Stochastic and Robust Optimization in Logistics

Alan Erera[1]

[1]The Supply Chain and Logistics Institute and
School of Industrial and Systems Engineering, Georgia Tech

alerera@isye.gatech.edu

Introductions

About Me

- At Georgia Tech for 9 years
- Research interests in dynamic and stochastic logistics optimization; routing and scheduling; logistics system resiliency
- www.isye.gatech.edu/∼alerera

Introductions

About Me

- At Georgia Tech for 9 years
- Research interests in dynamic and stochastic logistics optimization; routing and scheduling; logistics system resiliency
- www.isye.gatech.edu/∼alerera

If you don't understand

Please interrupt me and ask questions...

What to remember

1. Stochastic and robust optimization are for *dynamic* decision problems

What to remember

1. Stochastic and robust optimization are for *dynamic* decision problems
2. Many ways to effectively incorporate parameter uncertainty in logistics optimization

What to remember

1. Stochastic and robust optimization are for *dynamic* decision problems
2. Many ways to effectively incorporate parameter uncertainty in logistics optimization
3. Modeling and treatment of *recourse* especially critical

What to remember

1. Stochastic and robust optimization are for *dynamic* decision problems
2. Many ways to effectively incorporate parameter uncertainty in logistics optimization
3. Modeling and treatment of *recourse* especially critical
4. Ensure that your model is useful (and interesting), then solve

Scope

What I will cover:

- Stochastic integer programming
- Chance constraints and integer programming
- Robust (worst-case) constraints and integer programming
- Primarily modeling, and solution heuristics

Scope

What I will cover:

- Stochastic integer programming
- Chance constraints and integer programming
- Robust (worst-case) constraints and integer programming
- Primarily modeling, and solution heuristics

What I will not cover:

- Dynamic programming (MDP)
- Approximate dynamic programming

Dynamic planning

Dynamic planning problems

A *dynamic planning problem* is one where planning decisions are made sequentially in time

Dynamic planning

Dynamic planning problems

A *dynamic planning problem* is one where planning decisions are made sequentially in time

Motivating Examples

- Each month, a load plan (freight routing plan) is determined for the following month

Dynamic planning

Dynamic planning problems

A *dynamic planning problem* is one where planning decisions are made sequentially in time

Motivating Examples

- Each month, a load plan (freight routing plan) is determined for the following month
- Each week, an empty trailer repositioning plan is determined for the following week

Dynamic planning

Dynamic planning problems

A *dynamic planning problem* is one where planning decisions are made sequentially in time

Motivating Examples

- Each month, a load plan (freight routing plan) is determined for the following month
- Each week, an empty trailer repositioning plan is determined for the following week
- Each evening, distribution vehicle routes are planned for tomorrow

Dynamic planning

Dynamic planning problems

A *dynamic planning problem* is one where planning decisions are made sequentially in time

Motivating Examples

- Each month, a load plan (freight routing plan) is determined for the following month
- Each week, an empty trailer repositioning plan is determined for the following week
- Each evening, distribution vehicle routes are planned for tomorrow
- Each time a new customer request arrives, it is added to a vehicle route

Are most logistics planning problems dynamic?

Are most logistics planning problems dynamic?

Yes!

Are most logistics planning problems dynamic?

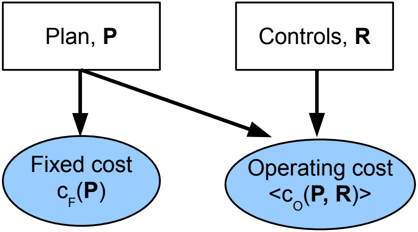
Yes!

- Many, if not most, quantitative decision problems in logistics are inherently dynamic, by my definition.
- Our focus: **how to build and solve an appropriate optimization model for each such problem?**

Planning and control

Control decisions

Between planning periods, *controls* are used to implement a plan feasibly and effectively



Simple rules, or may result from (**recourse**) optimization problems

Deterministic and Stochastic Planning Models

Two classes of optimization models for dynamic logistics planning:

Deterministic and Stochastic Planning Models

Two classes of optimization models for dynamic logistics planning:

Deterministic Model

A model in which all parameters are *assumed* to be **known** when planning

Deterministic and Stochastic Planning Models

Two classes of optimization models for dynamic logistics planning:

Deterministic Model

A model in which all parameters are *assumed* to be **known** when planning

Stochastic Model

A model in which one or more parameters are *assumed* to be **uncertain** when planning

When Should Uncertainty Be Ignored?

When Should Uncertainty Be Ignored?

Parameter Availability

Are all model parameters known when planning?

When Should Uncertainty Be Ignored?

Parameter Availability

Are all model parameters known when planning?

Parameter Variability

If uncertain model parameters are replaced with nominal (expected) values when planning, does the model produce good results?

When Should Uncertainty Be Ignored?

Parameter Availability

Are all model parameters known when planning?

Parameter Variability

If uncertain model parameters are replaced with nominal (expected) values when planning, does the model produce good results?

The latter is another *engineering* decision

How Should Uncertainty Be Incorporated?

1. Probabilistic programming models
2. Two-stage models
3. Multiple-stage models

How Should Uncertainty Be Incorporated?

1. Probabilistic programming models
2. Two-stage models
3. Multiple-stage models

How Should Uncertainty Be Incorporated?

1. Probabilistic programming models
   - Chance-constrained programming and heuristics
2. Two-stage models
3. Multiple-stage models

How Should Uncertainty Be Incorporated?

1. Probabilistic programming models
   - Chance-constrained programming and heuristics
2. Two-stage models
   - Two-stage stochastic integer programming and heuristics
3. Multiple-stage models

How Should Uncertainty Be Incorporated?

1. Probabilistic programming models
   - Chance-constrained programming and heuristics
2. Two-stage models
   - Two-stage stochastic integer programming and heuristics
   - Two-stage SIP with robust or chance constraints
3. Multiple-stage models

How Should Uncertainty Be Incorporated?

1. Probabilistic programming models
   - Chance-constrained programming and heuristics
2. Two-stage models
   - Two-stage stochastic integer programming and heuristics
   - Two-stage SIP with robust or chance constraints
   - Two-stage robust optimization
3. Multiple-stage models

How Should Uncertainty Be Incorporated?

1. Probabilistic programming models
   - Chance-constrained programming and heuristics
2. Two-stage models
   - Two-stage stochastic integer programming and heuristics
   - Two-stage SIP with robust or chance constraints
   - Two-stage robust optimization
3. Multiple-stage models
   - Multi-stage SIP

How Should Uncertainty Be Incorporated?

1. Probabilistic programming models
   - Chance-constrained programming and heuristics
2. Two-stage models
   - Two-stage stochastic integer programming and heuristics
   - Two-stage SIP with robust or chance constraints
   - Two-stage robust optimization
3. Multiple-stage models
   - Multi-stage SIP
   - Dynamic programming and approximate dynamic programming

How Should Uncertainty Be Incorporated?

Two-Stage Models

Imagine an environment with two decision stages:

1. *First (Planning) Stage*: Planning decisions are made, some parameters uncertain
2. *Second (Control, Recourse) Stage*: Control decisions are made, **all** uncertain parameters revealed (known)

How Should Uncertainty Be Incorporated?

Two-Stage Models

Imagine an environment with two decision stages:

1. *First (Planning) Stage*: Planning decisions are made, some parameters uncertain
2. *Second (Control, Recourse) Stage*: Control decisions are made, **all** uncertain parameters revealed (known)

- Many real-world problems can be modeled with precision in this way

How Should Uncertainty Be Incorporated?

Two-Stage Models

Imagine an environment with two decision stages:

1. *First (Planning) Stage*: Planning decisions are made, some parameters uncertain
2. *Second (Control, Recourse) Stage*: Control decisions are made, **all** uncertain parameters revealed (known)

- Many real-world problems can be modeled with precision in this way
- **Even for those that cannot**, this is still frequently a reasonable first approximation for including uncertainty during planning

Two-stage Recourse Models

Explicit modeling of control decisions:

1. *First (Planning) Stage*: Planning decisions are made, some parameters uncertain
2. *Second (Control, Recourse) Stage*: Control decisions are made, **all** uncertain parameters revealed (known)

Two-stage Recourse Models

Explicit modeling of control decisions:

1. *First (Planning) Stage*: Planning decisions are made, some parameters uncertain
2. *Second (Control, Recourse) Stage*: Control decisions are made, **all** uncertain parameters revealed (known)

A Key Modeling Issue

How are second stage (recourse) decisions to be modeled?

