Modern Robust Optimization: Opportunities for Enterprise-Wide Optimization

Chrysanthos E. Gounaris
Dept. of Chemical Engineering and
Center for Advanced Process Decision-making
Carnegie Mellon University
gounaris@cmu.edu

Thursday, September 7
EWO Seminar Fall 2017 Series

Uncertainty in EWO Setting

- Uncertainty is inherent in virtually all EWO settings, both strategic and operational ones

- Typical sources of uncertainty:
  - Market behavior
    - e.g., prices, customer demands
  - Unexpected events
    - e.g., disruptions
  - Model-system mismatch
    - e.g., unknown thermodynamics and/or kinetics

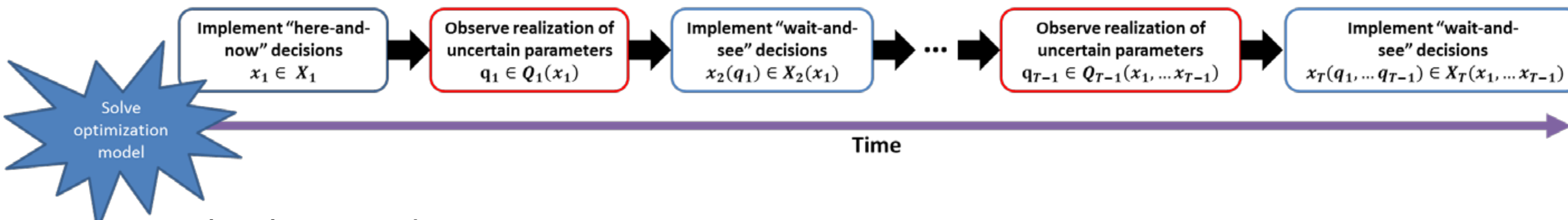
- Optimization in view of only the nominal case can lead to suboptimal/infeasible solutions

  → Need for risk-averse decision-making

Challenges for EWO under Uncertainty

- Large combinatorial component
  - Mixed-integer models with lots of discrete decisions
  - Often custom-built approaches (decomposition, branch-and-cut/price, etc.)
  - Discreteness also prevalent on the side of uncertain parameters

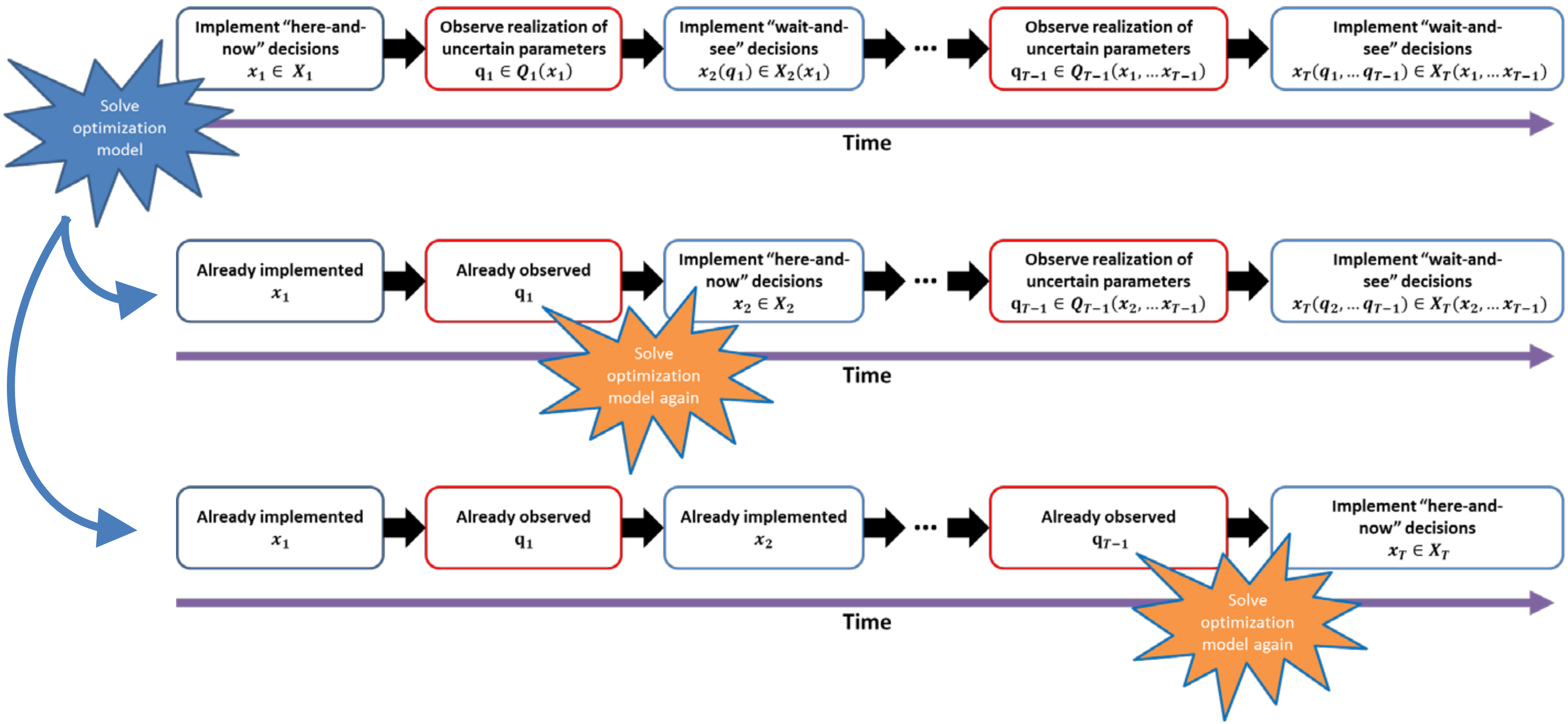
- Multi-stage horizons



  - Multiple paradigms:
    - Invest now, then operate every fiscal year
    - Place/receive order now, then route material through the supply chain
    - Decide control actions for the whole control horizon
  - Set up a DSS to work autonomously in a rolling horizon fashion
    - Re-optimization round-the-clock, as often as you can afford

Multi-stage Horizons (cont'd)

- Re-optimization frequency dictated by tractability
- Number of stages dictated by need to locate "better solutions"



- Need high-quality "wait-and-see" decisions (prepare to adopt them!)
- "Closed-loop feasibility" is more likely as you account for more stages

Robust Optimization

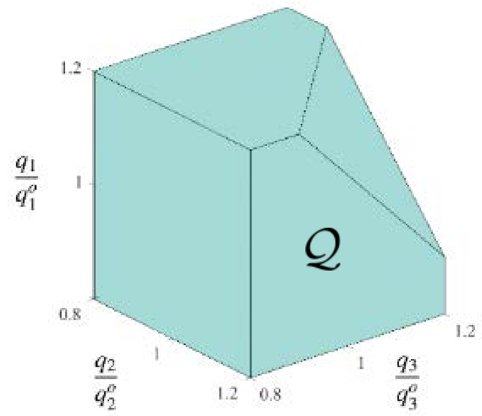
- Game theory interpretation:
  - Player 1 (decision-maker) tries to minimize the objective
  - Player 2 (adversary nature) tries to minimize feasibility margin
- Foundations in "pessimistic optimization" (Wald, Soyster)
  - Attempts to find the solution that would perform best in the "worst-case"
  - The above implies that the solution must remain feasible under all cases we want to insure against (uncertainty set)
- Some references to start with:
  - A. Ben-Tal, L. El Ghaoui and A. Nemirovski (2009). Robust Optimization. Princeton University Press
  - D. Bertsimas, D.B. Brown and C. Caramanis (2011). Theory and Applications of Robust Optimization. SIAM Review, 53(3):464
  - B.L. Gorissen, I. Yanikoglu and D. den Hertog (2015). A Practical Guide to Robust Optimization. Omega, 53:124
  - C.E. Gounaris (2017). Advances in Robust Optimization and Opportunities for Process Operations. In: Proceedings of FOCAPO 2017/CPC IX, Paper ID IF110

1-Stage Robust Optimization

- a.k.a. "Static" Robust Optimization

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & f(x,q) \\ \text{s.t.} \quad & g_i(x,q) \le 0 \quad \forall i \end{aligned}$$

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad \max_{q \in \mathcal{Q}} \quad & f(x,q) \\ \text{s.t.} \quad & g_i(x,q) \le 0 \quad \forall q \in \mathcal{Q} \quad \forall i \end{aligned}$$



Carnegie Mellon ENGINEERING

CAPD CENTER

Solving Robust Optimization Problems

Reformulation Approach

- Address semi-infinite formulation via duality-based treatment of inner problems

$$\begin{aligned} \min_{x\in\mathcal{X}} \quad & \max_{q\in\mathcal{Q}} \; f(x,q) \\ \text{s.t.} \quad & g_i(x,q) \le 0 \quad \forall q \in \mathcal{Q} \quad \forall i \end{aligned}$$

$$\begin{aligned} \min_{x\in\mathcal{X}, z\in\mathbb{R}} \quad & z \\ \text{s.t.} \quad & \max_{q\in\mathcal{Q}} f(x,q) \le z \\ & \max_{q\in\mathcal{Q}} g_i(x,q) \le 0 \quad \forall i \end{aligned}$$

$$\begin{aligned} \min_{x\in\mathcal{X}, z\in\mathbb{R}} \quad & z \\ \text{s.t.} \quad & \min_{p\in\mathcal{P}} f^D(x,p) \le z \\ & \min_{p\in\mathcal{P}} g_i^D(x,p) \le 0 \quad \forall i \end{aligned}$$

✔ The problem can be solved "monolithically," via direct call to an appropriate optimization solver

✘ Restricted to settings where strong duality holds

✘ The size of the problem grows a lot (unnecessarily?) as the uncertainty set dimensionality grows

Robust Cutting-Plane Approach[1]

- Given a (feasible or relaxed) solution, solve a separation problem to identify realizations from within the uncertainty set for which this solution violates a constraint
- Gradually enforce robustness by adding select deterministic constraints using violating realizations

✔ Can accommodate non-standard settings, e.g.,
  - discrete uncertain parameters
  - non MathOpt-based solvers

✘ Requires more elaborate implementation (and lots of "tuning")

[1] *Mutapcic and Boyd, 2009*

Robust Optimization

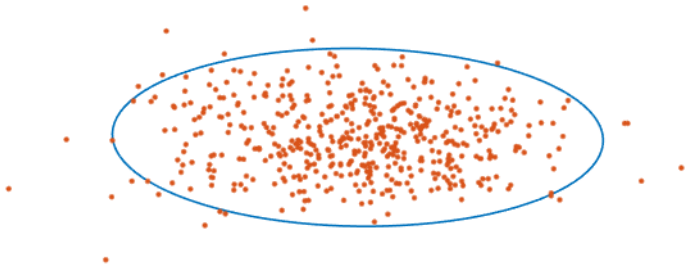
- When to consider:
  - When you can routinely solve the deterministic problem
  - Feasibility is important
    - Safety reasons
    - Cannot monetize infeasibility
  - Large number of parameters that only sparsely participate in constraints
  - Stochastic description of uncertainty meets certain criteria
    - No detailed (joint) probability distributions
    - Well-motivated, strong correlations among parameters

- Size and shape of uncertainty set is chosen by the modeler
  - Usually some norm-based set (e.g., interval, ellipsoid, box)
  - No explicit requirement for scenarios and/or probability distributions
  - If distributional information exists, uncertainty sets can be related to confidence intervals

Examples of Uncertainty Sets

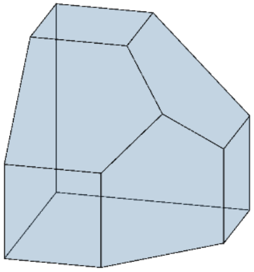
- **Ellipsoids**
  - Gaussian confidence intervals

$$\mathcal{Q}_E = \{q \in \mathbb{R}^n : (q-\mu)^\top \Sigma^{-1}(q-\mu) \leq \beta\}$$



- **Budget sets**
  - aggregate forecasts at various hierarchies

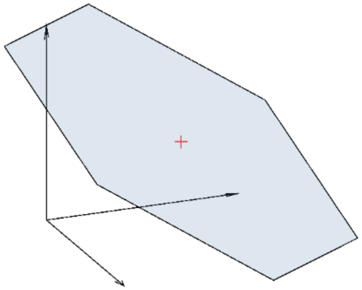
$$\mathcal{Q}_B = \left\{q \in [\underline{q}, \bar{q}]^n : \sum_{i \in B_l} q_i \leq b_l, \ \forall l \in \{1, \ldots, L\}\right\}$$



- **Factor models**
  - bounded disturbances around nominal values

$$\mathcal{Q}_F = \{q \in \mathbb{R}^n : q = q^0 + \Phi\xi \text{ for some } \xi \in \Xi\}$$
$$\Xi = \{\xi \in [-1,1]^F : |e^\top \xi| \leq \beta F\}, \quad F \ll n$$

  - *"zero-net-alpha" models in portfolio optimization*

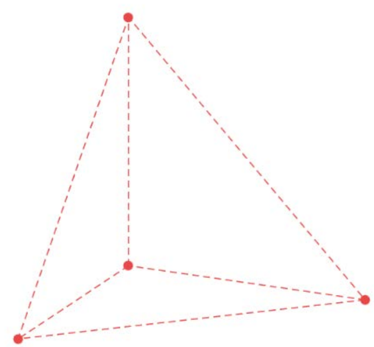


Examples of Uncertainty Sets

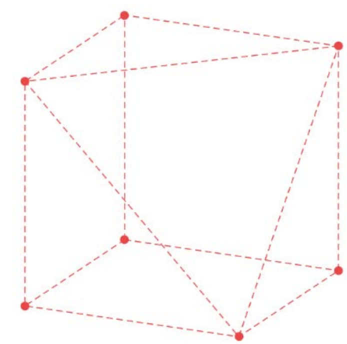
- **Cardinality-constrained sets**
  - "Gamma" sets *(Bertsimas & Sim, 2004)*

$$\mathcal{Q}_\Gamma = \left\{ q \in [q^0, q^0 + \hat{q}] : \exists W \subseteq \{1,\dots,n\},\ |W| \leq \Gamma : q_i = \begin{cases} q_i^0 + \hat{q}_i, & \text{if } i \in W \\ q_i^0, & \text{if } i \notin W \end{cases} \forall i \in \{1,\dots,n\} \right\}$$

$\Gamma = 1$

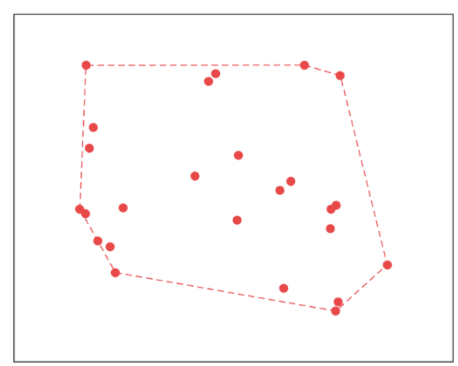


$\Gamma = 2$



- **Discrete sets**
  - Collection of relevant scenarios

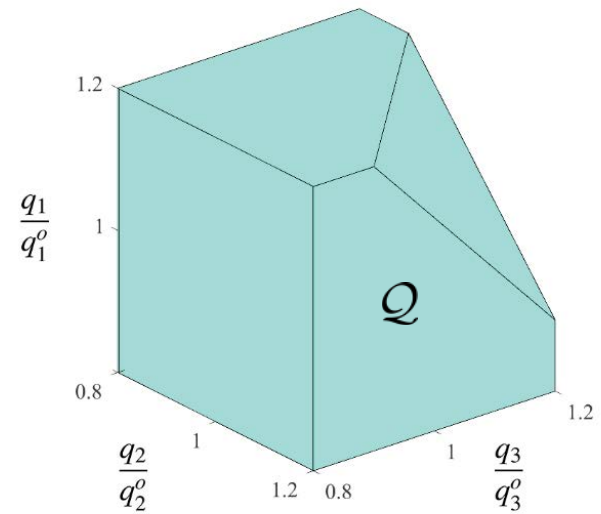
$$\mathcal{Q}_S = \{q^{(1)}, q^{(2)}, \dots, q^{(M)}\}$$



Polyhedral Uncertainty Sets

Advantages:

- Numerically tractable, maintaining class of deterministic counterpart
- Derivable from historical data via machine learning techniques
- Can always be used as approximations of non-polyhedral sets



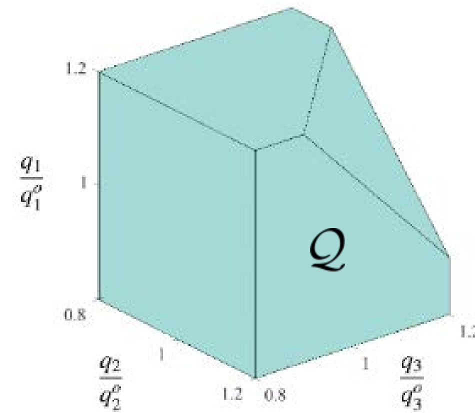
$$\mathcal{Q} = \left\{ \begin{array}{c} q \in \mathbb{R}^{nq}, \ p \in \{0,1\}^{np} : \\ Hq + Gp \leq d \\ q^L \leq q \leq q^U \end{array} \right\}$$

