) | 0.3112 (0.0537) | 0.6273 (0.0707) |
| 5 | 0.0450 (0.0307) | 0.3340 (0.0268) | 0.6043 (0.0571) |

Table : Optimal decisions (std. deviations) with $f_t = 0$ and $\lambda_t = \frac{1}{2}$

| stages | ČEZ | PHILL | TELE |
|---|---|---|---|
| 2 | 0.0663 (0.0087) | 0.3169 (0.0081) | 0.6168 (0.0092) |
| 3 | 0.0597 (0.0645) | 0.3429 (0.0650) | 0.5792 (0.0920) |
| 5 | 0.0392 (0.0415) | 0.4325 (0.0678) | 0.4975 (0.0652) |

Table : Optimal decisions (std. deviations) with $f_t = 0$ and $\lambda_t = \frac{t-1}{T}$

# Empirical study

| stages | ČEZ | PHILL | TELE |
|---|---|---|---|
| 2 | 0.0663 (0.0087) | 0.3169 (0.0081) | 0.6168 (0.0092) |
| 3 | 0.0405 (0.0279) | 0.2977 (0.0322) | 0.6438 (0.0409) |
| 5 | 0.0643 (0.0208) | 0.3115 (0.0231) | 0.6149 (0.0323) |

Table : Optimal decisions (std. deviations) with $f_t = 0.3\%$ and $\lambda_t = \frac{1}{2}$

| stages | ČEZ | PHILL | TELE |
|---|---|---|---|
| 2 | 0.0663 (0.0087) | 0.3169 (0.0081) | 0.6168 (0.0092) |
| 3 | 0.0412 (0.0389) | 0.3175 (0.0258) | 0.6192 (0.0403) |
| 5 | 0.0493 (0.0240) | 0.3274 (0.0346) | 0.6168 (0.0293) |

Table : Optimal decisions (std. deviations) with $f_t = 0.3\%$ and $\lambda_t = \frac{t-1}{T}$

# Empirical study

- Zero transaction costs
  - No significant difference between the optimal portfolios for 2, 3 or 5 stage in the constant risk coefficients setting
  - In the second case we can see a slight movement to the riskier asset in the first-stage decision (PHILIP MORRIS ČR)
- Nonzero transaction costs
  - In accordance with our model transaction costs have no effect in 2-stage models
  - Presence of the transaction costs reduces the differences found in the previous case with varying risk coefficients
  - Varying risk coefficients require the investor to change the portfolio in every stage significantly
  - Impact of transaction costs should be weaker in cases where stages cover longer time periods instead of just weeks

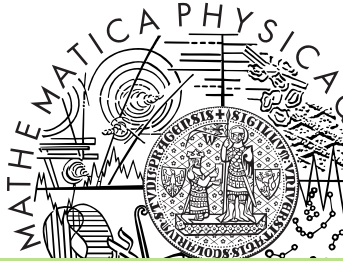
# Upper bound overview

- Risk-neutral problems
  - The value of the current optimal policy can be estimated easily
  - Expectation at each node can be estimated by single chosen descendant
- Risk-averse problems
  - To estimate the CVaR value we need more descendants in practice
  - Leads to intractable estimators with exponential computational complexity
- Current solution (to our knowledge)
  - Run the risk-neutral version of the same problem and determine the number of iterations needed to stop the algorithm, then run the same number of iterations on the risk-averse problem
  - Inner approximation scheme proposed by Philpott et al. [2013]
    - Works with different policy than the outer approximation
    - Probably the best alternative so far
- New solution by Kozmík & Morton

# Conclusion

Thank you for your attention!

Václav Kozmík
vaclav@kozmik.cz



# References


- ARTZNER, P., DELBAEN, F., EBER, J.-M., HEATH, D. and KU, H. (2007): *Coherent multiperiod risk adjusted values and Bellman's principle*, Annals of Operations Research 152, pp. 2–22
- BAYRAKSAN, G. and MORTON, D. P. (2009): *Assessing Solution Quality in Stochastic Programs via Sampling*, Tutorials in Operations Research, pp. 102–122, INFORMS, ISBN 978-1-877640-24-7.
- BENDERS, J. F. (1962): *Partitioning procedures for solving mixed-variables programming problems*, Numer. Math. 4 , pp. 238–252
- BIRGE, J. R. and LOUVEAUX, F. V. (1988): *A multicut algorithm for two-stage stochastic linear programs*, European J of Oper Res 34, pp. 384–392.

# References


- HIGLE, J. L., and SEN, S. (1991): *Stochastic Decomposition: An Algorithm for Two-Stage Linear Programs with Recourse*, Math of Oper Res 16, pp. 650–669.
- INFANGER, G. and MORTON, D. P. (1996): *Cut sharing for multistage stochastic linear programs with interstage dependency*, Mathematical Programming 75 pp. 241-256.
- PEREIRA, M. V. F. and PINTO, L. M. V. G. (1991): *Multi-stage stochastic optimization applied to energy planning*, Mathematical Programming 52, pp. 359–375.
- PHILPOTT, A. B., DE MATOS, V. L., FINARDI, E. C.: *On Solving Multistage Stochastic Programs with Coherent Risk Measures*. Oper. Res. 61, pp. 957–970 (2013)

# References


- PHILPOTT, A. B. and GUAN, Z. (2008): *On the convergence of sampling-based methods for multi-stage stochastic linear programs*, Operations Research Letters 36, pp. 450–455.
- PHILPOTT, A. B., DE MATOS, V. L.: *Dynamic sampling algorithms for multi-stage stochastic programs with risk aversion*. Eur. J. of Oper. Res. 218, pp. 470–483 (2012)
- REBENNACK, S., FLACH, B., PEREIRA, M. V. F., PARDALOS, P. M.: *Stochastic hydro-thermal scheduling under* $\mathrm{CO}_2$ *emissions constraints*. IEEE Transactions on Power Systems 27, pp. 58–68 (2012)
- ROCKAFELLAR, R. T. and WETS, R. J-B (1991): *Scenarios and policy aggregation in optimization under uncertainty*, Mathematics of Operations Research 16, pp. 119–147.

# References


- ROCKAFELLAR, R. T. and URYASEV, S. (2002): *Conditional value at risk for general loss distributions*, Journal of Banking & Finance 26, pp. 1443–1471
- RUSZCZYNSKI, A. (1986): *A regularized decomposition method for minimizing a sum of polyhedral functions*, Math Prog 35, pp. 305–333.
- RUSZCZYNSKI, A. and SHAPIRO, A. (2006): *Conditional risk mappings*, Mathematics of Operations Research 31, pp. 544–561.
- SHAPIRO, A. (2011): *Analysis of stochastic dual dynamic programming method*, European Journal of Operational Research 209, pp. 63-72.

# References


- SHAPIRO, A. (2009): *On a time consistency concept in risk averse multistage stochastic programming*, Operations Research Letters 37, pp. 143–147
- SHAPIRO, A., DENTCHEVA, D. and RUSZCZYNSKI, A. (2009): *Lectures on Stochastic Programming: Modeling and Theory*, SIAM-Society for Industrial and Applied Mathematics, ISBN 978-0898716870.
- VAN SLYKE, R. M. and WETS, J.-B. (1969): *L-Shaped linear programs with applications to optimal control and stochastic programming*, SIAM Journal of Applied Mathematics 17, pp. 638–663.