Two-stage Recourse Models

Explicit modeling of control decisions:

1. *First (Planning) Stage*: Planning decisions are made, some parameters uncertain
2. *Second (Control, Recourse) Stage*: Control decisions are made, **all** uncertain parameters revealed (known)

A Key Modeling Issue

How are second stage (recourse) decisions to be modeled?

1. Fixed operating rules, or

Two-stage Recourse Models

Explicit modeling of control decisions:

1. *First (Planning) Stage*: Planning decisions are made, some parameters uncertain
2. *Second (Control, Recourse) Stage*: Control decisions are made, **all** uncertain parameters revealed (known)

A Key Modeling Issue

How are second stage (recourse) decisions to be modeled?

1. Fixed operating rules, or
2. Optimization problem for control

How Should Uncertainty Be Incorporated?

Multi-Stage Models

In many dynamic planning settings, uncertainty is revealed in multiple stages over time

1. *First Stage*: Given $(0, \mathcal{I}_0)$, decisions x_1 are determined
2. *nth Stage*: Given $(x_{n-1}, \mathcal{I}_{n-1})$, decisions x_n are determined

How Should Uncertainty Be Incorporated?

Multi-Stage Models

In many dynamic planning settings, uncertainty is revealed in multiple stages over time

1. *First Stage*: Given $(0, \mathcal{I}_0)$, decisions x_1 are determined
2. *nth Stage*: Given $(x_{n-1}, \mathcal{I}_{n-1})$, decisions x_n are determined

- Information pattern $\mathcal{I}_k$ summarizes known and uncertain information available for stage $k+1$ decision-making

How Should Uncertainty Be Incorporated?

Probabilistic Programming

- Planning decisions are made using models that use probabilistic forms in the constraints or objective function
- Control or recourse decisions are not modeled explicitly

How Should Uncertainty Be Incorporated?

Probabilistic Programming

- Planning decisions are made using models that use probabilistic forms in the constraints or objective function
- Control or recourse decisions are not modeled explicitly

- Not typically used directly today, but

How Should Uncertainty Be Incorporated?

Probabilistic Programming

- Planning decisions are made using models that use probabilistic forms in the constraints or objective function
- Control or recourse decisions are not modeled explicitly

- Not typically used directly today, but
- Ideas like *chance constraints* or *robust constraints* can be useful, and can be incorporated if necessary within explicit two-stage models

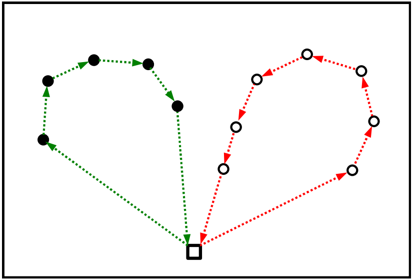
Remainder of Presentation

- Illustration of the ideas via examples
- References for more detailed information

VRP with Stochastic Demands (VRPSD)

Capacitated Vehicle Routing Problem

Given depot-based fleet of vehicles of capacity Q, travel cost matrix $\{c_{ij}\}$, and known customer demands $\{q_i\}$, find set of depot-based capacity feasible vehicle tours with minimum total travel cost



VRP with Stochastic Demands (VRPSD)

Parameter Availability

- Customer demand values not known with certainty when planning

VRP with Stochastic Demands (VRPSD)

Parameter Availability

- Customer demand values not known with certainty when planning
- Different models use different assumptions about when they are known

VRP with Stochastic Demands (VRPSD)

Parameter Availability

- Customer demand values not known with certainty when planning
- Different models use different assumptions about when they are known
  1. Before vehicle departure from depot each day

VRP with Stochastic Demands (VRPSD)

Parameter Availability

- Customer demand values not known with certainty when planning
- Different models use different assumptions about when they are known
  1. Before vehicle departure from depot each day
  2. Upon arrival at customer location

VRP with Stochastic Demands (VRPSD)

Vehicle Routing Problem with Stochastic Demands

Given depot-based fleet of vehicles of capacity Q, travel cost matrix $\{c_{ij}\}$, and **uncertain** customer demands $\{\tilde{q}_i\}$ independent with known distributions, find set of depot-based vehicle tours that (*)

VRP with Stochastic Demands (VRPSD)

Vehicle Routing Problem with Stochastic Demands

Given depot-based fleet of vehicles of capacity Q, travel cost matrix $\{c_{ij}\}$, and **uncertain** customer demands $\{\tilde{q}_i\}$ independent with known distributions, find set of depot-based vehicle tours that (*)

- Probabilistic programming version
  - (*) Minimize total travel cost subject to **chance constraints** on the capacity feasibility of each vehicle tour

VRP with Stochastic Demands (VRPSD)

Vehicle Routing Problem with Stochastic Demands

Given depot-based fleet of vehicles of capacity Q, travel cost matrix $\{c_{ij}\}$, and **uncertain** customer demands $\{\tilde{q}_i\}$ independent with known distributions, find set of depot-based vehicle tours that (*)

- Probabilistic programming version
  - (*) Minimize total travel cost subject to **chance constraints** on the capacity feasibility of each vehicle tour
- Tours must be planned before uncertainty revealed

Chance-constrained VRPSD Model

Stewart and Golden (1982)

(m-vehicle VRP)

$$\min \sum_k \sum_{i,j} c_{ij} x_{ijk}$$

$$\sum_{i,j} q_i x_{ijk} \leq Q \quad \forall\ k$$

$$\{x_{ijk}\} \in S_m$$

where S_m is set of all m-traveling salesperson solutions

Chance-constrained VRPSD Model

Stewart and Golden (1982)

(m-vehicle chance-constrained VRPSD)

$$\min \sum_{k} \sum_{i,j} c_{ij} x_{ijk}$$

$$P\left(\sum_{i,j} \tilde{q}_i x_{ijk} \leq Q\right) \geq 1 - \alpha \quad \forall\, k$$

$$\{x_{ijk}\} \in S_m$$

Chance-constrained VRPSD Model

Stewart and Golden (1982)

(m-vehicle chance-constrained VRPSD)

$$\min \sum_k \sum_{i,j} c_{ij} x_{ijk}$$

$$P\left(\sum_{i,j} \tilde{q}_i x_{ijk} \leq Q\right) \geq 1 - \alpha \quad \forall\, k$$

$$\{x_{ijk}\} \in S_m$$

- α is a *tour failure* probability

Representing chance constraints

Deterministic equivalents

Can we find a equivalent deterministic representation of the set of all solutions satisfying chance constraints:

$$K = \cap_i K^i$$

where

$$K^i = \{x | P(A^i(\omega)x \geq h^i(\omega)) \geq \rho^i\}$$

Representing chance constraints

Deterministic equivalents

Can we find a equivalent deterministic representation of the set of all solutions satisfying chance constraints:

$$K = \cap_i K^i$$

where

$$K^i = \{x | P(A^i(\omega)x \geq h^i(\omega)) \geq \rho^i\}$$

- Straightforward and linear when A^i fixed, but

Representing chance constraints

Deterministic equivalents

Can we find a equivalent deterministic representation of the set of all solutions satisfying chance constraints:

$$K = \cap_i K^i$$

where

$$K^i = \{x | P(A^i(\omega)x \geq h^i(\omega)) \geq \rho^i\}$$

- Straightforward and linear when A^i fixed, but
- More difficult when $A^i(\omega)$ varies (even if $h^i(\omega)$ fixed)

Deterministic equivalent for capacity chance constraint

$$P\left(\sum_{i,j} \tilde{q}_i x_{ijk} \leq Q\right) \geq 1 - \alpha \quad \forall\, k$$

Deterministic equivalent

$$M_k + \tau S_k \leq Q \quad \forall\, k$$

where $M_k = \sum_{i,j} \mu_i x_{ijk}$ and $S_k = \sqrt{\sum_{i,j} \sigma_i^2 x_{ijk}}$, and

$$P\left(\frac{\sum_{i,j} \tilde{q}_i x_{ijk} - M_k}{S_k} \leq \tau\right) = 1 - \alpha$$

Deterministic equivalent for capacity chance constraint

Deterministic equivalent

$$M_k + \tau S_k \leq Q \quad \forall\, k$$

where $M_k = \sum_{i,j} \mu_i x_{ijk}$ and $S_k = \sqrt{\sum_{i,j} \sigma_i^2 x_{ijk}}$, and

$$P\left(\frac{\sum_{i,j} \widetilde{q}_i x_{ijk} - M_k}{S_k} \leq \tau\right) = 1 - \alpha$$

Approach works when $\widetilde{q}_i$ are independent, then there may exist a τ that satisfies the expression (true for normal, Poisson, binomial random variables)

Normal distribution example

- Suppose $\{\widetilde{q}_1, \widetilde{q}_2, \ldots, \widetilde{q}_n\}$ are independent and normally distributed
    - Means μ_i, variances σ_i^2