1-Stage Robust Optimization

- a.k.a. "Static" Robust Optimization

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & f(x, q) \\ \text{s.t.} \quad & g_i(x, q) \leq 0 \quad \forall i \end{aligned}$$

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad \max_{q \in \mathcal{Q}} \quad & f(x, q) \\ \text{s.t.} \quad & g_i(x, q) \leq 0 \quad \forall q \in \mathcal{Q} \quad \forall i \end{aligned}$$



- Main limitation: All decisions are considered as "here-and-now" (irrespectively of whether the application mandates this or not)

- Consequently, we cannot enforce equalities involving uncertain parameters (e.g., mass balances with uncertain reaction rates)

Handling Equalities with Static RO

- SRO affords us only a single value (solution) for each decision variable, making it hard to satisfy an equality constraint that references an uncertain parameter for all its realizations

$$\max_{x_1, x_2 \in \mathbb{R}} \quad x_1 + x_2$$
$$\text{s.t.} \quad q\, x_1 + x_2 = 0 \ \ \forall\, q \in \{q_1, q_2, \cdots q_N\}$$

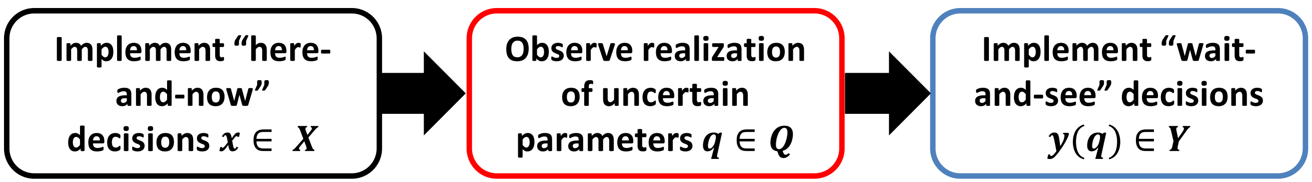
$$\max_{x_1, x_2 \in \mathbb{R}} \quad x_1 + x_2$$
$$\text{s.t.} \quad q_1\, x_1 + x_2 = 0$$
$$q_2\, x_1 + x_2 = 0$$
$$\cdots$$
$$q_N\, x_1 + x_2 = 0$$

$$x_1^* = 0$$
$$x_2^* = 0$$

- State-variable elimination could sometimes remedy the issue

2-Stage Robust Optimization

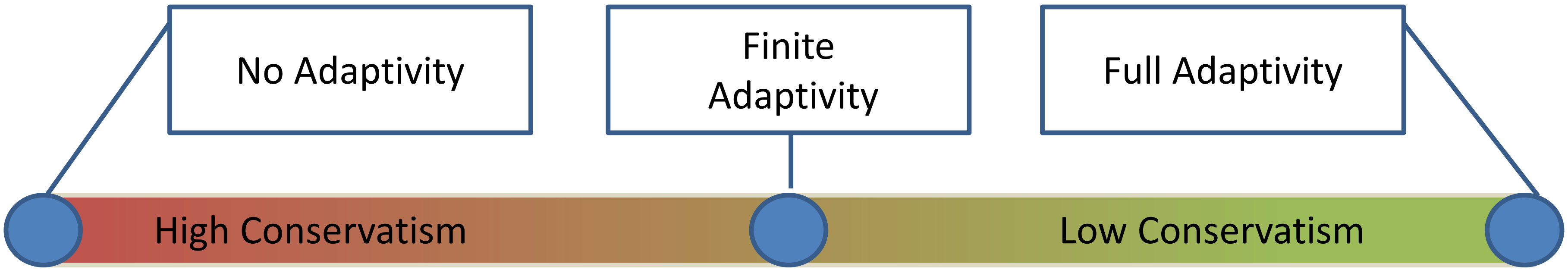
- "Adjustable" Robust Optimization



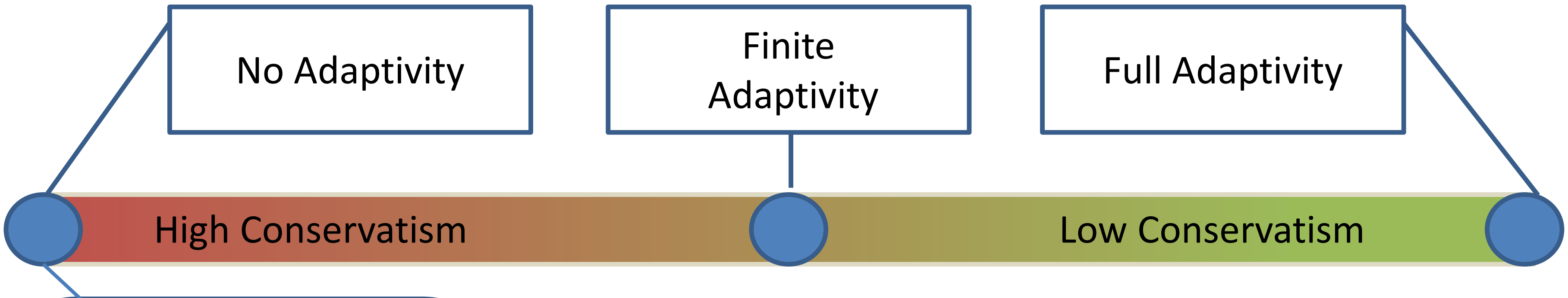
$$\begin{aligned}\min_{x\in\mathcal{X}}\max_{q\in\mathcal{Q}}\min_{y(q)\in\mathcal{Y}} \quad & f\left(x,y,q\right)\\ \text{s.t.} \quad & g_i\left(x,y,q\right)\leq 0 \quad \forall q\in\mathcal{Q} \quad \forall i\end{aligned}$$

- A specific y may be optimal for a scenario q, but suboptimal for a scenario q'
- Best $y(q)$ may correspond to any arbitrary functional dependence
- Ideally we would like to identify the best feasible y for each possible q

2-Stage Robust Optimization



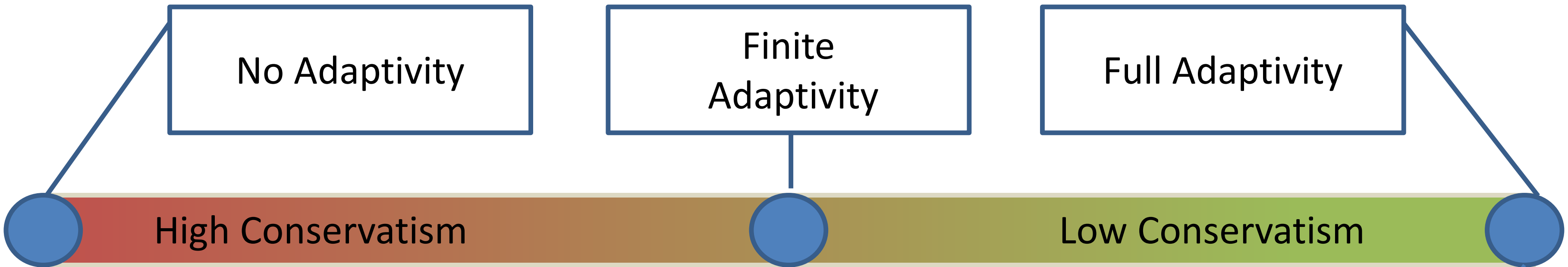
2-Stage Robust Optimization



$$y(q) \leftarrow y$$

- y variables do not depend on the uncertainty
- A single policy must be feasible for any possible realization of uncertainty
- Equivalent to 1-stage RO

2-Stage Robust Optimization

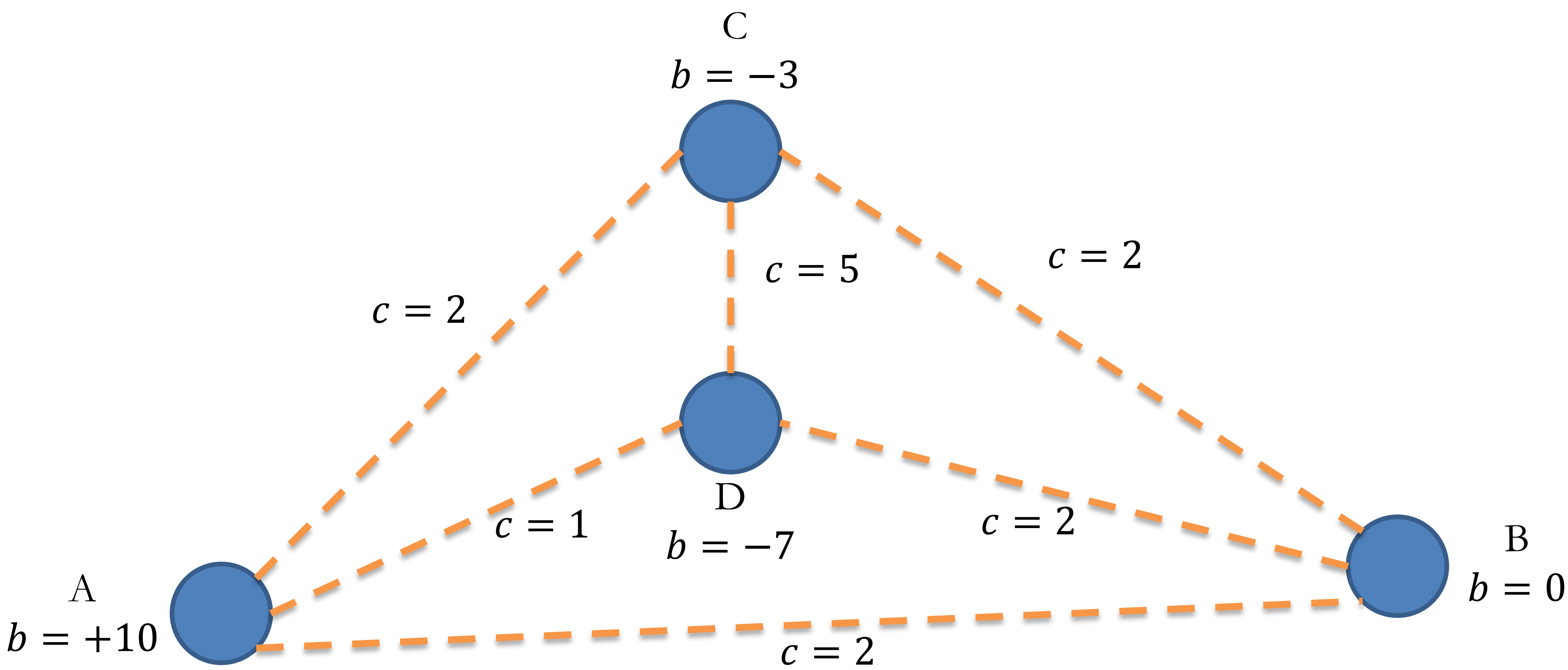


$$y(q)$$

- Add Benders cuts to restrict the first-stage decisions and associated cost of recourse (Thiele, 2009)
- Progressively identify violating scenarios and cover them via new policies, which are to be collectively robust in the end (*Zeng and Zhao, 2013*)

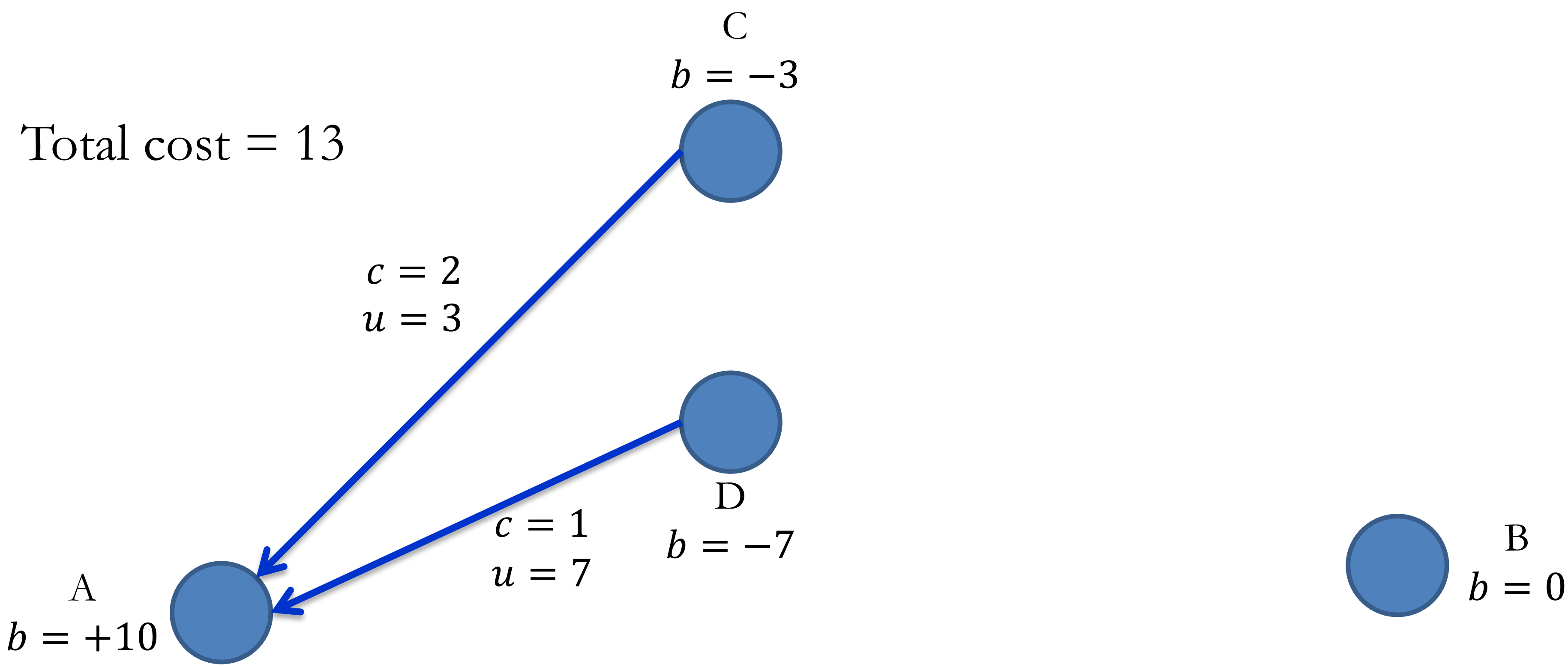
Supply Chain Network Design

- Consider the transportation of a commodity between four locations



Supply Chain Network Design

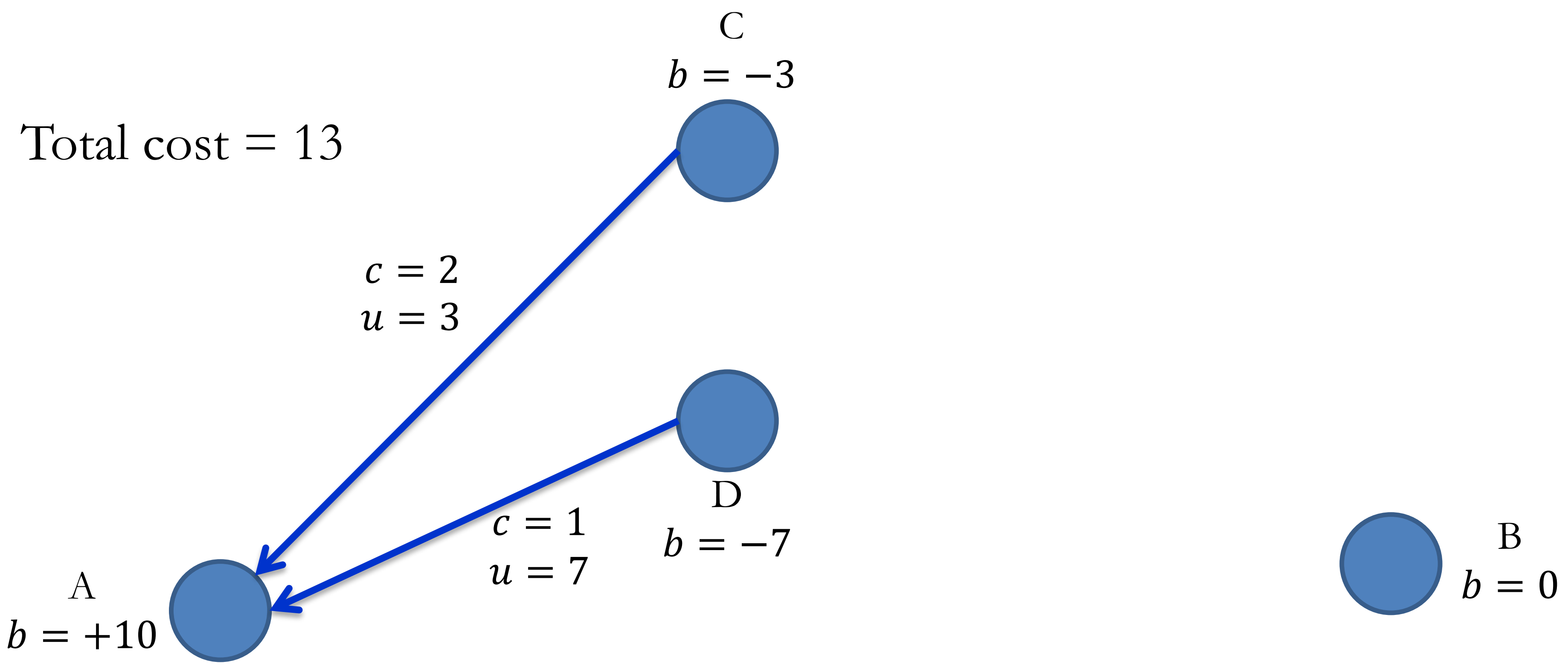
- Optimal solution is to build exactly two links



Supply Chain Network Design

- What if up to 1 transportation link fails?