Normal distribution example

- Suppose $\{\tilde{q}_1, \tilde{q}_2, \ldots, \tilde{q}_n\}$ are independent and normally distributed
  - Means μ_i, variances σ_i^2
- Then $\tilde{q}(S) = \sum_{i \in S} \tilde{q}_i$ remains normal
  - Mean $\mu(S) = \sum_{i \in S} \mu_i$, variance $\sigma^2(S) = \sum_{i \in S} \sigma_i^2$

Normal distribution example

- Suppose $\{\widetilde{q}_1, \widetilde{q}_2, \dots, \widetilde{q}_n\}$ are independent and normally distributed
  - Means μ_i, variances σ_i^2
- Then $\widetilde{q}(S) = \sum_{i \in S} \widetilde{q}_i$ remains normal
  - Mean $\mu(S) = \sum_{i \in S} \mu_i$, variance $\sigma^2(S) = \sum_{i \in S} \sigma_i^2$
- And $\frac{\widetilde{q}(S) - \mu(S)}{\sigma(S)}$ is $N(0, 1)$

Normal distribution example

- Suppose $\{\tilde{q}_1, \tilde{q}_2, \ldots, \tilde{q}_n\}$ are independent and normally distributed
  - Means μ_i, variances σ_i^2
- Then $\tilde{q}(S) = \sum_{i \in S} \tilde{q}_i$ remains normal
  - Mean $\mu(S) = \sum_{i \in S} \mu_i$, variance $\sigma^2(S) = \sum_{i \in S} \sigma_i^2$
- And $\frac{\tilde{q}(S) - \mu(S)}{\sigma(S)}$ is $N(0, 1)$
- Therefore $\tau = \Phi^{-1}(1 - \alpha)$

Using deterministic equivalent

Heuristics

- Computing M_k and S_k^2 for all routes not difficult
- Simple updating procedures when customers enter or leave routes in neighborhood search

- Remember, *variance* of a sum of independent random variables is the sum of the *variances* of the individual random variables

Using deterministic equivalent

Exact approaches

Laporte, Louveaux, and Mercure (1989): "subtour" elimination for 2-index formulation

- Consider customer set U
- Let $V_\alpha(U)$ be smallest integer s.t. $P\left(\sum_{i \in U} \tilde{q}_i > QV_\alpha(U)\right) \leq \alpha$
- "Subtour" elimination cut:

$$\sum_{i \in U, j \in \overline{U}} x_{ij} + \sum_{i \in \overline{U}, j \in U} x_{ij} \geq 2V_\alpha(U)$$

Using deterministic equivalent

Exact approaches

Laporte, Louveaux, and Mercure (1989): "subtour" elimination for 2-index formulation

- Consider customer set U
- Let $V_\alpha(U)$ be smallest integer s.t. $P\left(\sum_{i\in U}\tilde{q}_i > QV_\alpha(U)\right) \leq \alpha$
- "Subtour" elimination cut:

$$\sum_{i\in U, j\in \overline{U}} x_{ij} + \sum_{i\in \overline{U}, j\in U} x_{ij} \geq 2V_\alpha(U)$$

- We can determine $V_\alpha(U)$ as follows: $V_\alpha(U) - 1Q < M_U + \tau S_U \leq V_\alpha(U)Q$

Two-stage models: fixed routes

- What are fixed routes?

Two-stage models: fixed routes

- What are fixed routes?
  - Delivery routes used essentially unchanged daily for some period of time

Two-stage models: fixed routes

- What are fixed routes?
  - Delivery routes used essentially unchanged daily for some period of time
- Why use fixed routes?

Two-stage models: fixed routes

- What are fixed routes?
  - Delivery routes used essentially unchanged daily for some period of time
- Why use fixed routes?
  - Reduce costs
    - Simplify picking/staging costs at distribution center
    - Eliminate daily use of optimization software

Two-stage models: fixed routes

- What are fixed routes?
  - Delivery routes used essentially unchanged daily for some period of time
- Why use fixed routes?
  - Reduce costs
    - Simplify picking/staging costs at distribution center
    - Eliminate daily use of optimization software
  - Improve driver performance
    - Develop familiarity with a delivery area and set of customers

Two-stage models: fixed routes

- What are fixed routes?
  - Delivery routes used essentially unchanged daily for some period of time
- Why use fixed routes?
  - Reduce costs
    - Simplify picking/staging costs at distribution center
    - Eliminate daily use of optimization software
  - Improve driver performance
    - Develop familiarity with a delivery area and set of customers
  - Improve customer service
    - Driver develops relationship with customer
    - Driver performs additional services for customer

VRP with Stochastic Demands (VRPSD)

Vehicle Routing Problem with Stochastic Demands (collection version)

Given depot-based fleet of vehicles of capacity Q, travel cost matrix $\{c_{ij}\}$, and **uncertain** customer demands $\{\tilde{q}_i\}$ independent with known distributions, find set of depot-based vehicle tours that (*)

VRP with Stochastic Demands (VRPSD)

Vehicle Routing Problem with Stochastic Demands (collection version)

Given depot-based fleet of vehicles of capacity Q, travel cost matrix $\{c_{ij}\}$, and **uncertain** customer demands $\{\tilde{q}_i\}$ independent with known distributions, find set of depot-based vehicle tours that (*)

- Two-stage integer programming version
  - (*) Minimize **expected** total travel cost given a **recourse policy** (control decision strategy)

VRP with Stochastic Demands (VRPSD)

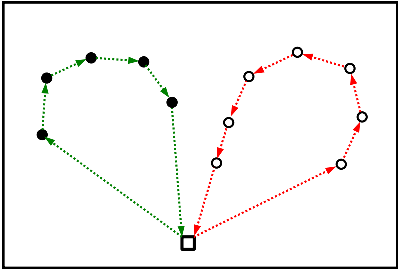
Vehicle Routing Problem with Stochastic Demands (collection version)

Given depot-based fleet of vehicles of capacity Q, travel cost matrix $\{c_{ij}\}$, and **uncertain** customer demands $\{\tilde{q}_i\}$ independent with known distributions, find set of depot-based vehicle tours that (*)

- Two-stage integer programming version
  - (*) Minimize **expected** total travel cost given a **recourse policy** (control decision strategy)
- *A priori* tours must be planned before uncertainty revealed
  - Parameter availability: customer demands known upon vehicle arrival

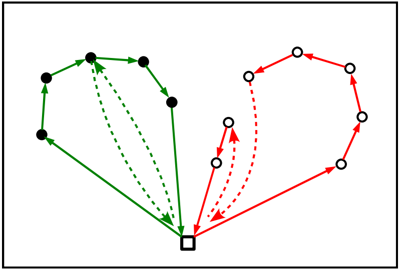
Two-stage model for VRPSD

1st stage: *A priori* tours



Minimize expected cost

2nd stage: Operational tours



Use recourse (control) policy

Dror et al. (1989) Recourse Policy

- Follow *a priori* tour
- When vehicle capacity met or exceeded, *detour to depot* to unload

Models

Two-stage stochastic integer program

$$\min_{x \in X} z = c^T x + E[\min_y \{q(\omega)^T y \mid Wy = h(\omega) - T(\omega)x \ , \ y \in Y\}]$$

$$s.t. \quad Ax = b$$

where X and/or Y impose integrality restrictions.

Models

Two-stage stochastic integer program

$$\min_{x \in X} z = c^T x + E[\min_y \{q(\omega)^T y \mid Wy = h(\omega) - T(\omega)x \ , \ y \in Y\}]$$

$$s.t. \quad Ax = b$$

where X and/or Y impose integrality restrictions.

Deterministic equivalent form

$$\min_{x \in X} z = c^T x + \mathcal{Q}(x)$$

$$s.t. \quad Ax = b$$

Computing $\mathcal{Q}(x)$

Consider single tour with homogeneous discrete customer demand distributions, and recourse only initiated if observed customer demand would exceed remaining vehicle capacity

- Tour $\mathcal{T} = \{1, 2, \cdots, n\}$
- $p_i(\delta)$ probability that customer i demand value is δ
- $\beta(i, s, q)$ probability of remaining capacity q after serving customer i, $s = 1$ if recourse action occurred at i, 0 otherwise

$$\beta(i, 0, q) = \sum_{s} \sum_{\bar{q} \in [q, Q]} \beta(i-1, s, \bar{q})\ p_i(\bar{q} - q)$$

$$\beta(i, 1, q) = \sum_{s} \sum_{\bar{q} \in [0, Q-q-1]} \beta(i-1, s, \bar{q})\ p_i(Q - q)$$

Computing $\mathcal{Q}(x)$

Consider single tour with homogeneous discrete customer demand distributions, and recourse only initiated if observed customer demand would exceed remaining vehicle capacity

- π_i probability of a tour failure and recourse at customer i

$$\pi_i = \sum_q \beta(i, 1, q)$$

- Expected recourse cost

$$\sum_{i \in \mathcal{T}} 2\pi_i \left(c_{i,0}\right)$$

Example $\mathcal{Q}(x)$ computation

Setting

- Suppose tour has three customers, and $Q = 2$
- Each customer demand distribution:

$$p(\delta) = \begin{cases} 0.2 & \delta = 0 \\ 0.8 & \delta = 1 \end{cases}$$

Example $\mathcal{Q}(x)$ computation

Computations

Customer 1
$\beta(1, s = 0, q = 0) = p(Q - q = 2) = 0$

Customer 2

Example $\mathcal{Q}(x)$ computation

Computations

Customer 1
$\beta(1, s = 0, q = 0) = p(Q - q = 2) = 0$
$\beta(1, s = 0, q = 1) = p(1) = 0.8$

Customer 2

Example $\mathcal{Q}(x)$ computation

Computations

Customer 1

$\beta(1, s = 0, q = 0) = p(Q - q = 2) = 0$

$\beta(1, s = 0, q = 1) = p(1) = 0.8$

$\beta(1, s = 0, q = 2) = p(0) = 0.2$

Customer 2

Example $\mathcal{Q}(x)$ computation

Computations

Customer 1

$\beta(1, s = 0, q = 0) = p(Q - q = 2) = 0$

$\beta(1, s = 0, q = 1) = p(1) = 0.8$

$\beta(1, s = 0, q = 2) = p(0) = 0.2$

$\beta(1, s = 1, q) = 0$

Customer 2

Example $Q(x)$ computation

Computations

Customer 1

$\beta(1, s = 0, q = 0) = p(Q - q = 2) = 0$

$\beta(1, s = 0, q = 1) = p(1) = 0.8$

$\beta(1, s = 0, q = 2) = p(0) = 0.2$

$\beta(1, s = 1, q) = 0$

Customer 2

$\beta(2, s = 0, q = 0) = \beta(1, 0, 1)p(1) = 0.64$

Example $Q(x)$ computation

Computations

Customer 1

$\beta(1, s = 0, q = 0) = p(Q - q = 2) = 0$

$\beta(1, s = 0, q = 1) = p(1) = 0.8$

$\beta(1, s = 0, q = 2) = p(0) = 0.2$

$\beta(1, s = 1, q) = 0$

Customer 2

$\beta(2, s = 0, q = 0) = \beta(1, 0, 1)p(1) = 0.64$

$\beta(2, s = 0, q = 1) = \beta(1, 0, 1)p(0) + \beta(1, 0, 2)p(1) =$
$(0.8)(0.2) + (0.2)(0.8) = 0.32$

Example $\mathcal{Q}(x)$ computation

Computations

Customer 1

$\beta(1, s = 0, q = 0) = p(Q - q = 2) = 0$

$\beta(1, s = 0, q = 1) = p(1) = 0.8$

$\beta(1, s = 0, q = 2) = p(0) = 0.2$

$\beta(1, s = 1, q) = 0$

Customer 2

$\beta(2, s = 0, q = 0) = \beta(1, 0, 1)p(1) = 0.64$

$\beta(2, s = 0, q = 1) = \beta(1, 0, 1)p(0) + \beta(1, 0, 2)p(1) =$
$(0.8)(0.2) + (0.2)(0.8) = 0.32$

$\beta(2, s = 0, q = 2) = \beta(1, 0, 2)p(0) = 0.04$

Example $\mathcal{Q}(x)$ computation

Computations

Customer 1

$\beta(1, s = 0, q = 0) = p(Q - q = 2) = 0$
$\beta(1, s = 0, q = 1) = p(1) = 0.8$
$\beta(1, s = 0, q = 2) = p(0) = 0.2$
$\beta(1, s = 1, q) = 0$

Customer 2

$\beta(2, s = 0, q = 0) = \beta(1, 0, 1)p(1) = 0.64$
$\beta(2, s = 0, q = 1) = \beta(1, 0, 1)p(0) + \beta(1, 0, 2)p(1) =$
$(0.8)(0.2) + (0.2)(0.8) = 0.32$
$\beta(2, s = 0, q = 2) = \beta(1, 0, 2)p(0) = 0.04$
$\beta(2, s = 1, q) = 0$

Example $Q(x)$ computation

Computations

Customer 3

Example $\mathcal{Q}(x)$ computation

Computations

Customer 3
$\beta(3, s = 0, q = 0) = \beta(2, 0, 0)p(0) + \beta(2, 0, 1)p(1) = (0.64)(0.2) + (0.32)(0.8) = 0.384$

Example $Q(x)$ computation

Computations

Customer 3

$\beta(3, s = 0, q = 0) = \beta(2, 0, 0)p(0) + \beta(2, 0, 1)p(1) =$
$(0.64)(0.2) + (0.32)(0.8) = 0.384$

$\beta(3, s = 0, q = 1) = \beta(2, 0, 1)p(0) + \beta(2, 0, 2)p(1) =$
$(0.32)(0.2) + (0.04)(0.8) = 0.096$

Example $\mathcal{Q}(x)$ computation

Computations

Customer 3

$\beta(3, s = 0, q = 0) = \beta(2, 0, 0)p(0) + \beta(2, 0, 1)p(1) = (0.64)(0.2) + (0.32)(0.8) = 0.384$

$\beta(3, s = 0, q = 1) = \beta(2, 0, 1)p(0) + \beta(2, 0, 2)p(1) = (0.32)(0.2) + (0.04)(0.8) = 0.096$

$\beta(3, s = 0, q = 2) = \beta(2, 0, 2)p(0) = (0.04)(0.2) = 0.008$

Example $Q(x)$ computation

Computations

Customer 3

$\beta(3, s = 0, q = 0) = \beta(2, 0, 0)p(0) + \beta(2, 0, 1)p(1) =$
$(0.64)(0.2) + (0.32)(0.8) = 0.384$

$\beta(3, s = 0, q = 1) = \beta(2, 0, 1)p(0) + \beta(2, 0, 2)p(1) =$
$(0.32)(0.2) + (0.04)(0.8) = 0.096$

$\beta(3, s = 0, q = 2) = \beta(2, 0, 2)p(0) = (0.04)(0.2) = 0.008$

$\beta(3, s = 1, q = 1) = \beta(2, 0, 0)p(1) = (0.64)(0.8) = 0.512$

Example $Q(x)$ computation

Computations

Customer 3

$\beta(3, s = 0, q = 0) = \beta(2, 0, 0)p(0) + \beta(2, 0, 1)p(1) =$
$(0.64)(0.2) + (0.32)(0.8) = 0.384$

$\beta(3, s = 0, q = 1) = \beta(2, 0, 1)p(0) + \beta(2, 0, 2)p(1) =$
$(0.32)(0.2) + (0.04)(0.8) = 0.096$

$\beta(3, s = 0, q = 2) = \beta(2, 0, 2)p(0) = (0.04)(0.2) = 0.008$

$\beta(3, s = 1, q = 1) = \beta(2, 0, 0)p(1) = (0.64)(0.8) = 0.512$

Failure Probabilities $\pi_1 = \pi_2 = 0$; $\pi_3 = 0.512$

Recourse Cost $= 2(0.512)c_{03} = 1.024c_{03}$

Integer L-shaped method

Extension of Van Slyke and Wets (1969) for SP

- Bender's decomposition approach
- Use cuts on first-stage decisions to:
    - Ensure second-stage *feasibility* with feasibility cuts
    - Create a linear approximation of $\mathcal{Q}(x)$ with optimality cuts

Integer L-shaped method

Extension of Van Slyke and Wets (1969) for SP

- Bender's decomposition approach
- Use cuts on first-stage decisions to:
  - Ensure second-stage *feasibility* with feasibility cuts
  - Create a linear approximation of $\mathcal{Q}(x)$ with optimality cuts

First-stage feasibility is trivial for VRPSD, so focus only on optimality cuts.