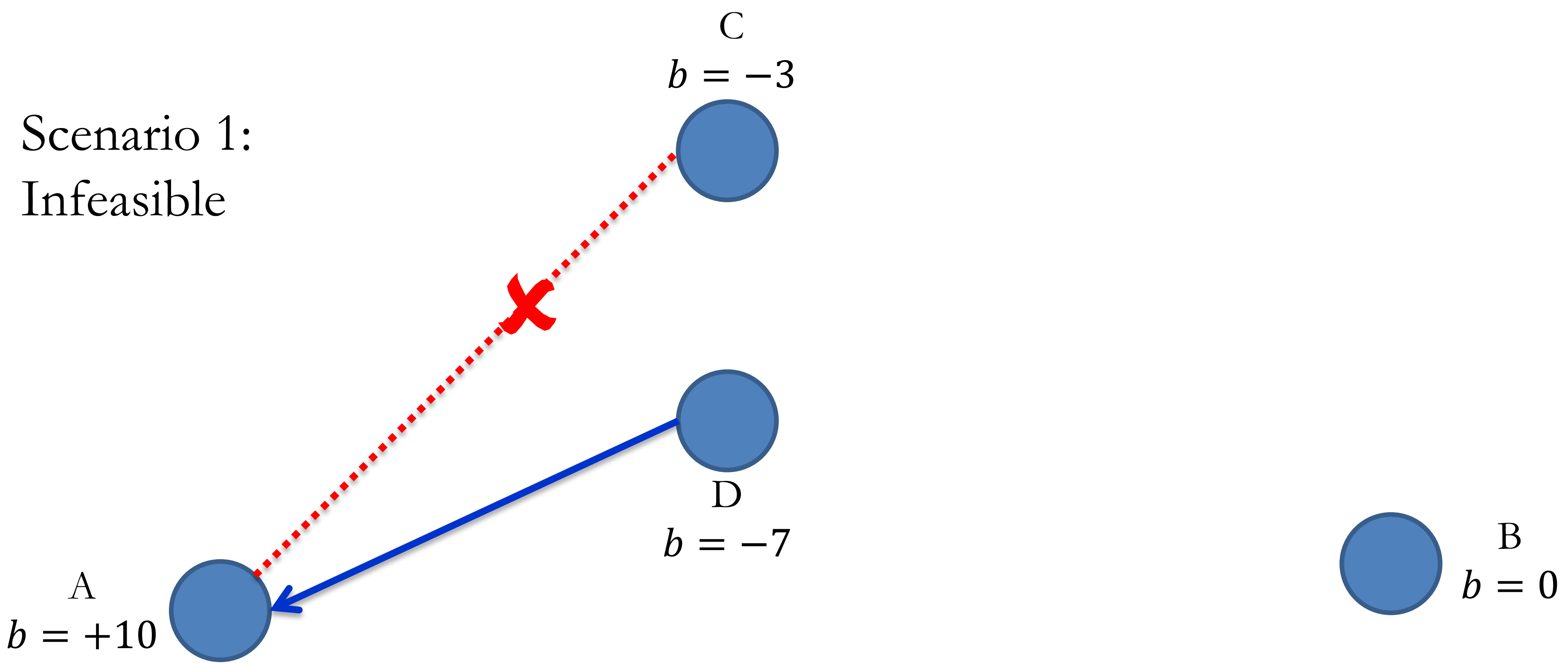
Total cost = 13



Supply Chain Network Design

- What if up to 1 transportation link fails?

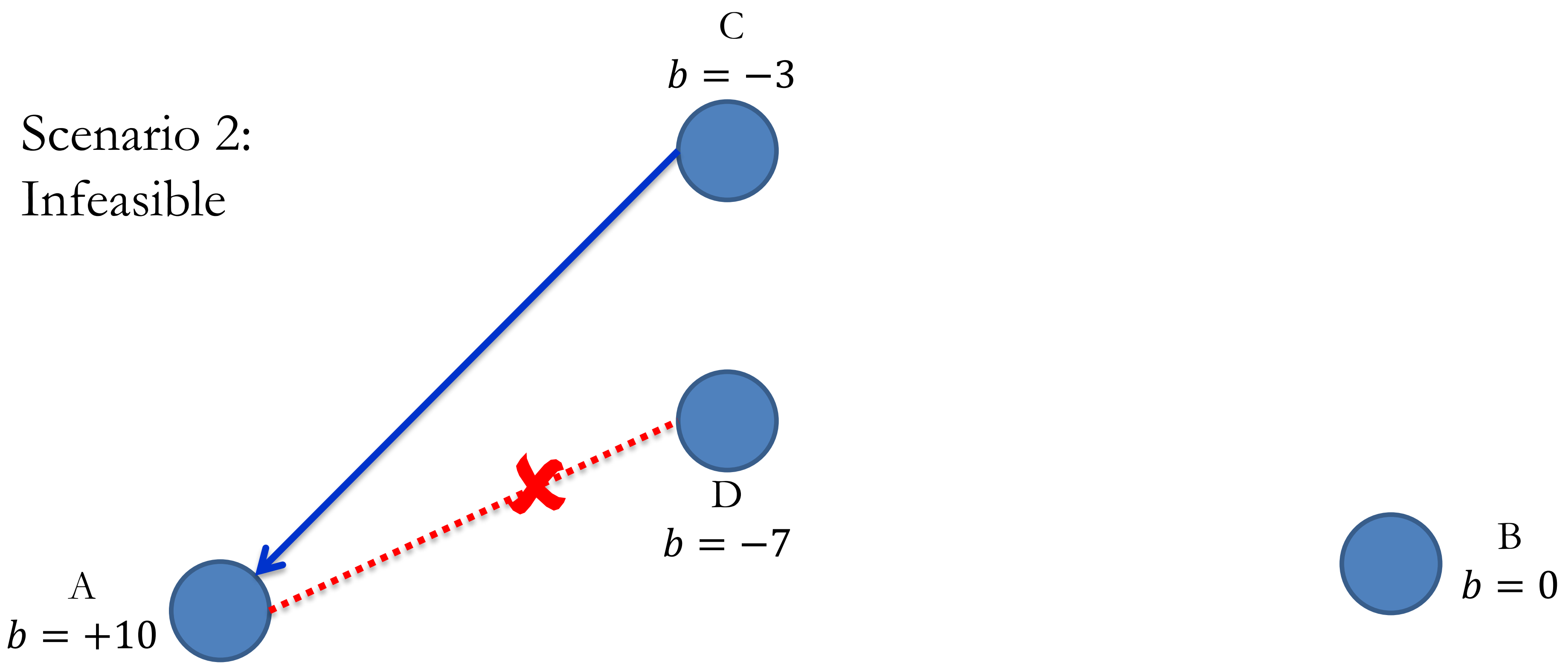
Scenario 1:
Infeasible



Supply Chain Network Design

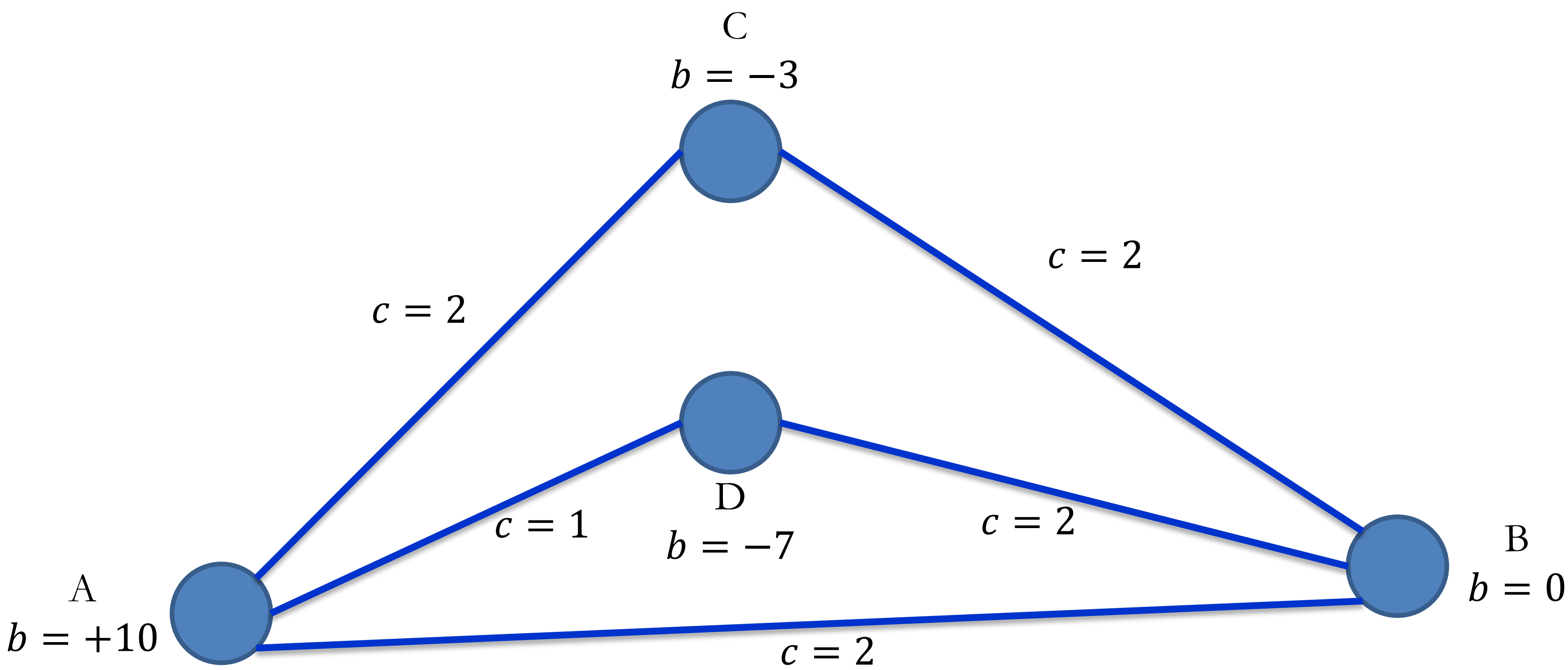
- What if <u>up to 1</u> transportation link fails?

Scenario 2:
Infeasible



Supply Chain Network Design

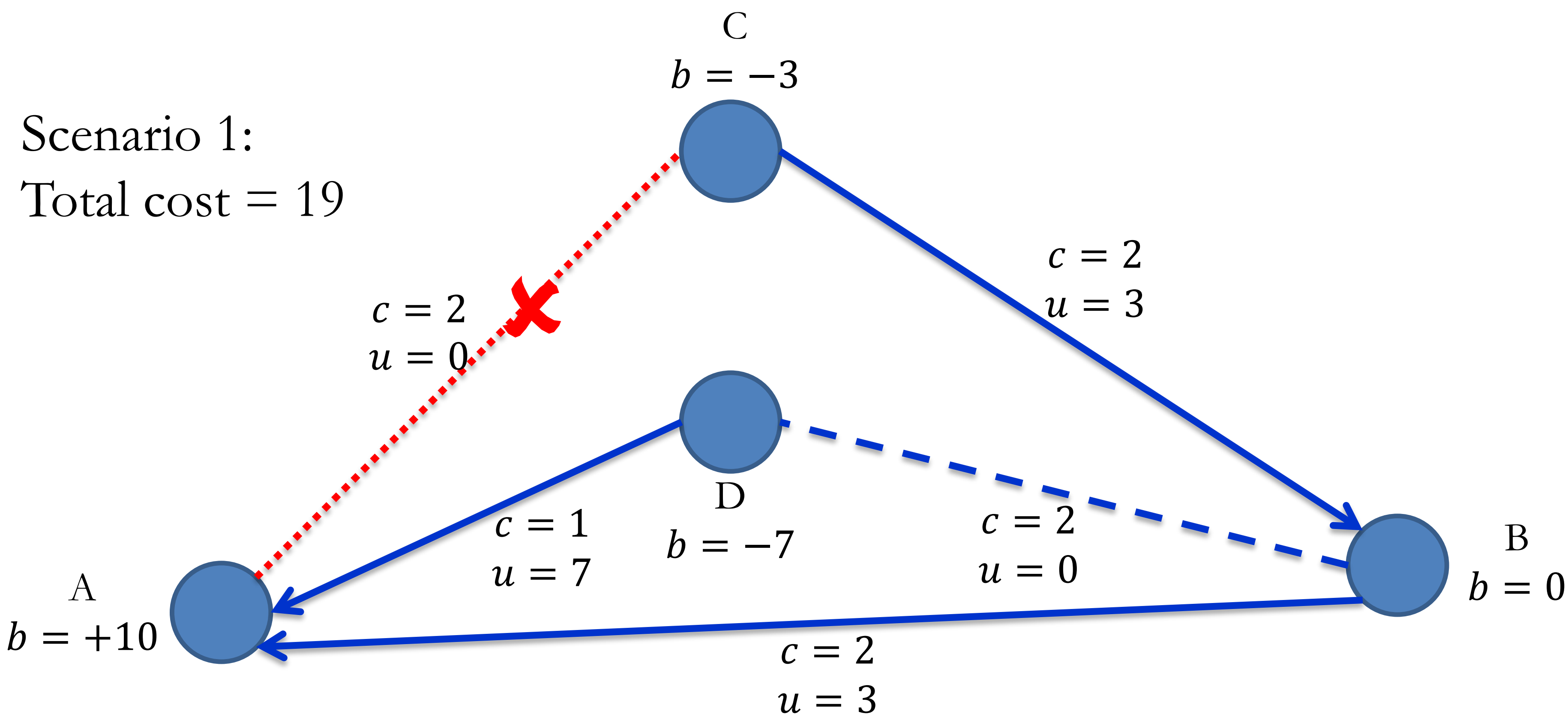
- What if up to 1 transportation link fails?



Supply Chain Network Design

- What if <u>up to 1</u> transportation link fails?

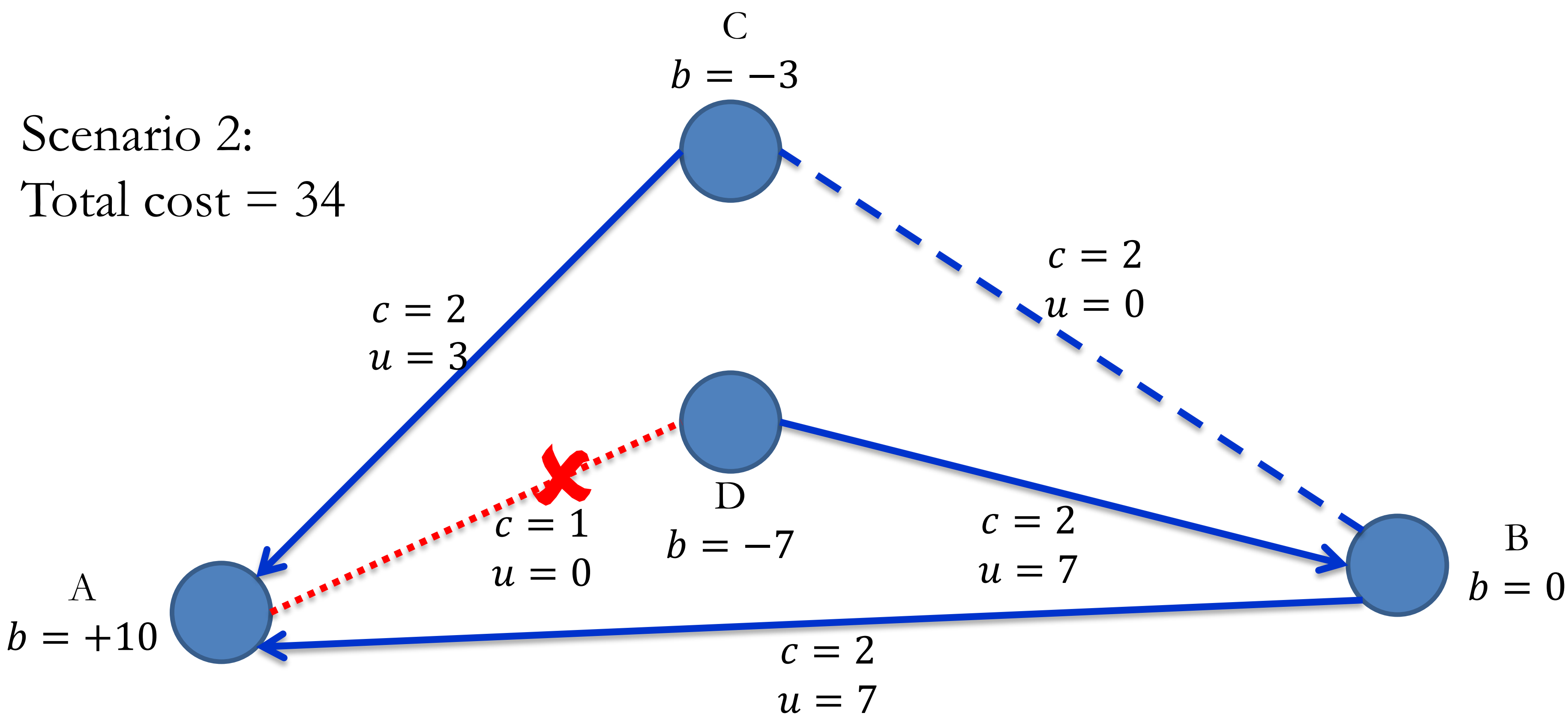
Scenario 1:
Total cost $= 19$



Supply Chain Network Design

- What if <u>up to 1</u> transportation link fails?