Integer L-shaped method for VRPSD

Gendreau, Laporte, and Seguin (1995); Laporte, Louveaux, Van Hamme (2002)

- Add constraint that expected demand of each *a priori* tour cannot exceed vehicle capacity

Integer L-shaped method for VRPSD

Gendreau, Laporte, and Seguin (1995); Laporte, Louveaux, Van Hamme (2002)

- Add constraint that expected demand of each *a priori* tour cannot exceed vehicle capacity

Relaxed Formulation

$$\min_{i<j} c_{ij}x_{ij} + \theta$$

subject to constraints of two-index undirected CVRP problem, where customer demand $\tilde{q}_i = E[\tilde{q}_i]$ for "subtour elimination"

Integer L-shaped method for VRPSD

Basic branch-and-cut approach

- Let θ^v be a lower bound on $\mathcal{Q}(x)$, and fathom if $cx^v + \theta^v \geq \bar{z}$
- When integer x^v found, compute $\mathcal{Q}(x^v)$ and update best found solution
  - If $\theta^v \geq Q(x^v)$, fathom (branch is optimal)
  - Else, introduce cut to move away from this solution, and continue

Integer L-shaped method for VRPSD

Basic branch-and-cut approach

- Let θ^v be a lower bound on $\mathcal{Q}(x)$, and fathom if $cx^v + \theta^v \geq \bar{z}$
- When integer x^v found, compute $\mathcal{Q}(x^v)$ and update best found solution
  - If $\theta^v \geq Q(x^v)$, fathom (branch is optimal)
  - Else, introduce cut to move away from this solution, and continue

Lower bounding of $\mathcal{Q}(x)$; Laporte et al. (2002)

Add cuts that are valid lower bounds on $\mathcal{Q}(x^v)$ (regardless of x^v integer)

- Detailed, but based on the idea of approximating expected recourse cost of failure by considering first tour failure only

Integer L-shaped method for VRPSD

Computational results from Laporte, et al. (2002)

- Heterogeneous Poisson demands
- Number of customers $n \in \{25, 50, 75, 100\}$
- Number of vehicles $m \in \{2, 3, 4\}$, 2 only for $n \geq 75$
- "Fill rate" $0.9 =$ Total expected demand divided by total capacity of all vehicles
  - As fill rate approaches 1, evidence suggests computational difficult increases much faster than linearly

Sample average approximation (SAA)

Kleywegt, Shapiro, and Homem-de-Mello (2002)

- An alternative approach for solving two-stage SIP

SAA

Sample average approximation (SAA)

Kleywegt, Shapiro, and Homem-de-Mello (2002)

- An alternative approach for solving two-stage SIP

SAA

- Generate a *sample* of all realizations of uncertain parameters

Sample average approximation (SAA)

Kleywegt, Shapiro, and Homem-de-Mello (2002)

- An alternative approach for solving two-stage SIP

SAA

- Generate a *sample* of all realizations of uncertain parameters
- Solve deterministic problem, explicitly satisfying all constraints for each realization in the sample

Sample average approximation (SAA)

Kleywegt, Shapiro, and Homem-de-Mello (2002)

- An alternative approach for solving two-stage SIP

SAA

- Generate a *sample* of all realizations of uncertain parameters
- Solve deterministic problem, explicitly satisfying all constraints for each realization in the sample
- Use many such samples to:
  - Identify best plan
  - Approximate the optimality gap

Sample average approximation (SAA)

Kleywegt, Shapiro, and Homem-de-Mello (2002)

- An alternative approach for solving two-stage SIP

SAA

- Generate a *sample* of all realizations of uncertain parameters
- Solve deterministic problem, explicitly satisfying all constraints for each realization in the sample
- Use many such samples to:
  - Identify best plan
  - Approximate the optimality gap

- See Verweij, Ahmed, Kleywegt, Nemhauser, Shapiro (2003) for stochastic routing applications

Limitations of VRPSD models

Metaheuristics for practical instance sizes

- Gendreau et al. (1996) extend TABUROUTE to problems with stochastic demands: TABUSTOCH
- Reasonable performance, although solutions degrade as n increases to 50

Limitations of VRPSD models

Detour-to-depot recourse policy is limited

Limitations of VRPSD models

Detour-to-depot recourse policy is limited

- Each vehicle operates its *a priori* tour independently

Limitations of VRPSD models

Detour-to-depot recourse policy is limited

- Each vehicle operates its *a priori* tour independently
- Enables analysis, but does not provide any opportunity for *risk pooling*

Limitations of VRPSD models

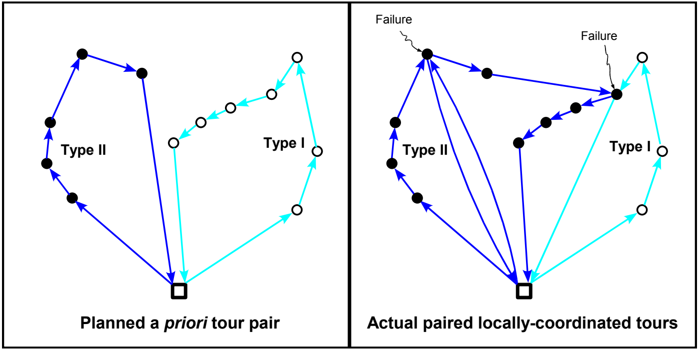
Detour-to-depot recourse policy is limited

- Each vehicle operates its *a priori* tour independently
- Enables analysis, but does not provide any opportunity for *risk pooling*

Multi-vehicle coordinated recourse policies

- Erera (2000): analysis of many policies using techniques of *continuous approximation*
- Ak and Erera (2007): detailed analysis and tabu search heuristic for two-vehicle sharing policy

Two-vehicle sharing recourse policy



Paired locally-coordinated (PLC) recourse; Ak and Erera (2007)

- Type I capacity failure: unserved customers appended to type II tour
- Type II capacity failure: use detour-to-depot

Tabu search for PLC recourse strategy

PLC Tabu Search

Adapted from Gendreau et al. (1996)

- Exact recursive expected recourse cost $\mathcal{Q}(x)$ computation for a given solution for homogeneous discrete demand distributions
  - Condition on the customer η in the Type I tour where failure occurs, with probability $\overline{q}_\eta$
  - Insert customers $\eta + 1, \ldots$ into Type II tour after final customer, and use detour-to-depot
- Randomized neighborhood $N(p, r, q)$
  - Each of q randomly selected customers is reinserted before or after one of p randomly selected close neighbors from the list of r nearest

Results for PLC recourse strategy

Number of Customers	Center Depot			
	DD 1	DD 2	DD 3	DD 4
10	0.38%	0.31%	0.91%	0.83%
25	2.04%	1.75%	1.54%	2.88%
50	5.67%	3.88%	2.64%	5.03%
100	8.42%	6.42%	4.87%	9.61%
150	11.17%	8.47%	7.02%	10.73%

Table: Average percent improvement in expected travel cost generated by the PLC recourse strategy for test problems; average over ten instances

Results for PLC recourse strategy

Number of Customers	Corner Depot			
	DD 1	DD 2	DD 3	DD 4
10	4.01%	4.16%	0.38%	3.04%
25	5.85%	5.57%	4.95%	6.19%
50	8.66%	8.83%	7.53%	9.23%
100	15.46%	10.60%	10.05%	12.65%
150	15.88%	11.21%	10.81%	12.70%

Table: Average percent improvement in expected travel cost generated by the PLC recourse strategy for test problems; average over ten instances

Limitations of VRPSD models

Physical capacity Q does not create a need for additional vehicles

Limitations of VRPSD models

Physical capacity Q does not create a need for additional vehicles

Single vehicle feasibility

There exists a feasible solution to VRPSDC problems in which a single tour is planned from the depot.

Limitations of VRPSD models

Physical capacity Q does not create a need for additional vehicles

Single vehicle feasibility

There exists a feasible solution to VRPSDC problems in which a single tour is planned from the depot.

- Remember, if capacity fails then we can always *detour-to-depot* to unload

Limitations of VRPSD models

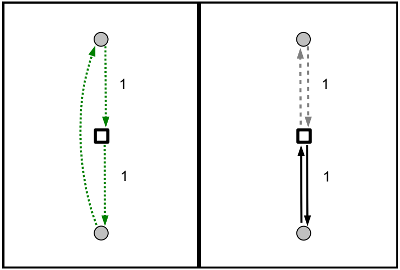
Using multiple vehicles to reduce expected cost

Any "vehicle" beyond the first used can be interpreted as a *pre-emptive* detour to the depot for vehicle one!

Limitations of VRPSD models

Using multiple vehicles to reduce expected cost

Any "vehicle" beyond the first used can be interpreted as a *pre-emptive* detour to the depot for vehicle one!



- $\tilde{q}_i = 1$ or 2 each with probability $\frac{1}{2}$, and $Q = 3$
- Expected cost (left) $= 4.5$, expected cost (right) $= 4$

Limitations of VRPSD models

Time constraints are the real reason why multiple vehicles are needed for stochastic routing problems

Ad-hoc modeling

- Insist on target fleet size m that implicitly limits tour durations

Limitations of VRPSD models

Time constraints are the real reason why multiple vehicles are needed for stochastic routing problems

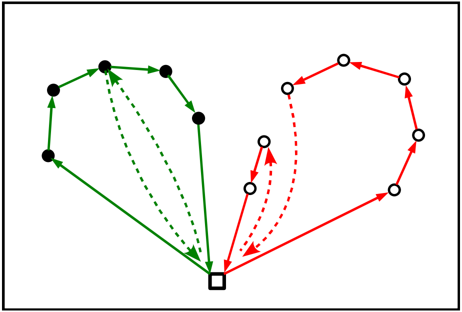
Ad-hoc modeling

- Insist on target fleet size m that implicitly limits tour durations

Explicit modeling

- VRPSD with Duration Constraints (VRPSD-DC)
- VRPSDC with Time Window Constraints

Detour-to-depot adds to tour duration



Assumptions

- All travel times known with certainty
- Uncertain number and location of recourse actions creates uncertainty in tour duration

Using Robust Constraints for VRPSD-DC

Morales (2006); Erera, Morales, and Savelsbergh (2010, to appear)

Robust duration constraints

Using Robust Constraints for VRPSD-DC

Morales (2006); Erera, Morales, and Savelsbergh (2010, to appear)

Robust duration constraints

- Use a two-stage model to minimize expected tour costs under a recourse policy

Using Robust Constraints for VRPSD-DC

Morales (2006); Erera, Morales, and Savelsbergh (2010, to appear)

Robust duration constraints

- Use a two-stage model to minimize expected tour costs under a recourse policy
- No recourse for a tour requiring too much time!
  1. Chance constraint, or
  2. Objective function penalty, or
  3. **Robust constraint**

Modeling using robust constraints

- **Uncertainty space** $\mathcal{U}$: a subset (not necessarily strict) of the *support* of the random parameters

Modeling using robust constraints

- **Uncertainty space** $\mathcal{U}$: a subset (not necessarily strict) of the *support* of the random parameters
- We will say that a second-stage constraint is a **robust constraint** if it must hold for every parameter realization in $\mathcal{U}$

Modeling using robust constraints

- **Uncertainty space** $\mathcal{U}$: a subset (not necessarily strict) of the *support* of the random parameters
- We will say that a second-stage constraint is a **robust constraint** if it must hold for every parameter realization in $\mathcal{U}$
    - Note: if $\mathcal{U}$ contains all outcomes, this idea is covered by two-stage recourse model formulations

Formulation

VRPSD with Robust Duration Constraints

- Set of n customers, stochastic integer demand $\widetilde{q}_i$
- Recourse policy $\mathcal{P}$, separable by tour
- Change in fixed tour duration due to recourse, $\phi(T, \mathcal{P}, q)$

$$\min_{T_1,\ldots,T_m} \sum_{k=1}^{m} t(T_k) + E_{\widetilde{q}}[\phi(T_k, \mathcal{P}, \widetilde{q})]$$

st

Each customer on single tour

$$t(T_k) + \phi(T_k, \mathcal{P}, q) \leq D \quad \forall \;\; q \in \mathcal{U}$$

Formulation

VRPSD with Robust Duration Constraints

- Set of n customers, stochastic integer demand $\tilde{q}_i$
- Recourse policy $\mathcal{P}$, separable by tour
- Change in fixed tour duration due to recourse, $\phi(T, \mathcal{P}, q)$

$$\min_{T_1,\dots,T_m} \sum_{k=1}^{m} t(T_k) + E_{\tilde{q}}[\phi(T_k, \mathcal{P}, \tilde{q})]$$

$$st$$

Each customer on single tour

$$t(T_k) + \max_{q \in \mathcal{U}} \phi(T_k, \mathcal{P}, q) \leq D$$

Adversarial problem

Adversarial problem

$$\max_{q \in \mathcal{U}} \phi(T, \mathcal{P}, q)$$

- Separability of recourse policy allows tour-by-tour evaluation

Adversarial problem

Adversarial problem

$$\max_{q \in \mathcal{U}} \phi(T, \mathcal{P}, q)$$

- Separability of recourse policy allows tour-by-tour evaluation

Is adversarial problem challenging?

- Is $q^* = \overline{q}$?

Adversarial problem

Adversarial problem

$$\max_{q \in \mathcal{U}} \phi(T, \mathcal{P}, q)$$

- Separability of recourse policy allows tour-by-tour evaluation

Is adversarial problem challenging?

- Is $q^* = \overline{q}$?
  - Not worst for most recourse policies $\mathcal{P}$

Adversarial problem

Adversarial problem

$$\max_{q \in \mathcal{U}} \phi(T, \mathcal{P}, q)$$

- Separability of recourse policy allows tour-by-tour evaluation

Is adversarial problem challenging?

- Is $q^* = \overline{q}$?
  - Not worst for most recourse policies $\mathcal{P}$
- Is $q^* \in \{q \in \mathbb{Z}_+^n \ : \ q_i \in \{\underline{q}_i, \overline{q}_i\} \ \forall \ i\}$?

Adversarial problem

Adversarial problem

$$\max_{q \in \mathcal{U}} \phi(T, \mathcal{P}, q)$$

- Separability of recourse policy allows tour-by-tour evaluation

Is adversarial problem challenging?

- Is $q^* = \overline{q}$?
  - Not worst for most recourse policies $\mathcal{P}$
- Is $q^* \in \{q \in \mathbb{Z}^n_+ \ : \ q_i \in \{\underline{q}_i, \overline{q}_i\} \ \forall\, i\}$?
  - Again, no.

Solving adversarial problem for detour-to-depot recourse

Conceptual idea

Given a tour T, adversary can choose the demands q_i of each customer to **maximize** the additional duration of the tour due to recourse actions

Solving adversarial problem for detour-to-depot recourse

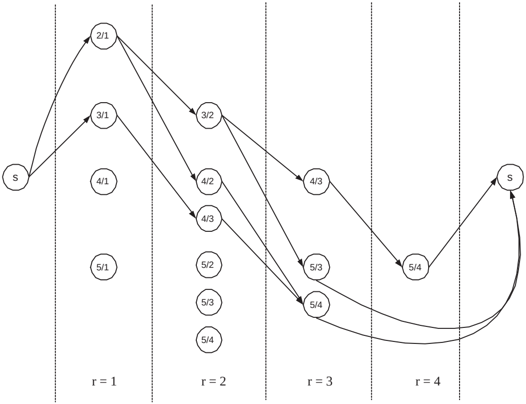
Conceptual idea

Given a tour T, adversary can choose the demands q_i of each customer to **maximize** the additional duration of the tour due to recourse actions

Polynomial longest-path problem

For a tour T with n customers, the maximum duration can be computed in $O(n^4)$ by generating an acyclic network and solving a longest-path problem

Previous-recourse Network $\mathcal{G}_1$



- Cost of arcs into node $(r, i/j)$ is the additional travel time: $2t_{0i}$

Previous-recourse network $\mathcal{G}_1$

Recourse conditions define which arcs exist

Observation

Given a demand realization q such that a recourse action occurs at customer i, then remaining vehicle capacity when departing i is $Q - q(i)$.

Previous-recourse network $\mathcal{G}_1$

Recourse conditions define which arcs exist

Observation

Given a demand realization q such that a recourse action occurs at customer i, then remaining vehicle capacity when departing i is $Q - q(i)$.

Observation

If a recourse action occurs at i, and the prior recourse occurred at j, then there exists a minimum demand $\underline{q}(i/j)$ at i that can cause recourse:

$$\underline{q}(i/j) = \max \left\{ 1, Q + 1 - \sum_{\ell=j}^{i-1} \overline{q}(\ell) \right\}$$

Recourse conditions for $\mathcal{G}_1$

Theorem

Assume there exists $q \in \mathcal{U}$ such that the $(r-1)$-th recourse occurs at j and the r-th recourse occurs at $i > j$. The $(r+1)$-th recourse can occur at $k > i$ if and only if

$$\max\left\{\underline{q}(i/j), \underline{q}(i)\right\} + \sum_{\ell=i+1}^{k-1} \underline{q}(\ell) \leq Q \quad \text{and} \quad \sum_{\ell=i}^{k} \overline{q}(\ell) \geq Q+1$$

How do duration constraints affect the solution?