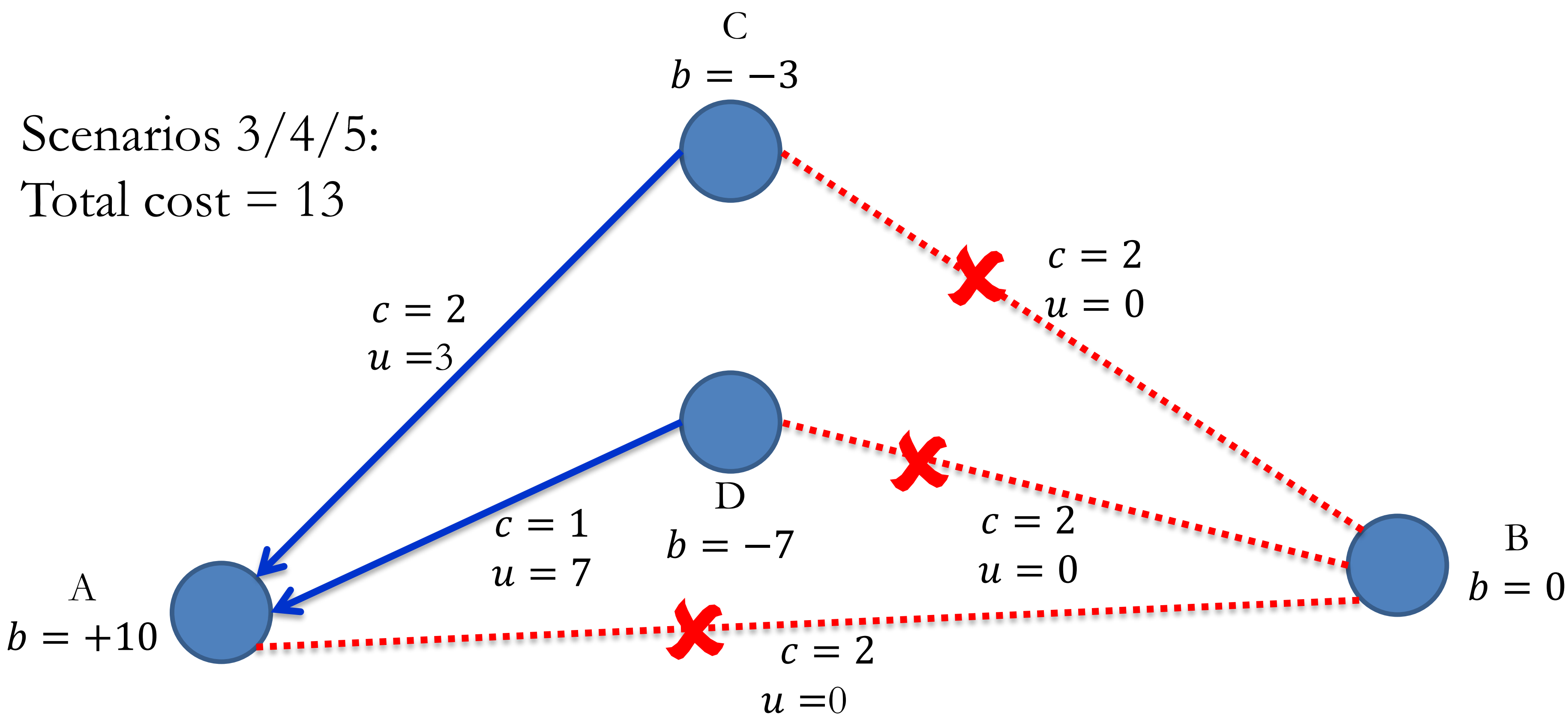
Scenario 2:
Total cost = 34



Supply Chain Network Design

- What if <u>up to 1</u> transportation link fails?

Scenarios 3/4/5:
Total cost = 13

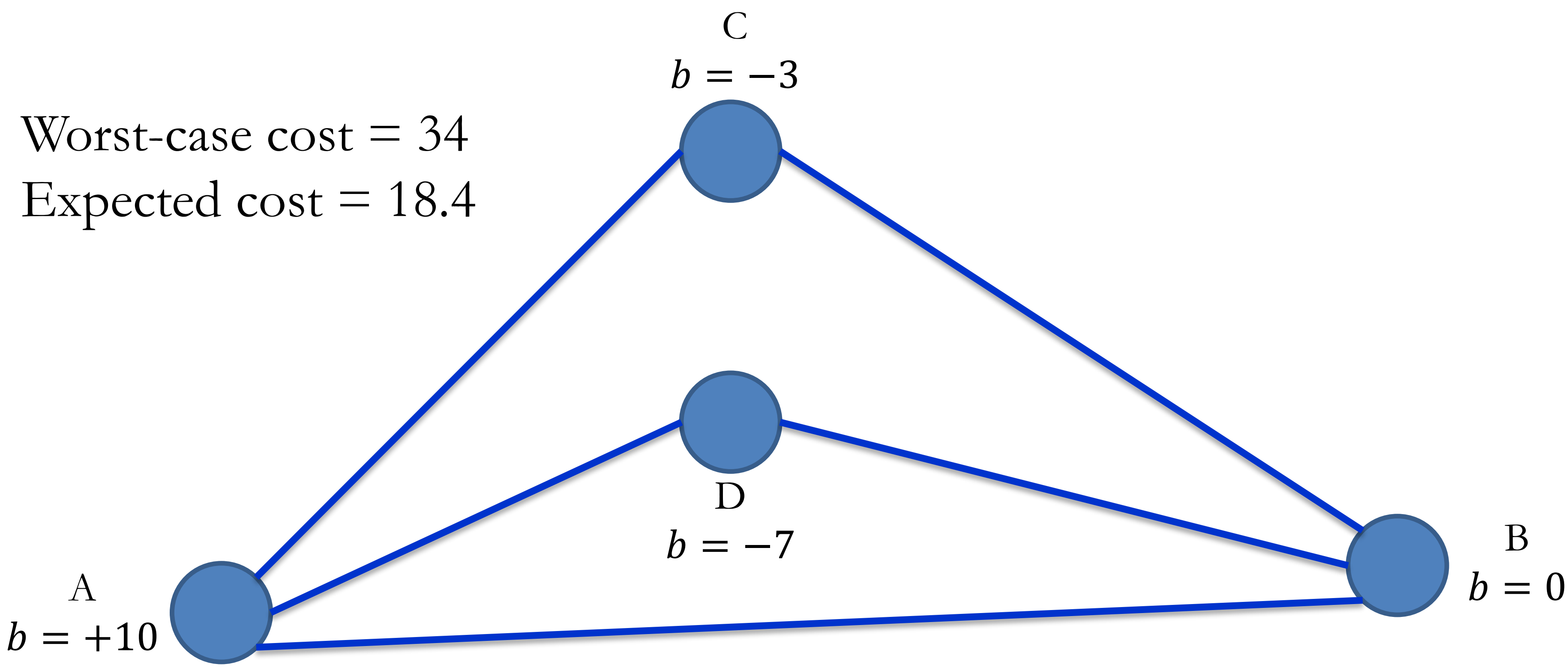


Supply Chain Network Design

- Robust (up to 1 disrupted links) Network ✓

Worst-case cost = 34

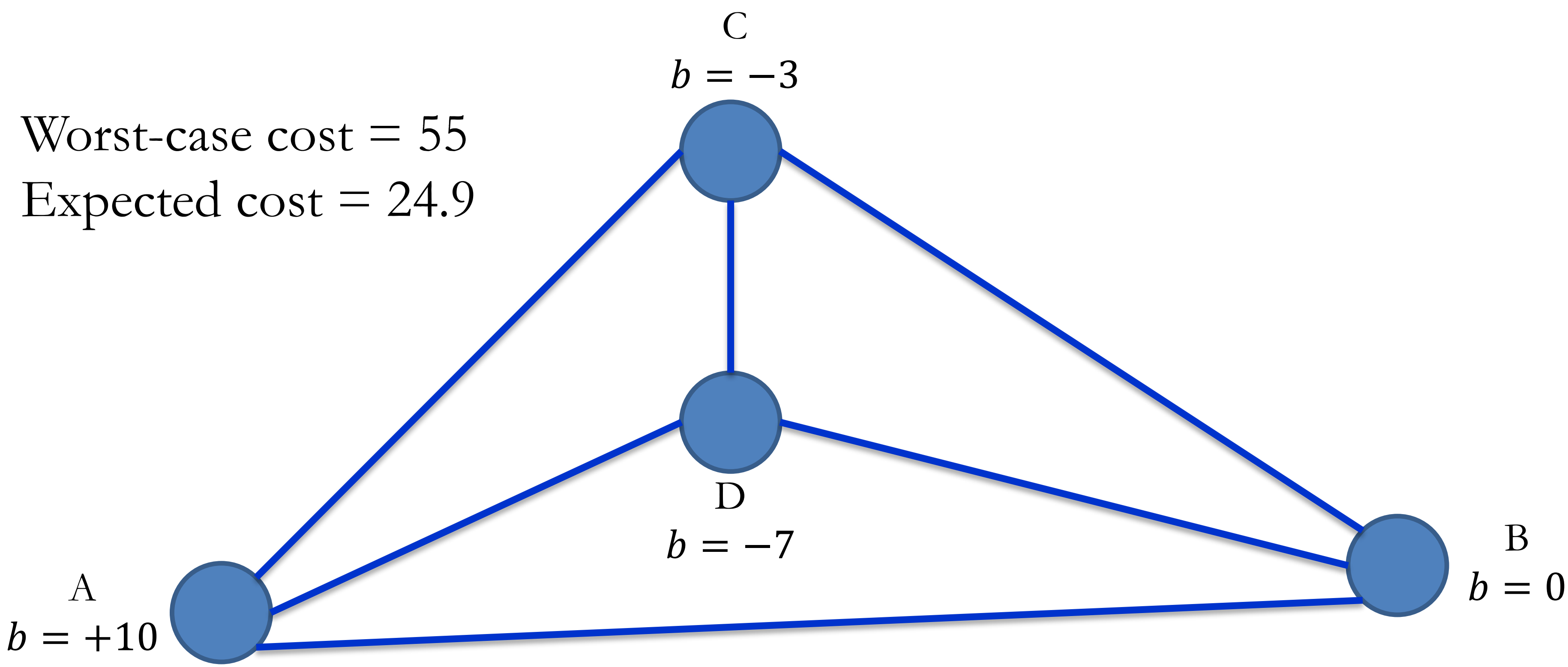
Expected cost = 18.4



Supply Chain Network Design

- Robust (up to 2 disrupted links) Network ✓

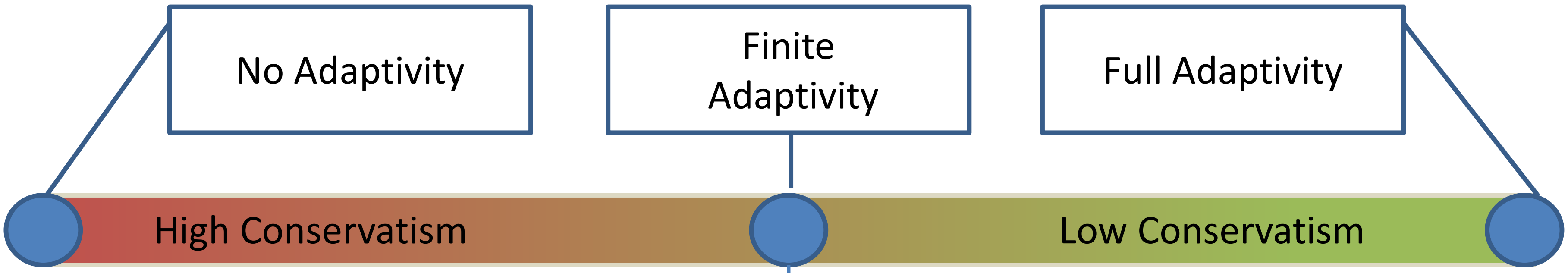
Worst-case cost = 55
Expected cost = 24.9



Apply C&CG to solve this problem on a larger scale!

L.R. Matthews, C.E. Gounaris and Y.G. Kevrekidis (2017). Designing Networks with Uncertain Edge Failures Using Two-Stage Robust Optimization. Under Review

2-Stage Robust Optimization

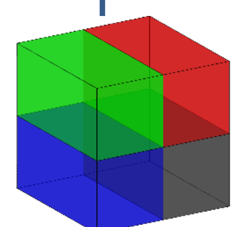


$$y(q) \leftarrow \sum_{k=1}^{K} z_k(q) y_k \qquad \text{(where } z_k \text{ SOS1)}$$

- Partition the uncertainty set into K subsets, and assign each of these to a separate policy (each policy guarantees robustness only against the subset)

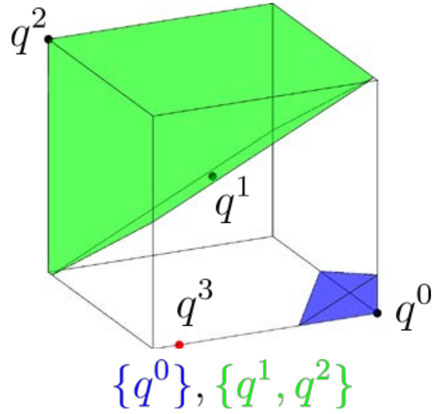
- K impacts level of adaptivity

- Partitioning can be:
  - a–priori fixed (Bertsimas and Caramanis, 2010)
  - iteratively constructed (*Bertsimas and Dunning, 2016; Postek and den Hertog, 2016*)
  - optimal (*Hanasusanto et al., 2015; Subramanyam et al., 2017*)



Optimal Finite Adaptivity

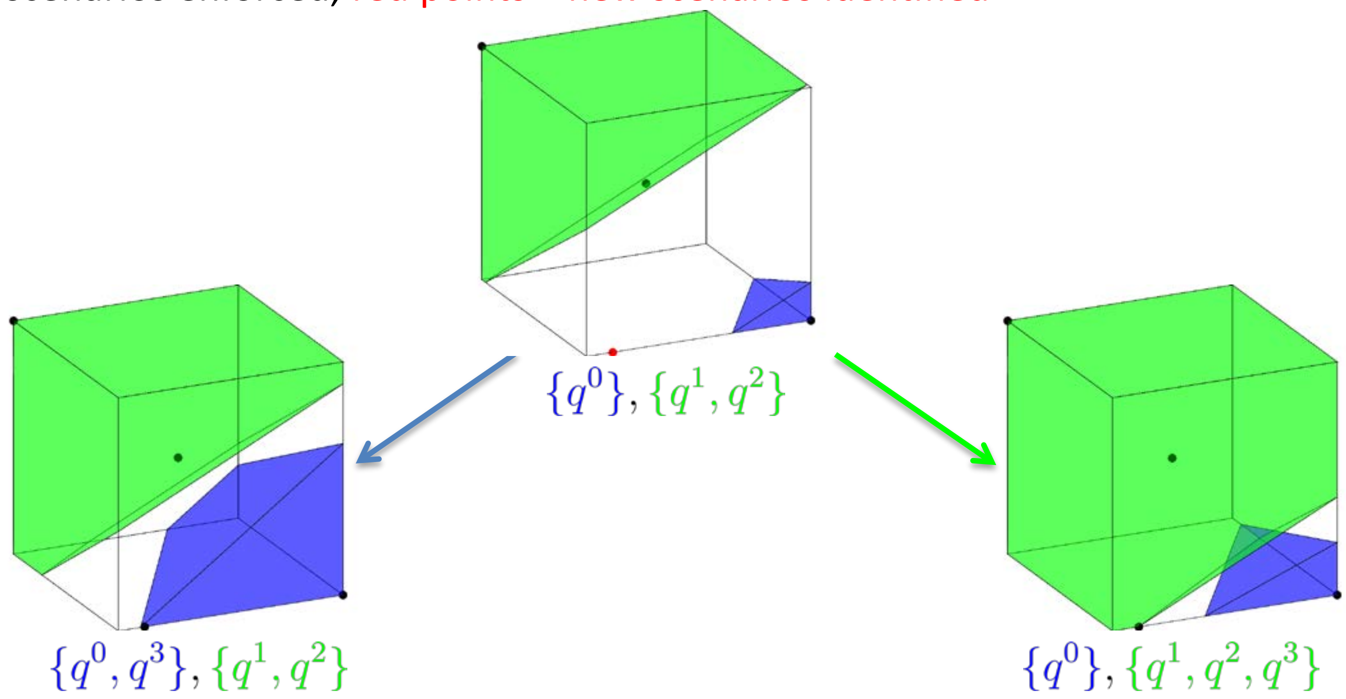
- Example with $K = 2$
  - Blue region = insured by policy 1, green region = insured by policy 2
  - Black points = scenarios enforced, red points = new scenarios identified