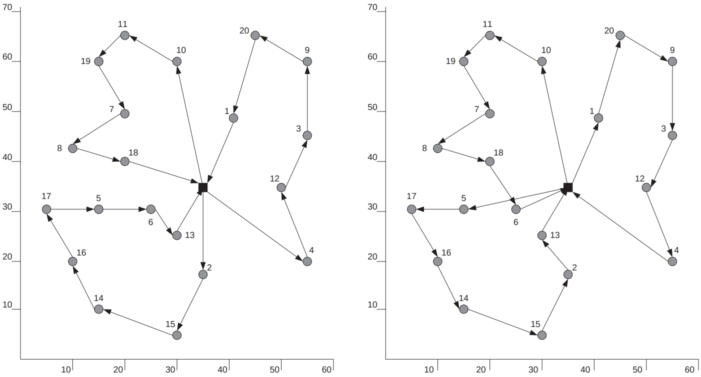


Figure 7: Best unconstrained solution and best constrained solution ($\alpha = 0.95$).

How do duration constraints affect the solution?

Table: Unconstrained version (total expected time 345.84)

Tour	{4, ... ,1}	{10,...,18}	{ 2,...,13}
Fixed L	109.90	93.39	107.42
Expected $\mathcal{L}_E$	115.73	99.34	130.77
Max $\mathcal{L}$	173.14	145.88	168.25

Table: Constrained version (total expected time 352.32)

Tour	{1, ... ,4}	{10, ... ,6}	{5, ... ,13}
Fixed L	109.90	99.93	102.40
Expected $\mathcal{L}_E$	118.73	115.03	118.56
Max $\mathcal{L}$	159.90	152.43	163.23

Impact of robust duration constraints

(N, Q, σ, m)	α	$\Delta\mathcal{L}_E$	$m + z$
(100, Q_1, High, 6)	0.95	0.69%	6
	0.85	1.60%	6
	0.75	-0.31%	7
(100, Q_1, Med, 6)	0.95	1.01%	6
	0.85	2.86%	6
	0.75	-0.41%	7
(100, Q_1, Low, 6)	0.95	1.18%	6
	0.85	1.18%	6
	0.75	5.22%	7
(100, Q_2, High, 3)	0.95	0.67%	3
	0.85	1.28%	3
	0.75	-1.66%	4
(100, Q_2, Med, 3)	0.95	0.17%	3
	0.85	0.61%	3
	0.75	-1.15%	4
(100, Q_2, Low, 3)	0.95	0.17%	3
	0.85	0.89%	3
	0.75	-0.74%	4

Recourse policies for time-constrained routing problems

Parameter availability

Assume that all customers to be served, and their demands, are known prior to vehicle loading

Recourse policies for time-constrained routing problems

Parameter availability

Assume that all customers to be served, and their demands, are known prior to vehicle loading

Question

Can we create a plan that preserves most of the benefits of traditional *a priori* routes, but can be used for problems with hard time constraints?

VRP with Stochastic Demand and Customers and Time Windows

Definition (VRPSDC-TW)

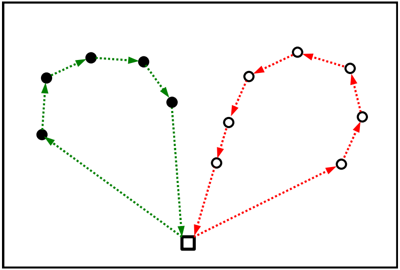
- Set of n possible customers
- Stochastic integer demand $\{\tilde{q}_i\}$, non-zero demand probability $\{p_i\}$
- Time windows $[e_i^1, \ell_i^1]$ and $[e_i^2, \ell_i^2]$
- Recourse (control) strategy

Find:

- Set of fixed routes such that
  - Each customer served by exactly one fixed route
  - Control strategy yields actual routes that are capacity and time feasible
  - Total expected travel costs given $\mathcal{P}$ minimized

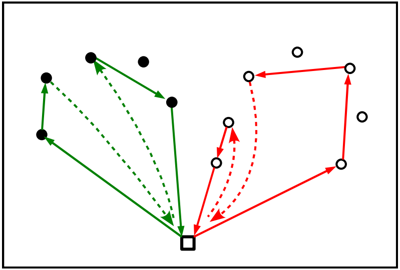
Traditional recourse policy for VRPSDC

1st stage: *A priori* tours



Minimize expected cost

2nd stage: Operational tours



Use fixed recourse policy

Bertsimas (1992) Type 'B' Recourse Policy

- Follow *a priori* tour, skipping customers with no demand
- When vehicle capacity met or exceeded, *detour to depot* to unload

Recourse Strategy for Time Hedging

Erera, Savelsbergh, Uyar (2009)

Introduce backup, or secondary, vehicles

- Each customer assigned to at most 2 fixed “routes”

Recourse Strategy for Time Hedging

Erera, Savelsbergh, Uyar (2009)

Introduce backup, or secondary, vehicles

- Each customer assigned to at most 2 fixed “routes”
    - Primary and secondary
    - A “route” now is simply an unordered set of customers

Recourse Strategy for Time Hedging

Erera, Savelsbergh, Uyar (2009)

Introduce backup, or secondary, vehicles

- Each customer assigned to at most 2 fixed "routes"
  - Primary and secondary
  - A "route" now is simply an unordered set of customers
- Recourse decisions determined by problem of finding set of actual routes such that:
  - Each customer served by either its primary or its secondary vehicle
  - All actual routes time and capacity feasible
  - Total travel cost of actual routes is minimized

Recourse Strategy for Time Hedging

Erera, Savelsbergh, Uyar (2009)

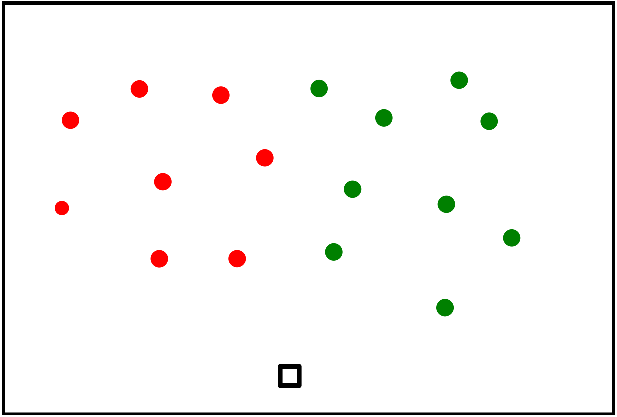
Introduce backup, or secondary, vehicles

- Each customer assigned to at most 2 fixed “routes”
  - Primary and secondary
  - A “route” now is simply an unordered set of customers
- Recourse decisions determined by problem of finding set of actual routes such that:
  - Each customer served by either its primary or its secondary vehicle
  - All actual routes time and capacity feasible
  - Total travel cost of actual routes is minimized

Features

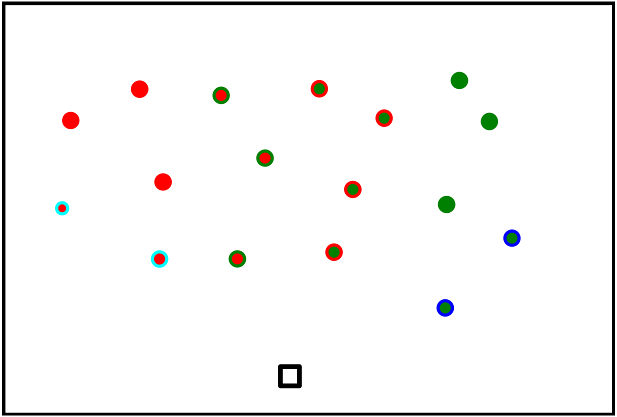
1. Preserves benefits of traditional fixed routes
2. Allows flexibility to restore feasibility and reduce costs

Primary+Secondary Recourse



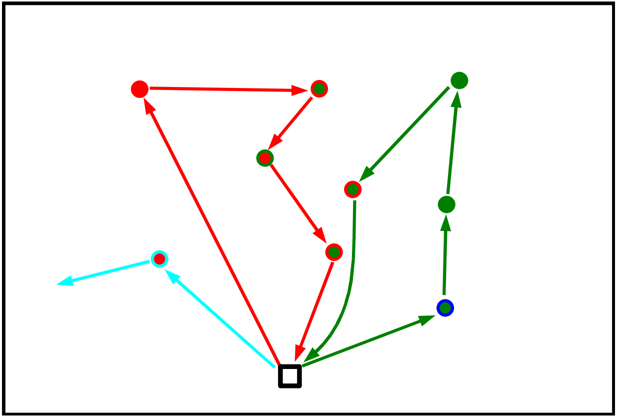
Primary assignments

Primary+Secondary Recourse



Primary + secondary assignments

Primary+Secondary Recourse



Actual operational routes

United Distributors, Atlanta, USA

- Distributor of beer, wine, and spirits
- Serve northern Georgia: (150 by 150 km)
- Customer set
  - approximately 2500 known customer locations
  - Wide variation in p_i
  - Moderate variation in $\tilde{q}_i$
- Single depot
- Fleet of approximately 50 homogeneous vehicles

United Distributors, Atlanta, USA

- Distributor of beer, wine, and spirits
- Serve northern Georgia: (150 by 150 km)
- Customer set
  - approximately 2500 known customer locations
  - Wide variation in p_i
  - Moderate variation in $\tilde{q}_i$
- Single depot
- Fleet of approximately 50 homogeneous vehicles

Conjecture

Problem size indicates that heuristic approach appropriate!