Subramanyam, *et al.*, 2017

Optimal Finite Adaptivity

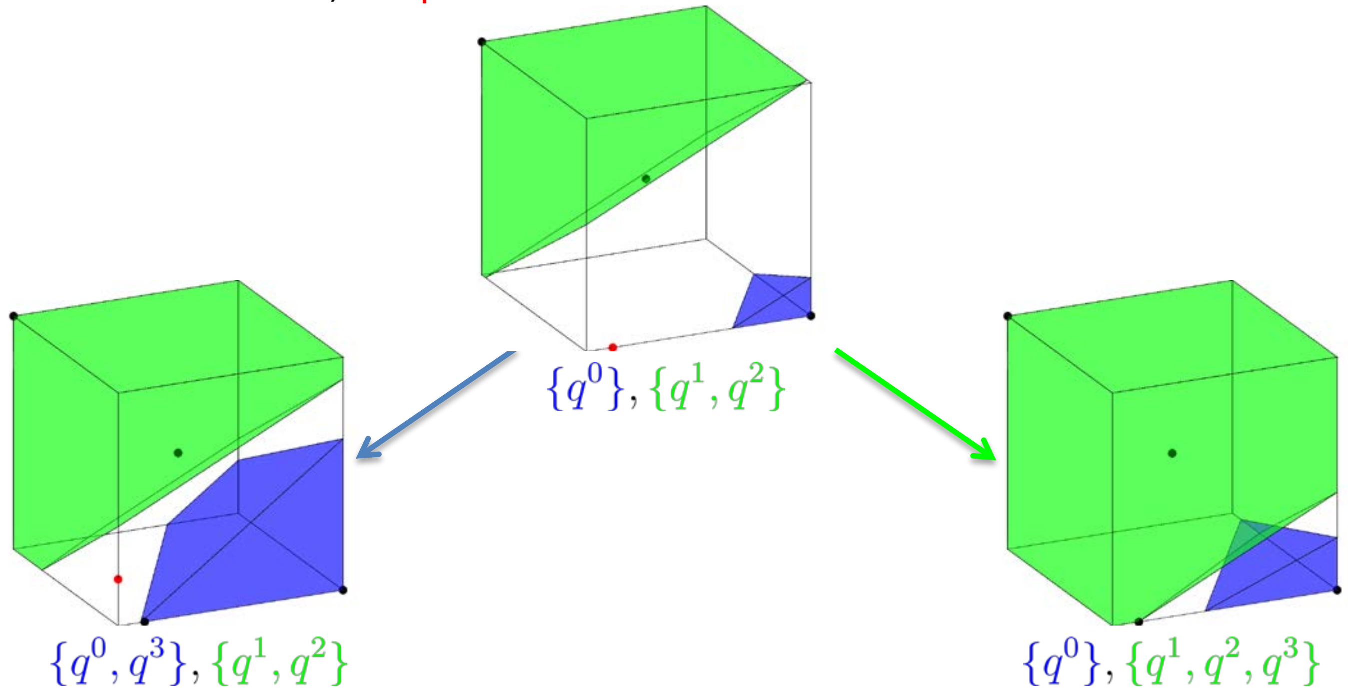
- Example with $K = 2$
  - Blue region = insured by policy 1, green region = insured by policy 2
  - Black points = scenarios enforced, red points = new scenarios identified



Subramanyam, *et al.*, 2017

Optimal Finite Adaptivity

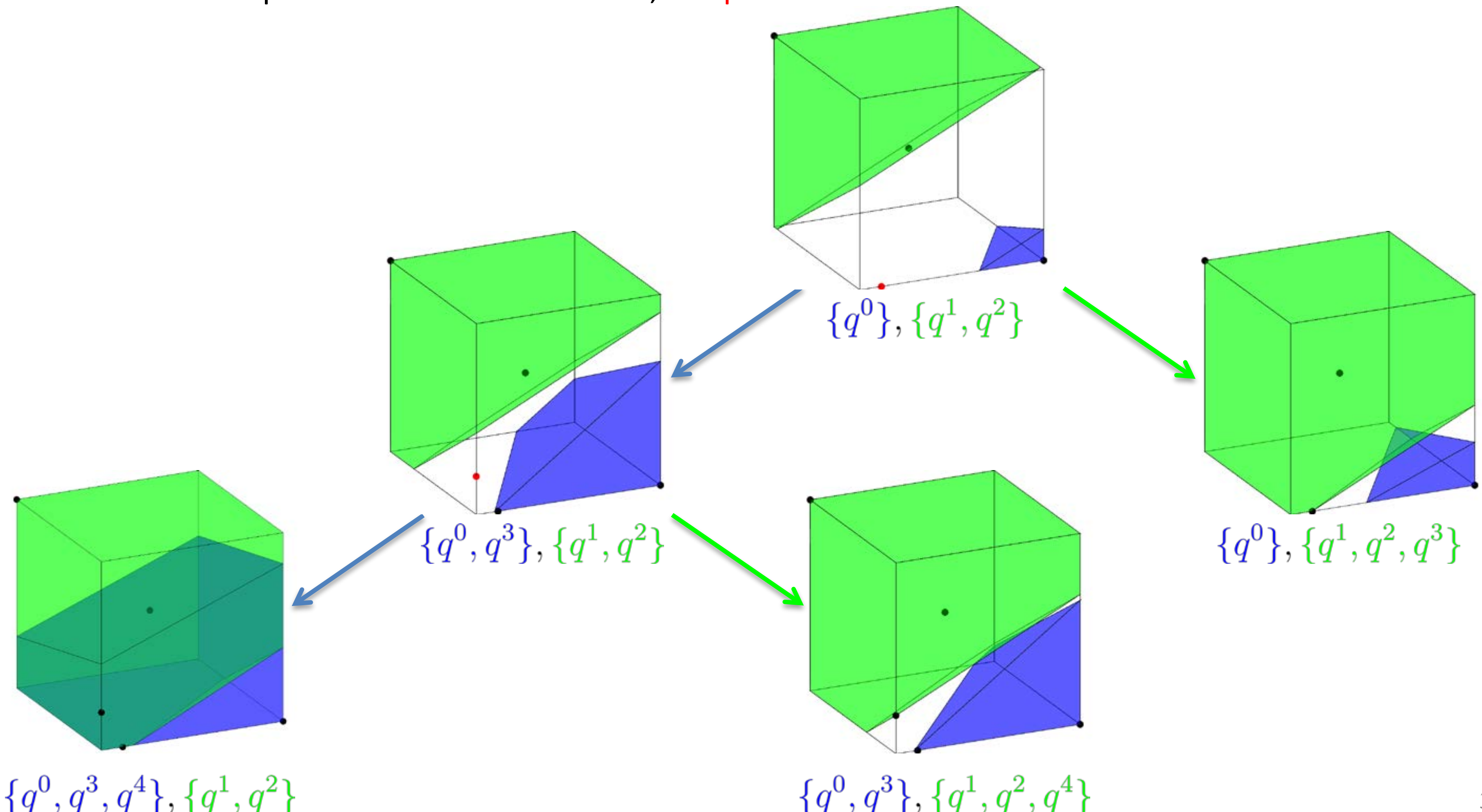
- Example with $K = 2$
  - Blue region = insured by policy 1, green region = insured by policy 2
  - Black points = scenarios enforced, red points = new scenarios identified



Subramanyam, *et al.*, 2017

Optimal Finite Adaptivity

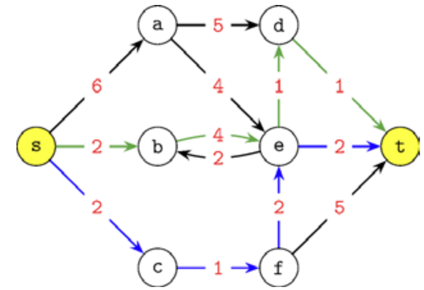
- Example with $K = 2$
  - Blue region = insured by policy 1, green region = insured by policy 2
  - Black points = scenarios enforced, red points = new scenarios identified

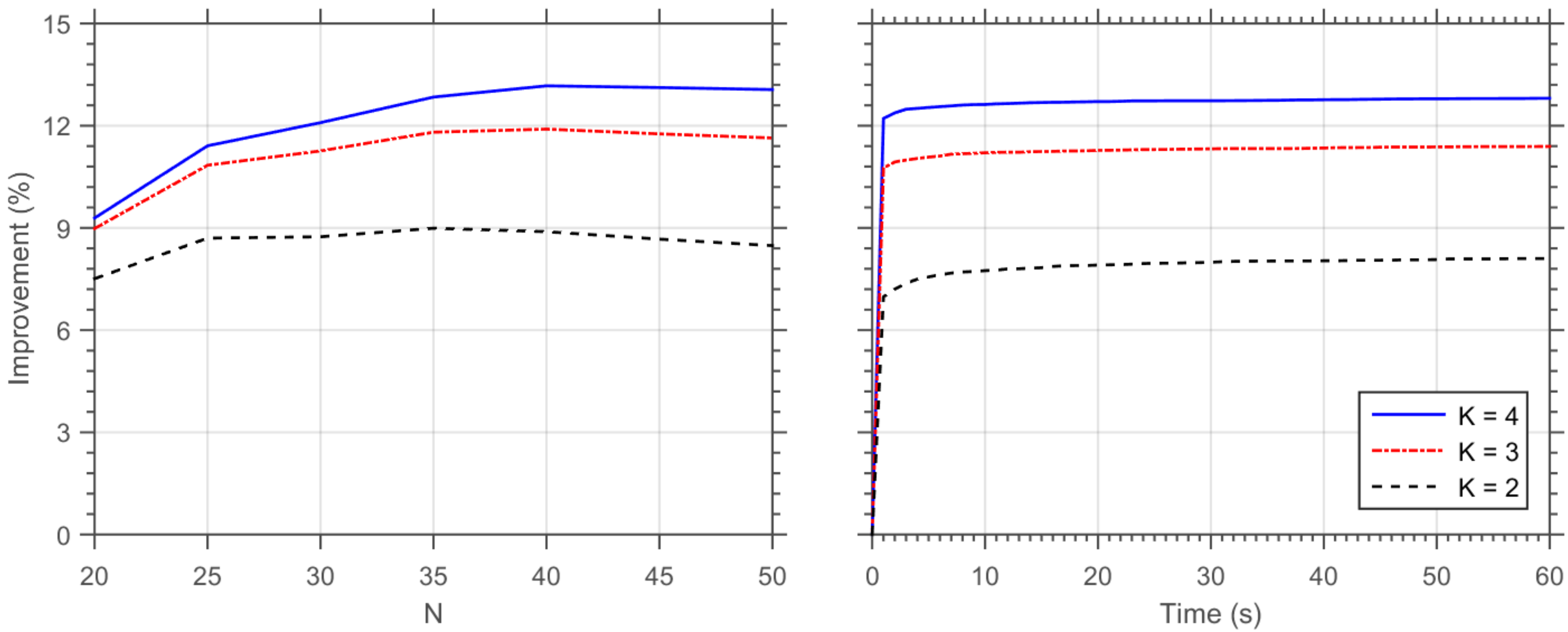


Subramanyam, *et al.*, 2017

Optimal Finite Adaptivity

- Shortest Paths with Uncertain Costs
  - Low K suffices for maximal WC-objective gains
  - High quality solutions identified quickly

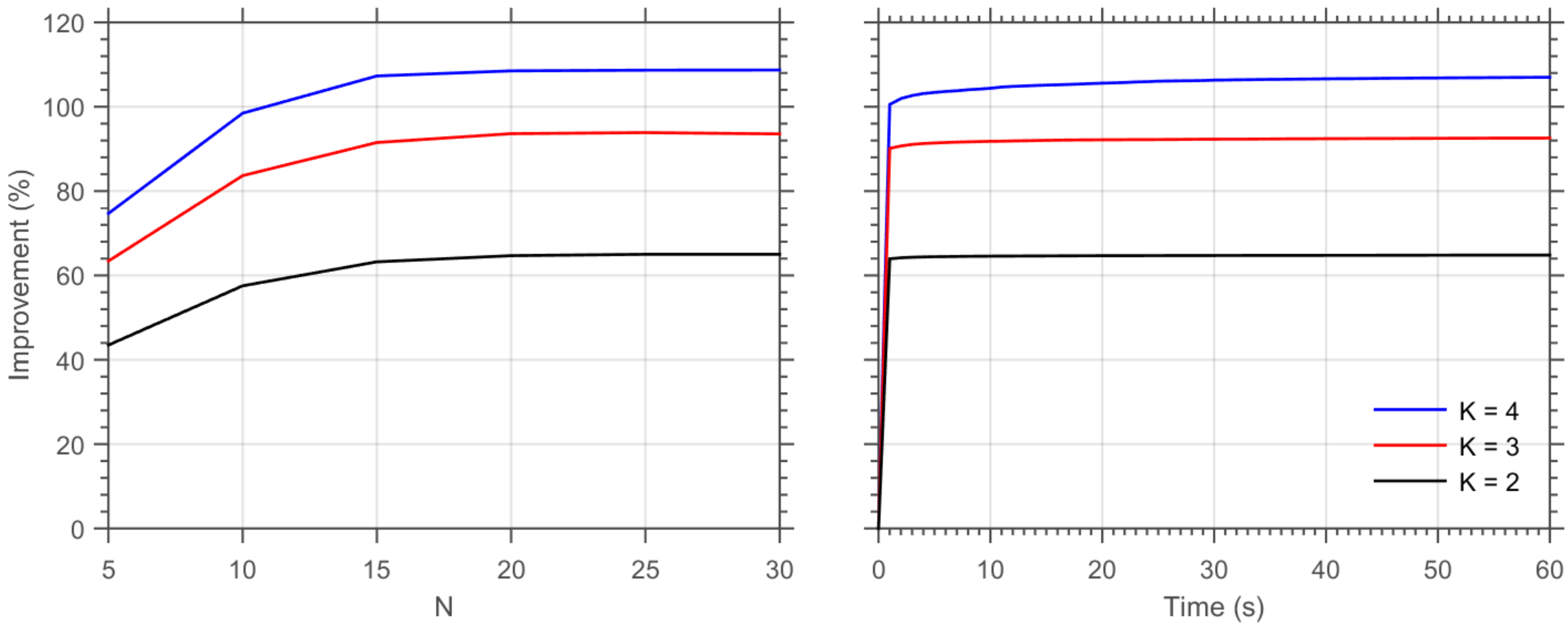




A. Subramanyam, C.E. Gounaris and W. Wiesemann (2017). K-Adaptability in Two-Stage Mixed-Integer Robust Optimization. Under Review. E-print available at: http://www.optimization-online.org/DB_HTML/2017/06/6093.html

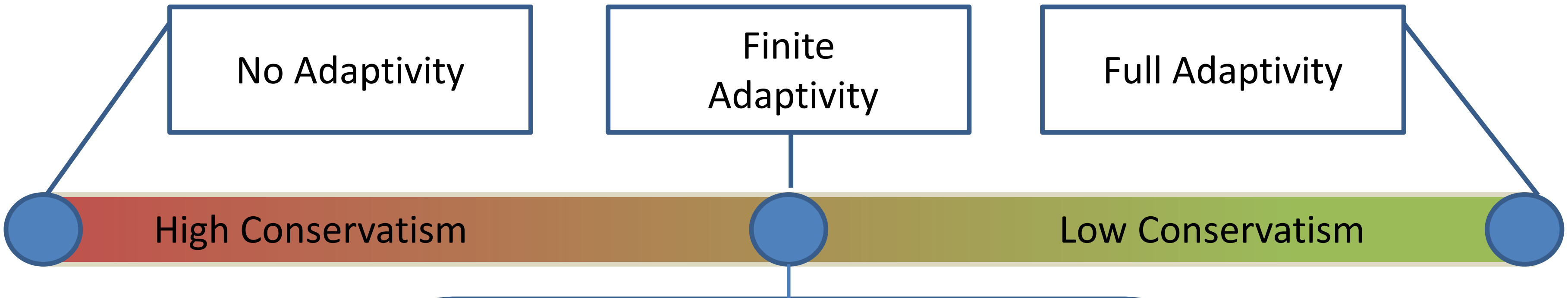
Optimal Finite Adaptivity

- Capital Budgeting with Uncertain ROIs
  - Low K suffices for maximal WC-objective gains
  - High quality solutions identified quickly



A. Subramanyam, C.E. Gounaris and W. Wiesemann (2017). K-Adaptability in Two-Stage Mixed-Integer Robust Optimization. Under Review. E-print available at: http://www.optimization-online.org/DB_HTML/2017/06/6093.html

2-Stage Robust Optimization



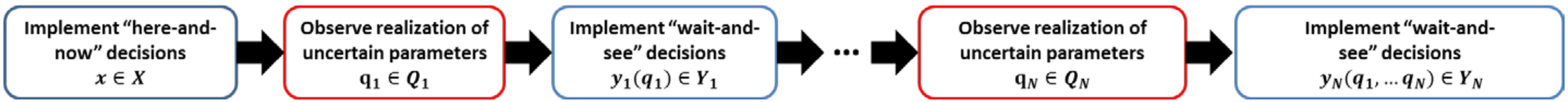
$$y(q) \leftarrow \psi(q)$$

- Postulate a specific functional form ("decision rule") and identify the best instantiation of these functions

- Many variants to choose from:
  - Affine $y(q) \leftarrow v + u^T q$ (*Ben-Tal et al., 2004*)
  - Piecewise-affine (*Bertsimas and Georghiou, 2015*)
  - Binary (*Bertsimas and Georghiou, 2014*)
  - Quadratic
  - Polynomial
  - …

- Affine decision rules are by far the most popular (tractable), and often the only viable option for multi-stage problems

Multi-stage Robust Optimization



$$\min_{x \in \mathcal{X}} \max_{q_1 \in \mathcal{Q}_1} \min_{y_1(q_1) \in \mathcal{Y}_1} \max_{q_2 \in \mathcal{Q}_2} \min_{y_2(q_1,q_2) \in \mathcal{Y}_2} \cdots \max_{q_N \in \mathcal{Q}_N} \min_{y_N(q_1,\ldots,q_N) \in \mathcal{Y}_N} f\left(x, y_1, \ldots, y_N, q_1, \ldots, q_N\right)$$

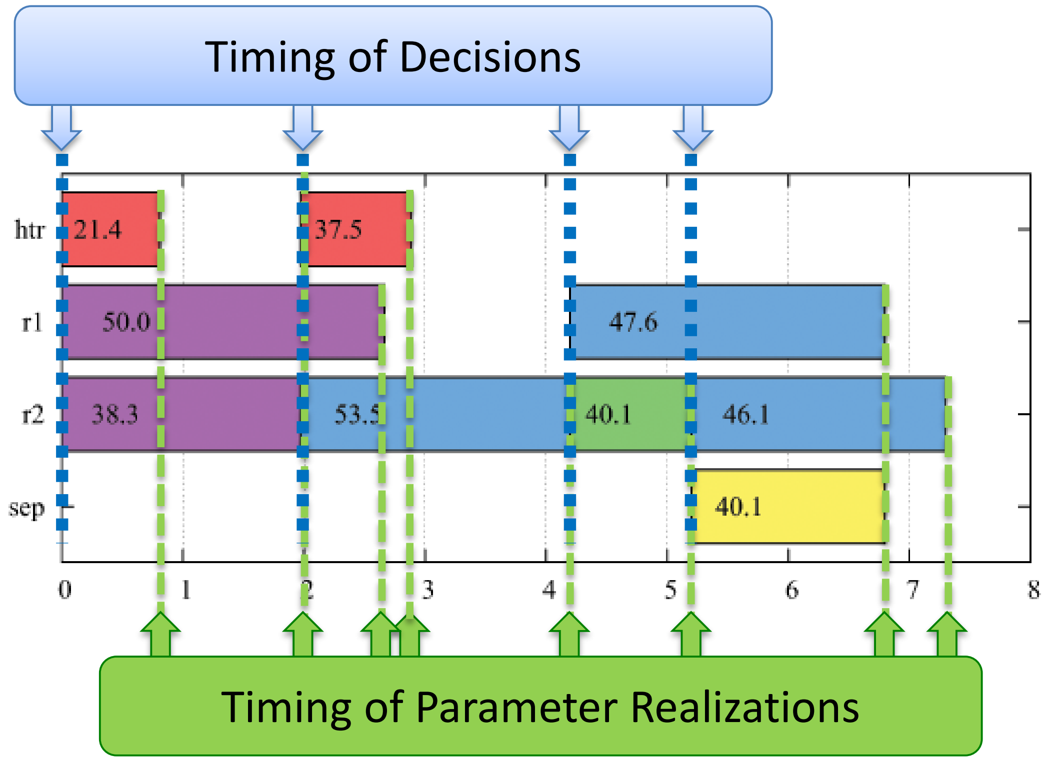
$$\text{s.t.} \quad g_i\left(x, y_1, \ldots, y_N, q_1, \ldots, q_N\right) \leq 0 \quad \forall \left(q_1, \ldots, q_N\right) \in \bigtimes_{j=1}^{N} \mathcal{Q}_j \quad \forall i$$

- Information gets revealed progressively
- Decisions have to be taken in between revelations
- Non-anticipativity must be obeyed
- Typical examples: **Scheduling**, **Inventory planning**, etc.

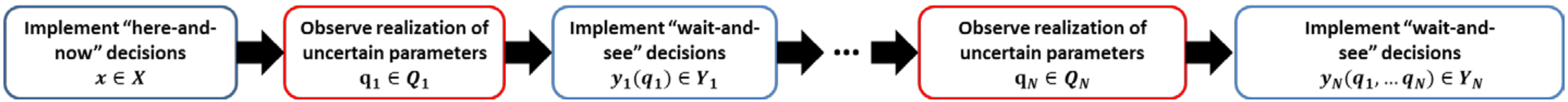
Multi-stage Robust Optimization

Implement "here-and-now" decisions $x \in X$ → Observe realization of uncertain parameters $q_1 \in Q_1$ → Implement "wait-and-see" decisions $y_1(q_1) \in Y_1$ → ... → Observe realization of uncertain parameters $q_N \in Q_N$ → Implement "wait-and-see" decisions $y_N(q_1, \dots q_N) \in Y_N$

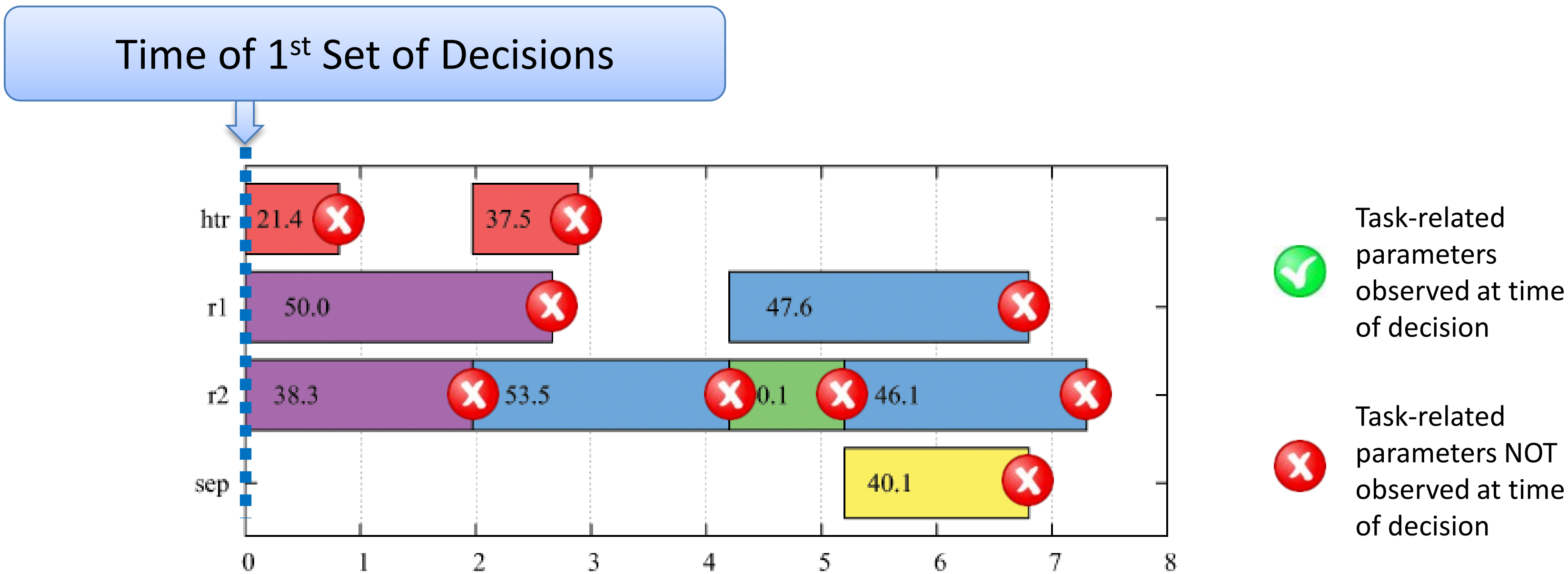
- Typical examples: **Scheduling, Inventory planning**, etc.



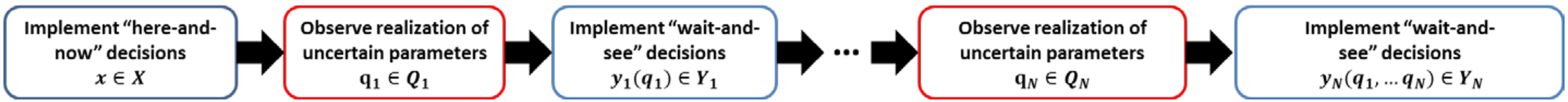
Multi-stage Robust Optimization



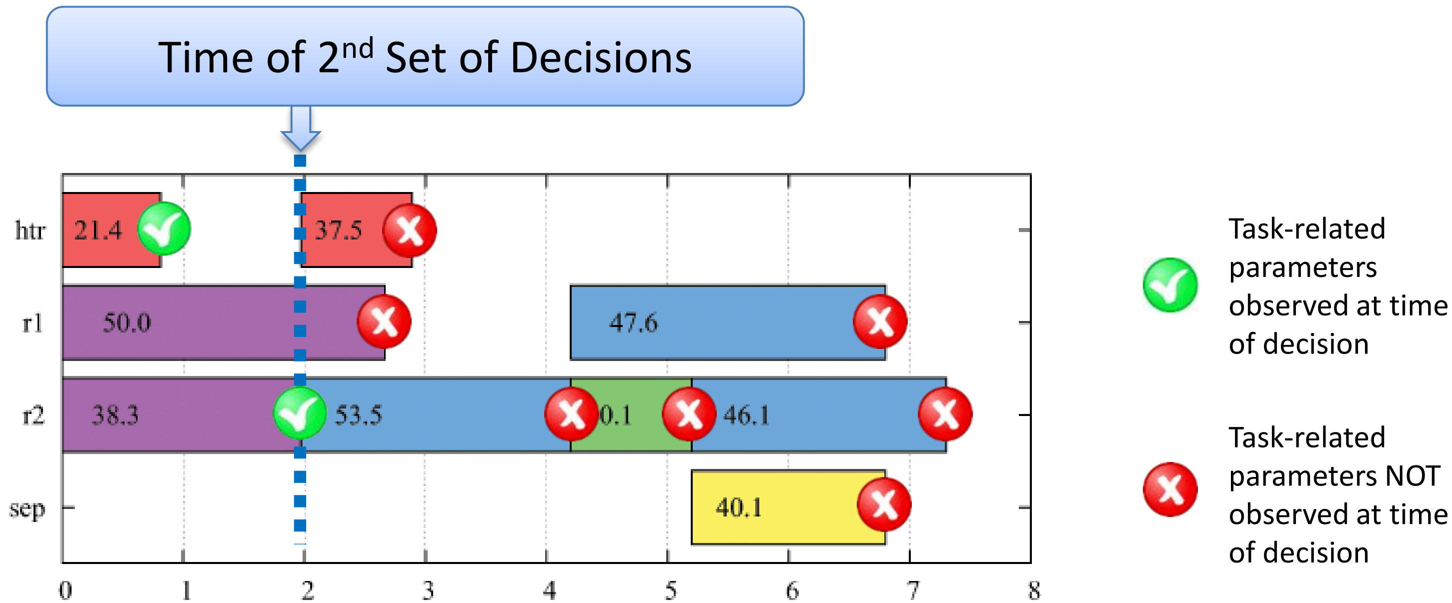
- Typical examples: **Scheduling, Inventory planning**, etc.



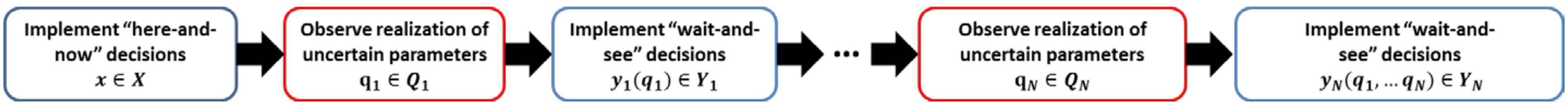
Multi-stage Robust Optimization



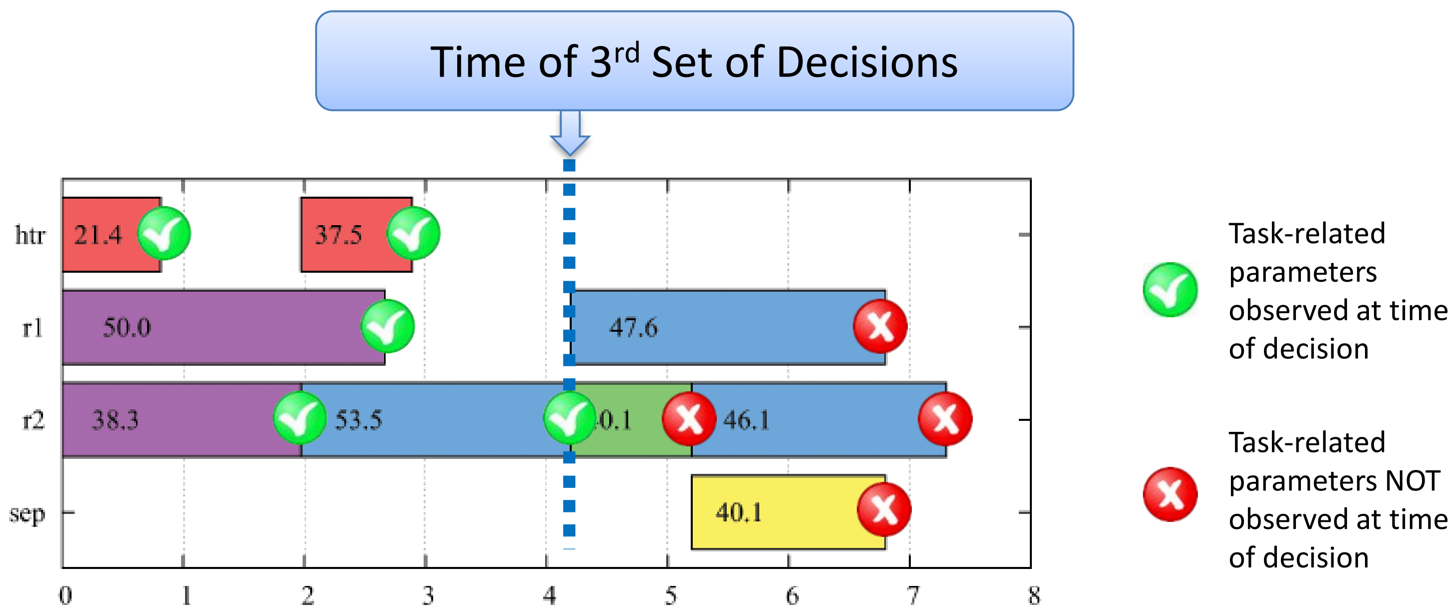
- Typical examples: **Scheduling, Inventory planning**, etc.



Multi-stage Robust Optimization



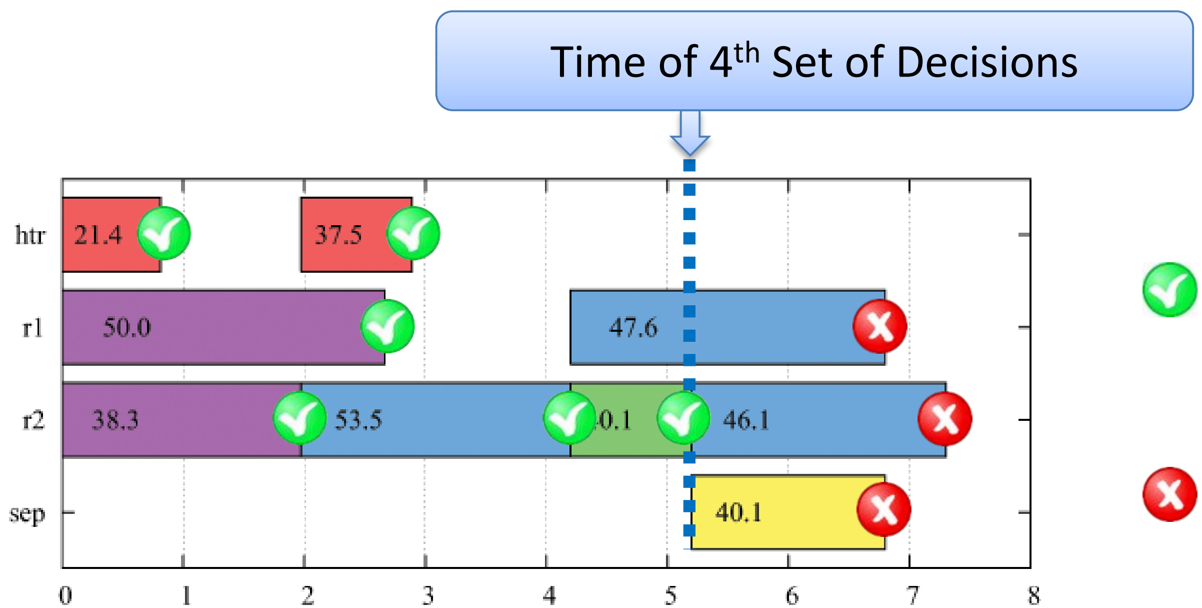
- Typical examples: **Scheduling, Inventory planning**, etc.



Multi-stage Robust Optimization

Implement "here-and-now" decisions $x \in X$ → Observe realization of uncertain parameters $q_1 \in Q_1$ → Implement "wait-and-see" decisions $y_1(q_1) \in Y_1$ → ... → Observe realization of uncertain parameters $q_N \in Q_N$ → Implement "wait-and-see" decisions $y_N(q_1, \ldots q_N) \in Y_N$

- Typical examples: **Scheduling, Inventory planning**, etc.



✓ Task-related parameters observed at time of decision

✗ Task-related parameters NOT observed at time of decision

Multi-stage Robust Optimization

Implement "here-and-now" decisions $x \in X$ → Observe realization of uncertain parameters $q_1 \in Q_1$ → Implement "wait-and-see" decisions $y_1(q_1) \in Y_1$ → … → Observe realization of uncertain parameters $q_N \in Q_N$ → Implement "wait-and-see" decisions $y_N(q_1, \dots q_N) \in Y_N$

- Affine Decision Rules

$$
\begin{aligned}
y_1 &\leftarrow v_1 + u_{11}^T q_1 \\
y_2 &\leftarrow v_2 + u_{21}^T q_1 + u_{22}^T q_2 \\
y_3 &\leftarrow v_3 + u_{31}^T q_1 + u_{32}^T q_2 + u_{33}^T q_3 \\
&\vdots \\
y_N &\leftarrow v_N + u_{N1}^T q_1 + u_{N2}^T q_2 + u_{N3}^T q_3 + \cdots + u_{NN}^T q_N
\end{aligned}
$$

Expect (lots of) degeneracy!

$$
\begin{aligned}
\min_{x \in \mathcal{X}} \max_{q \in \mathcal{Q}} \min_{y(q) \in \mathcal{Y}} \quad & f(x, y, q) \\
\text{s.t.} \quad & g_i(x, y, q) \le 0 \quad \forall q \in \mathcal{Q} \quad \forall i
\end{aligned}
$$

$y \leftarrow v + Uq$

Apply A.D.R.

$$
\begin{aligned}
\min_{\substack{x \in \mathcal{X}, \\ z \in \mathbb{R}, v \in \mathbb{R}, U \in \mathbb{R}}} \quad & z \\
\text{s.t.} \quad & z \ge f(x, v, U, q) \quad \forall q \in \mathcal{Q} \\
& g_i(x, v, U, q) \le 0 \quad \forall q \in \mathcal{Q} \quad \forall i \\
& v + Uq \in \mathcal{Y}
\end{aligned}
$$

Solve

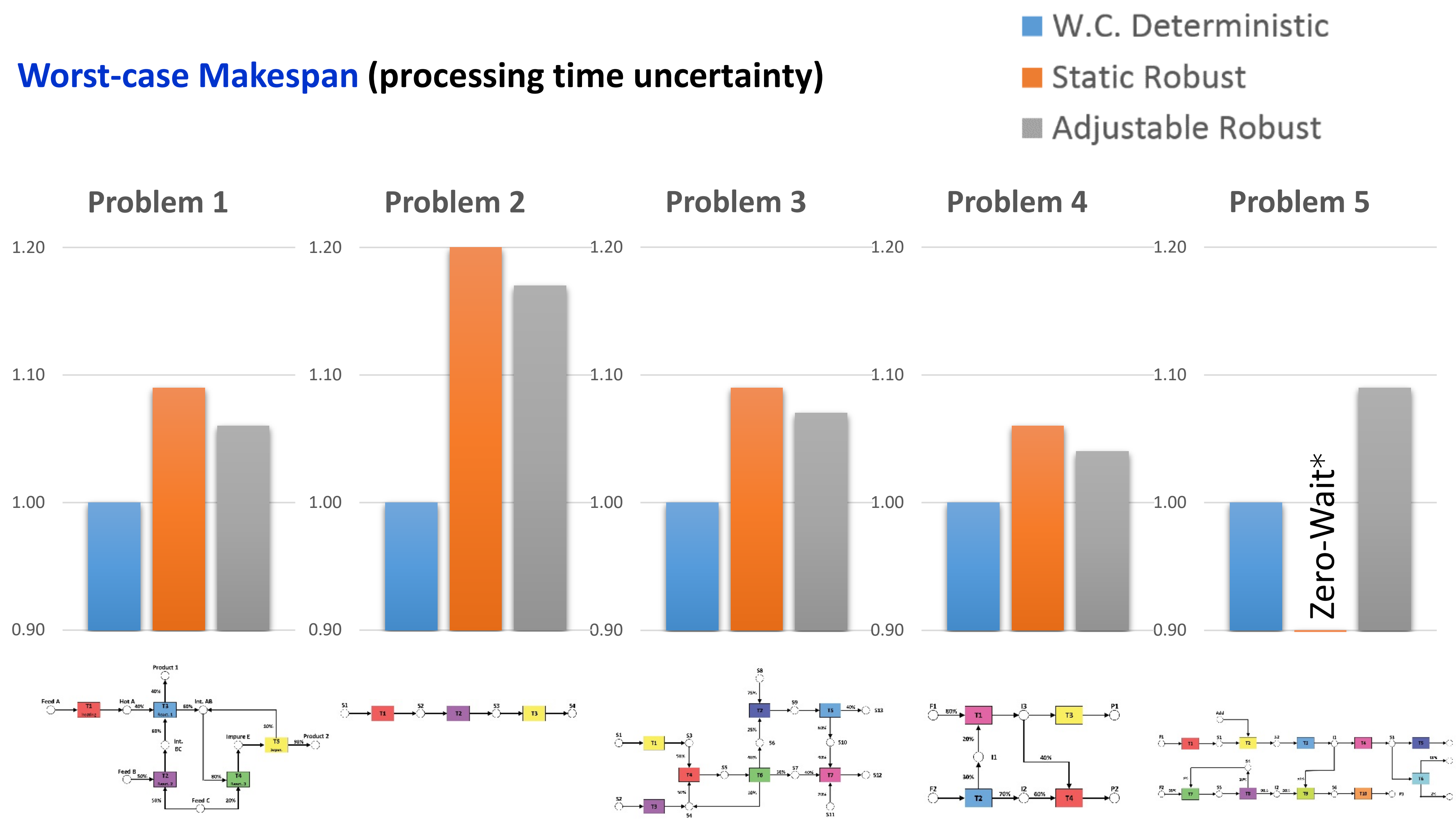
$$(x^\star, z^\star, v^\star, U^\star)$$

Polish solution

$$
\begin{aligned}
\min_{z \in \mathbb{R}, v \in \mathbb{R}, U \in \mathbb{R}} \quad & f(x^\star, v, U, q^0) \\
\text{s.t.} \quad & z^\star \ge f(x^\star, v, U, q) \quad \forall q \in \mathcal{Q} \\
& g_i(x^\star, v, U, q) \le 0 \quad \forall q \in \mathcal{Q} \quad \forall i \\
& v + Uq \in \mathcal{Y}
\end{aligned}
$$

Multi-stage RO Example: Process Scheduling

Worst-case Makespan (processing time uncertainty)

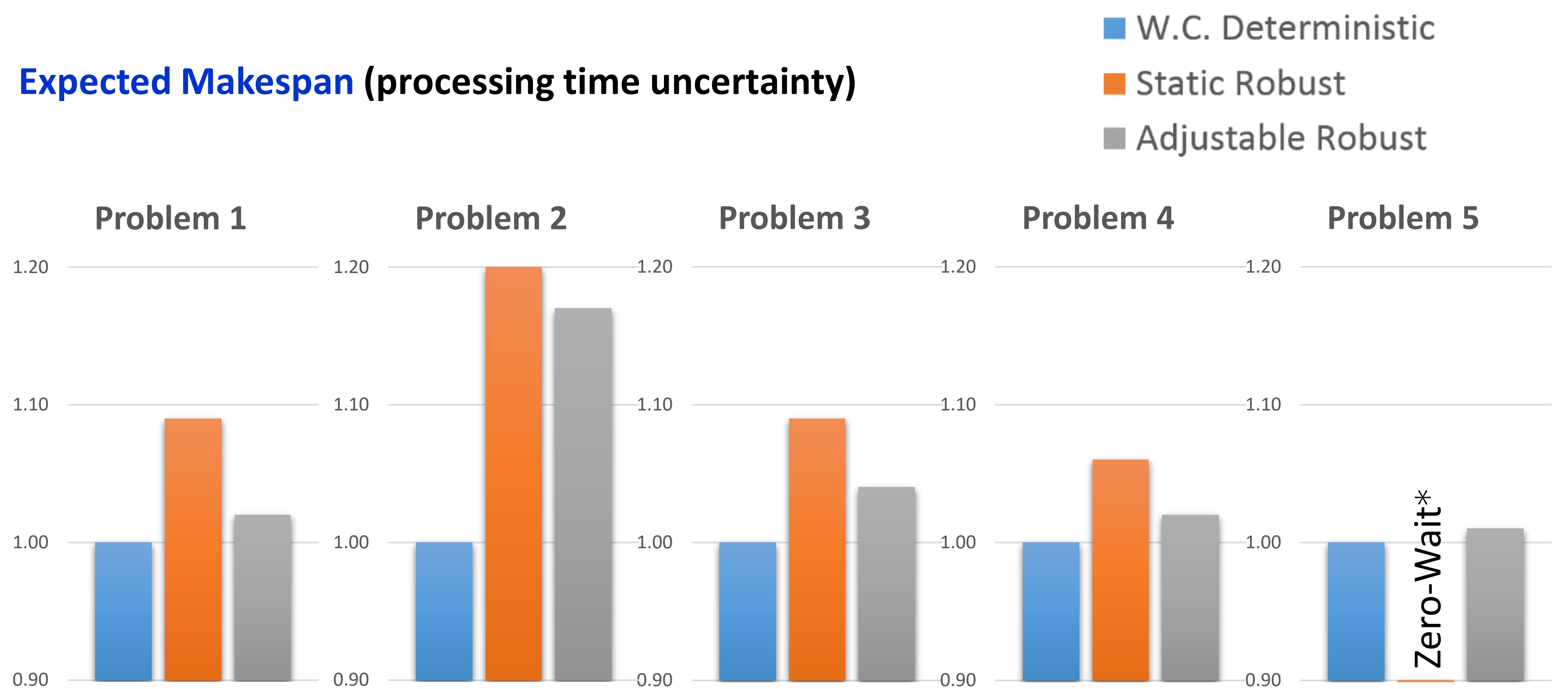


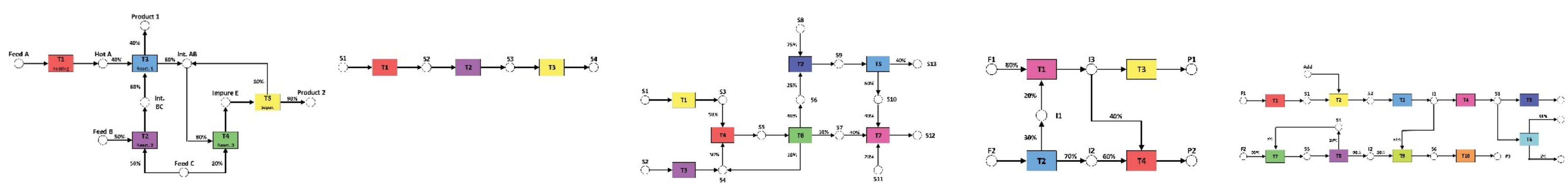
N.H. Lappas and C.E. Gounaris (2016). Multi-stage Adjustable Robust Optimization for Process Scheduling Under Uncertainty. AIChE Journal, 62(5):1646-1667. Selected as Editor's Choice Paper. DOI 10.1002/aic.15183

*This instance cannot be solved with the static robust approach

Multi-stage RO Example: Process Scheduling

Expected Makespan (processing time uncertainty)

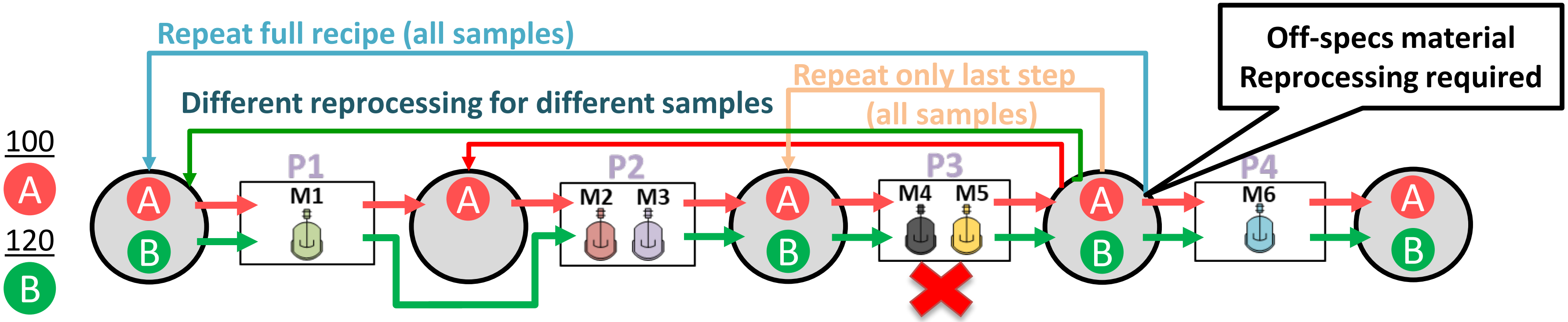




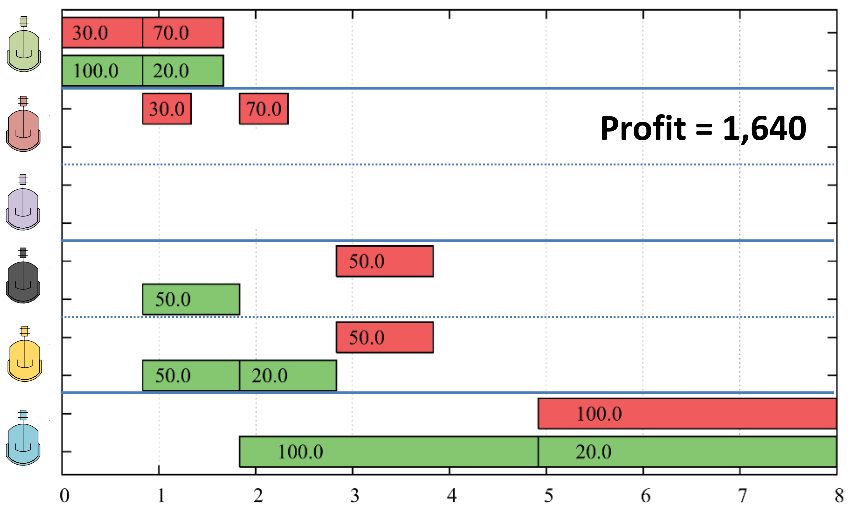
*This instance cannot be solved with the static robust approach

N.H. Lappas and C.E. Gounaris (2016). Multi-stage Adjustable Robust Optimization for Process Scheduling Under Uncertainty. AIChE Journal, 62(5):1646-1667. Selected as Editor's Choice Paper. DOI 10.1002/aic.15183

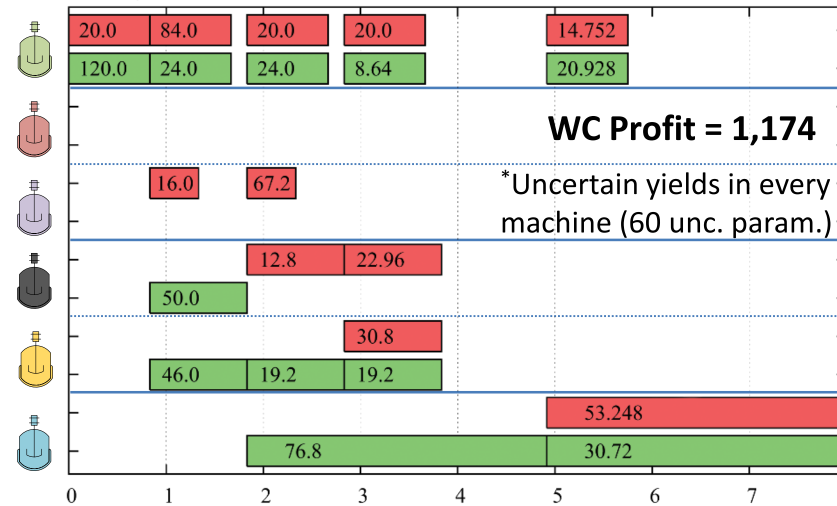
Multi-Tasking Scheduling with Reprocessing



Deterministic Optimal Schedule



Adjustable Robust Optimal Schedule*



N.H. Lappas, L.A. Ricardez-Sandoval, R. Fukasawa and C.E. Gounaris (2017). Adjustable Robust Optimization for Multi-tasking Scheduling with Reprocessing of Imperfect Tasks. Under Review

Endogenous Uncertainty

Uncertain parameters can be classified as:

Exogenous:

Parameter realizations do **not depend** on decisions

(e.g., weather)

Endogenous:

Decisions **can affect** the realizations of uncertain parameters

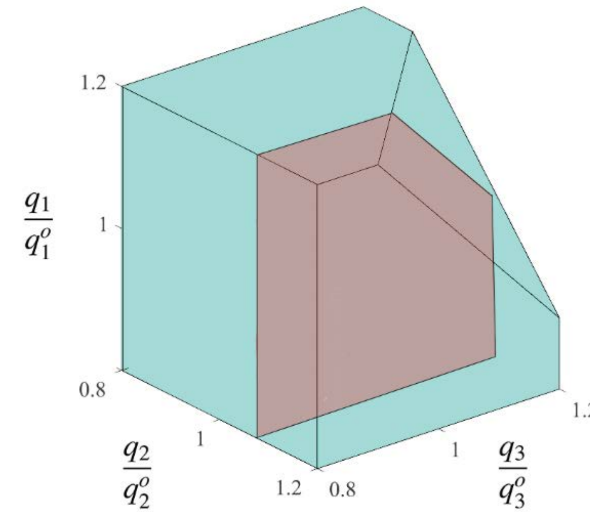
(e.g., maintenance decisions affect failure rates)

Flavors of endogeneity:

- Materialization: Decisions may make specific parameters lose their physical meaning (e.g., production yields of non-executed processes)
- Timing of realization: Decisions can affect the time stage at which parameters are observed (e.g., demand for a new product will be revealed after the period it is launched)
- Distributional support: Decisions can affect the underlying distributions from which a parameter realization draws (e.g., technology decisions can affect the production yields)

Decision-Dependent Uncertainty Sets

- Avoids unnecessarily conservative solutions that attempt to insure against risk we are not really exposed to
- Provides for a considerable degree of modeling flexibility so as to capture the endogenous nature of parameters



Example

$$w_1 q_1 + w_2 q_2 + w_3 q_3 \le 1.1\left(w_1 q_1^0 + w_2 q_2^0 + w_3 q_3^0\right)$$

If the binary decision $w_2 = 0$ prohibits parameter q_2 from materializing, then this parameter should not participate in the facet so as not to reduce the conservatism of the solution

$$\mathcal{Q} = \left\{ \begin{array}{c} q \in \mathbb{R}^{nq},\ p \in \{0,1\}^{np} : \\ Hq + Gp \le d \\ q^L \le q \le q^U \end{array} \right\}$$

$$\mathcal{Q}(w) = \left\{ \begin{array}{c} q \in \mathbb{R}^{nq},\ p \in \{0,1\}^{np} : \\ H\left(v^q(w) \circ q\right) + G\left(v^p(w) \circ p\right) \le Dw + d \\ q^L \le q \le q^U \end{array} \right\},$$

where: $v^q(w) \in \{0,1\}^{nq}$ and $v^p(w) \in \{0,1\}^{np}$

Modeling Capabilities

Modify distributional information

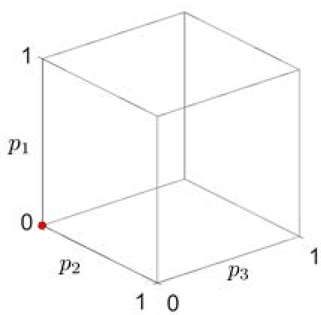
$$q \leq d_1 w_1 + d_2 w_2 + d_3 w_3$$

- No w selected
  $q \leq 0$
- w_1 selected
  $q \leq d_1$
- w_2 selected
  $q \leq d_2$
- w_3 selected
  $q \leq d_3$

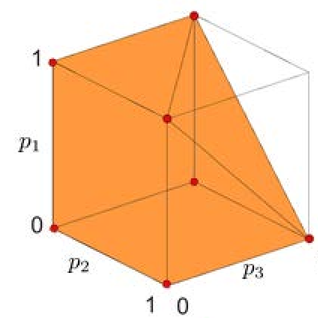
Control which scenarios are admitted by the set

$$p_1 + p_2 + p_3 \leq w_1 + w_2 + w_3$$

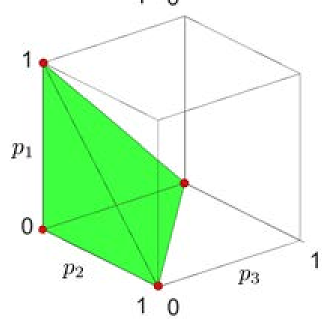
- No w selected
  $p_1 + p_2 + p_3 \leq 0$



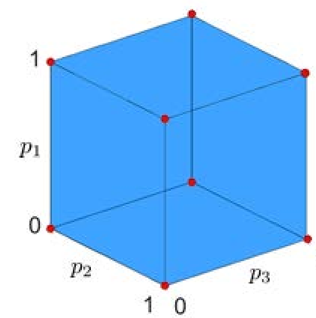
- Only two w selected
  $p_1 + p_2 + p_3 \leq 2$



- Only one w selected
  $p_1 + p_2 + p_3 \leq 1$



- All three w selected
  $p_1 + p_2 + p_3 \leq 3$



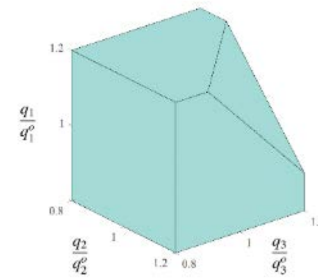
Modeling Capabilities

Remove the effect of non-materialized parameters

$$w_1 q_1 + w_2 q_2 + w_3 q_3 \leq (1 + 0.1)\left(w_1 q_1^0 + w_2 q_2^0 + w_3 q_3^0\right)$$

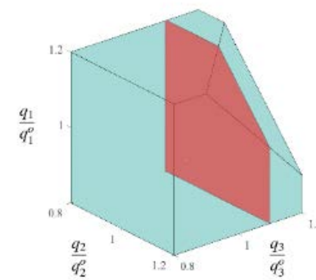
- All three w_1, w_2, w_3 have been selected:

$$1\ q_1 + 1\ q_2 + 1\ q_3 \leq (1 + 0.1)\left(1\ q_1^0 + 1\ q_2^0 + 1\ q_3^0\right)$$



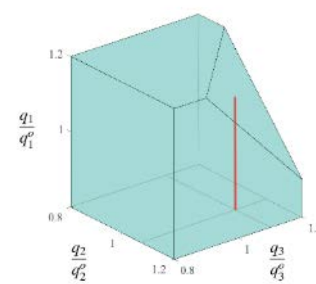
- Only w_1, w_2 have been selected:

$$1\ q_1 + 1\ q_2 + 0\ q_3 \leq (1 + 0.1)\left(1\ q_1^0 + 1\ q_2^0 + 0\ q_3^0\right)$$



- Only w_1 has been selected:

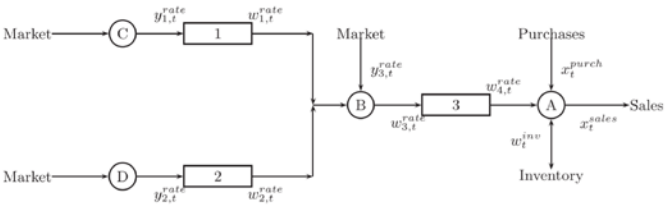
$$1\ q_1 + 0\ q_2 + 0\ q_3 \leq (1 + 0.1)\left(1\ q_1^0 + 0\ q_2^0 + 0\ q_3^0\right)$$



Case Studies

I. Capacity Expansion

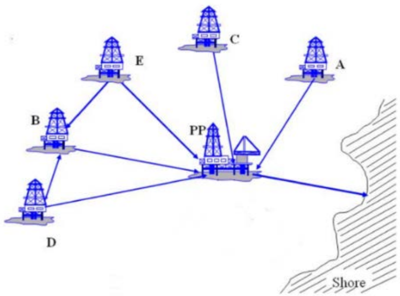
- *Production yield* of process i in time period t
  - Materialized, if process utilized in that period
- *Demand* for final product
  - Materialized always (exogenous)



(Goel and Grossmann, 2006)

II. Offshore Oil Production

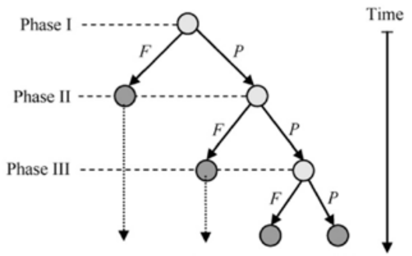
- *Initial deliverability* of field f
  - Materialized, if drilling occurs
  - Value depend on technology utilized
- *Reserve size* of field f
  - Materialized always (exogenous)



(Goel and Grossmann, 2004)

III. Clinical Trial Planning

- *Trial outcome* of drug i in phase j
  - Materialized, if trial begins in some time period t



(Colvin and Maravelias, 2008)

Benefits of Using DDUS

Normalized robust (max) objectives for 3 levels of uncertainty (low, medium, high)

Case Study	Deterministic	non-DDUS		
		L	M	H
I	100	71	65	44
II	100	69	63	41
III	100	57	45	36

N.H. Lappas and C.E. Gounaris (2017). Robust Optimization for Decision-making under Endogenous Uncertainty. Under Review. E-print available at: http://www.optimization-online.org/DB_HTML/2017/06/6105.html

Benefits of Using DDUS

Normalized robust (max) objectives for 3 levels of uncertainty (low, medium, high)

Case Study	Deterministic	DDUS			non-DDUS		
		L	M	H	L	M	H
I	100	90	84	79	71	65	44
II	100	87	79	60	69	63	41
III	100	71	65	54	57	45	36

N.H. Lappas and C.E. Gounaris (2017). Robust Optimization for Decision-making under Endogenous Uncertainty. Under Review. E-print available at: http://www.optimization-online.org/DB_HTML/2017/06/6105.html

Opportunities = Challenges

MODERN ROBUST OPTIMIZATION CAN (in principle) …

1. Solve problems using ”cutting-plane-like” approaches that progressively enforce the robustness of a solution
   - Capitalizes on deterministic optimization machinery
   - Provides more flexibility about what can be uncertain (e.g., disruptions)
   - Allows for better integration with custom-built solvers, including metaheuristics

2. Handle recourse (incl. mixed-integer recourse) in multi-stage decision making settings
   - Path to full adaptivity for 2-stage problems
   - Decision rules can be used (carefully) to address N-stage problems
   - Coping with equality constraints

3. Address endogenous uncertainty
   - Use of decision-dependent uncertainty sets

Group References

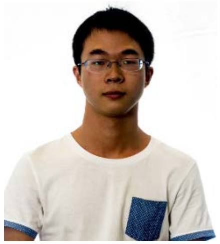
1. N.H. Lappas and C.E. Gounaris (2017). A Theoretical and Computational Comparison of Continuous-Time Process Scheduling Models for Adjustable Robust Optimization. Under Review
2. N.H. Lappas, L.A. Ricardez-Sandoval, R. Fukasawa and C.E. Gounaris (2017). Adjustable Robust Optimization for Multi-tasking Scheduling with Reprocessing of Imperfect Tasks. Under Review
3. C.E. Gounaris and D.R. Schmidt (2017). Generalized Hose Uncertainty in Single-Commodity Robust Network Design. Under Review
4. L.R. Matthews, C.E. Gounaris and Y.G. Kevrekidis (2017). Designing Networks with Uncertain Edge Failures Using-Two Stage Robust Optimization. Under Review
5. N.H. Lappas and C.E. Gounaris (2017). Robust Optimization for Decision-making under Endogenous Uncertainty. Under Review. E-print available at: http://www.optimization-online.org/DB_HTML/2017/06/6105.html
6. A. Subramanyam, C.E. Gounaris and W. Wiesemann (2017). K-Adaptability in Two-Stage Mixed-Integer Robust Optimization. Under Review. E-print available at: http://www.optimization-online.org/DB_HTML/2017/06/6093.html
7. A. Subramanyam, F. Mufalli, J. Pinto and C.E. Gounaris (2017). Robust Multi-Period Vehicle Routing Under Customer Order Uncertainty. Under Review. E-print available at: http://www.optimization-online.org/DB_HTML/2017/04/5947.html
8. A. Subramanyam, P.P. Repoussis and C.E. Gounaris (2017). Heuristics and Lower Bounds for Robust Heterogeneous Vehicle Routing Problems Under Demand Uncertainty. In: Proceedings of the First Triennial Conference of the INFORMS Transportation and Logistics Society, TSL-2017
9. C.E. Gounaris (2017). Advances in Robust Optimization and Opportunities for Process Operations. In: Proceedings of the Foundations of Computer-Aided Process Operations/Chemical Process Control (FOCAPO 2017/CPC IX), Paper ID IF110
10. A. Subramanyam and C.E. Gounaris (2017). Strategic Allocation of Time Windows in Vehicle Routing Problems under Uncertainty. In: Proceedings of the Foundations of Computer-Aided Process Operations/Chemical Process Control (FOCAPO 2017/CPC IX), Paper ID F62
11. C.E. Gounaris, P.P. Repoussis, C.D. Tarantilis, W. Wiesemann and C.A. Floudas (2016). An Adaptive Memory Programming Framework for the Robust Capacitated Vehicle Routing Problem. Transportation Science, 50(4):1239-1260. DOI 10.1287/trsc.2014.0559
12. N.H. Lappas and C.E. Gounaris (2016). Multi-stage Adjustable Robust Optimization for Process Scheduling Under Uncertainty. AIChE Journal, 62(5):1646-1667. Selected as Editor's Choice Paper. DOI 10.1002/aic.15183
13. N.H. Lappas and C.E. Gounaris (2016). Comparison of Continuous-Time Models for Adjustable Robust Optimization in Process Scheduling under Uncertainty. In: 26th European Symposium on Computer Aided Process Engineering, ESCAPE-26, Z. Kravanja and M. Bogataj (Eds.), Computer Aided Chemical Engineering, Vol. 38, pp. 391-396, Elsevier. ISBN: 978-0-444-63428-3. DOI 10.1016/S1570-7946(10)28142-7

Acknowledgements

Anirudh Subramanyam

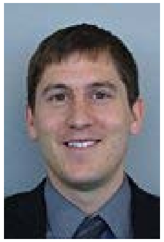
Nikolaos Lappas

Christopher Hanselman



Akang Wang

Natalie Isenberg



Logan Mathews

National Science Foundation
(CMMI – 1434682)
(CBET – 1510787)
(CMMI – 1634594)

Department of Energy, Institute for the Design of Advanced Energy Systems
(DE-AC02-05CH11231)

Dow
John Wassick et al.

Imperial College London
Wolfram Wiesemann

Lilly
Sal Garcia

National Energy Technology Laboratory
David Miller
Chris Matranga et al.

Praxair
Jose Pinto et al.

Stevens
Panagiotis Repoussis

Universität zu Köln
Daniel Schmidt

University of Pittsburgh
Ioannis Mpourmpakis
Goetz Veser

University of Waterloo
Luis Ricardez-Sandoval
Ricardo Fukasawa