Customer classification

- Company prefers different fixed routes for each delivery day
  - Mon, Tue, Wed, Thu, Fri
- Large fraction of customers have very low probability of delivery

Probability Range	M	Tu	W	Th	F
$p_i < 10\%$	522	621	820	870	925
$p_i \geq 10\%$	**330**	**1453**	**1366**	**1647**	**1573**

Customer classification

- Company prefers different fixed routes for each delivery day
  - Mon, Tue, Wed, Thu, Fri
- Large fraction of customers have very low probability of delivery

Probability Range	M	Tu	W	Th	F
$p_i < 10\%$	522	621	820	870	925
$p_i \geq 10\%$	**330**	**1453**	**1366**	**1647**	**1573**

Customer partition

- High probability customers assigned to primary and secondary routes
- Low probability customers only added dynamically to operational routes during recourse

Time Window Characteristics

Earliest 1	Latest 1	Earliest 2	Latest 2	Customers	%
9:00 AM	11:00 AM	2:00 PM	6:00 PM	1240	28.47
8:00 AM	4:00 PM	/	/	902	20.71
6:00 AM	11:00 AM	/	/	359	8.24
11:00 AM	6:00 PM	/	/	234	5.37
10:00 AM	6:00 PM	/	/	177	4.06
2:00 PM	6:00 PM	/	/	114	2.62
6:00 AM	1:00 PM	/	/	96	2.20
8:00 AM	1:00 PM	/	/	80	1.84
8:00 AM	12:00 PM	/	/	75	1.72
			TOTAL=	3277	75.23

Using two-stage model with chance constraints

Find primary assignments:

Primary assignments

- Set of n possible customers
- Stochastic integer demand $\{\tilde{q}_i\}$, non-zero demand probability $\{p_i\}$
- Time windows $[e_i^1, \ell_i^1]$ and $[e_i^2, \ell_i^2]$
- Skipping policy $\mathcal{P}$

Find:

- Set of fixed routes such that
  - Each customer served by exactly one fixed route
  - Policy $\mathcal{P}$ creates feasible actual routes **with high probability**
  - Total expected travel costs given $\mathcal{P}$ minimized

Primary routes heuristic

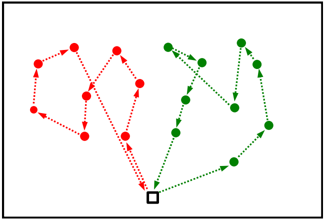
Main Ideas

- Construct primary routes via sequential insertion
- Periodic calls to local search improvement routine
- Evaluate feasibility and expected travel cost via sampling, *assuming operational routes will be constructed by skipping recourse strategy only*

Primary routes heuristic

Main Ideas

- Construct primary routes via sequential insertion
- Periodic calls to local search improvement routine
- Evaluate feasibility and expected travel cost via sampling, *assuming operational routes will be constructed by skipping recourse strategy only*



Primary routes planned as traditional fixed routes

Insertion feasibility

Capacity Feasibility

- Central limit theorem for tour demand normality
- Use traditional chance constraint form: $M_k + \tau S_k \leq Q$, with i added
- τ corresponds to $\alpha = 0.90$

Insertion feasibility

Capacity Feasibility

- Central limit theorem for tour demand normality
- Use traditional chance constraint form: $M_k + \tau S_k \leq Q$, with i added
- τ corresponds to $\alpha = 0.90$

Time Window Feasibility

- Using $\{p_j\}$, generate Monte Carlo sample of N customer realizations
- Customer i in all realizations (conditional sample)
- Time windows must be satisfied in fraction β of realizations ($\beta = 0.80$)

Secondary routes heuristic

- Generate a sample of realizations
- Solve a simple control problem for each realization
  - First apply simple skipping strategy to each primary route
  - Select customer on infeasible route to eject at random and find feasible reinsert location that minimizes change in route quality
  - Repeat until all routes feasible
  - Apply improvement local search to improve route quality
  - Record route serving each customer
- Most frequent route serving each customer, excluding the primary route, is secondary assignment

Operational routes heuristic

- Given a single actual realization of customers and their actual demands
- First apply simple skipping strategy to each primary route
- Restore feasibility using secondary assignments
  - Select customer on infeasible route to eject at random and find feasible reinsert location on secondary route that minimizes change in route quality
  - Repeat until all routes feasible
- Apply improvement local search to improve route quality
- Insert all low-probability customers arriving which do not have primary+secondary assignments
- Apply improvement local search to improve route quality
- Apply route elimination, respecting primary+secondary assignments

Thursdays: Comparison with History

Day	Routes	Customers	Total Miles	Travel Min	% in Miles	% in Min
1.H	43	856	5360	8428	/	/
1.GT	40	856	4459	7080	16.80	15.99
2.H	42	863	5245	8240	/	/
2.GT	40	863	4318	6872	17.67	16.61
3.H	43	876	5638	8805	/	/
3.GT	40	876	4495	7174	20.27	18.52
4.H	42	904	5587	8752	/	/
4.GT	42	904	4808	7588	13.94	13.30
5.H	42	839	5195	8120	/	/
5.GT	41	839	4363	6946	16.01	14.47

Thursdays: Comparison with History

Day	Routes	Customers	Total Miles	Travel Min	% in Miles	% in Min
1.H	43	856	5360	8428	/	/
1.GT	40	856	4459	7080	16.80	15.99
2.H	42	863	5245	8240	/	/
2.GT	40	863	4318	6872	17.67	16.61
3.H	43	876	5638	8805	/	/
3.GT	40	876	4495	7174	20.27	18.52
4.H	42	904	5587	8752	/	/
4.GT	42	904	4808	7588	13.94	13.30
5.H	42	839	5195	8120	/	/
5.GT	41	839	4363	6946	16.01	14.47

Route assignments

- 60-65 % of customers served by primary route
- 7 % of customers are dynamic (not on planned routes)

Impact of Sample Size N

Table: Fixed Route Results for Different Sample Size Parameter Values

N	Run Time (hours)	Avg. Time Feasibility	Final Number of Routes
500	1.70	0.839	43
1000	3.24	0.860	43
2000	10.03	0.878	43
3000	15.80	0.882	42

Impact of Sample Size N

Table: Daily Route Results for Different Sample Size Parameter Values

N	500	1000	2000	3000
Number of infeasible days	0	0	0	0
Avg. travel time	4,411	4,503	4,534	4,485
Avg. number of vehicles	39.83	40.25	40.50	39.25
Max. number of vehicles	42	42	42	41
Percentage of customers visited by primary vehicle	62%	64%	63%	64 %
Run times (secs.)	14.75	14.17	14.83	14.50

Approximating multiple stage problems

- When are multi-stage models appropriate?
  - When decisions made during each stage impact the initial state during the next (and future) stages

Approximating multiple stage problems

- When are multi-stage models appropriate?
    - When decisions made during each stage impact the initial state during the next (and future) stages
- Multi-stage models capture the true process of uncertain information becoming known during stages

Approximating multiple stage problems

- When are multi-stage models appropriate?
  - When decisions made during each stage impact the initial state during the next (and future) stages
- Multi-stage models capture the true process of uncertain information becoming known during stages
- A reasonable approximation, however, is to assume that **all** uncertainty is revealed after the first (planning) stage
  - A direct extension of rolling horizon models that assume all uncertainty is revealed during the planning stage

Approximating multiple stage problems

Rolling horizon two-stage approximation

Approximating multiple stage problems

Rolling horizon two-stage approximation

- Specify a planning horizon of a number of time periods

Approximating multiple stage problems

Rolling horizon two-stage approximation

- Specify a planning horizon of a number of time periods
- Partition planning horizon into two stages
    - Stage 1 periods: little to no uncertainty in parameters, and not modeled
    - Stage 2 periods: some parameters modeled with uncertainty

Approximating multiple stage problems

Rolling horizon two-stage approximation

- Specify a planning horizon of a number of time periods
- Partition planning horizon into two stages
    - Stage 1 periods: little to no uncertainty in parameters, and not modeled
    - Stage 2 periods: some parameters modeled with uncertainty
- Solve two-stage model, assuming that all uncertainty is revealed in the second stage

Approximating multiple stage problems

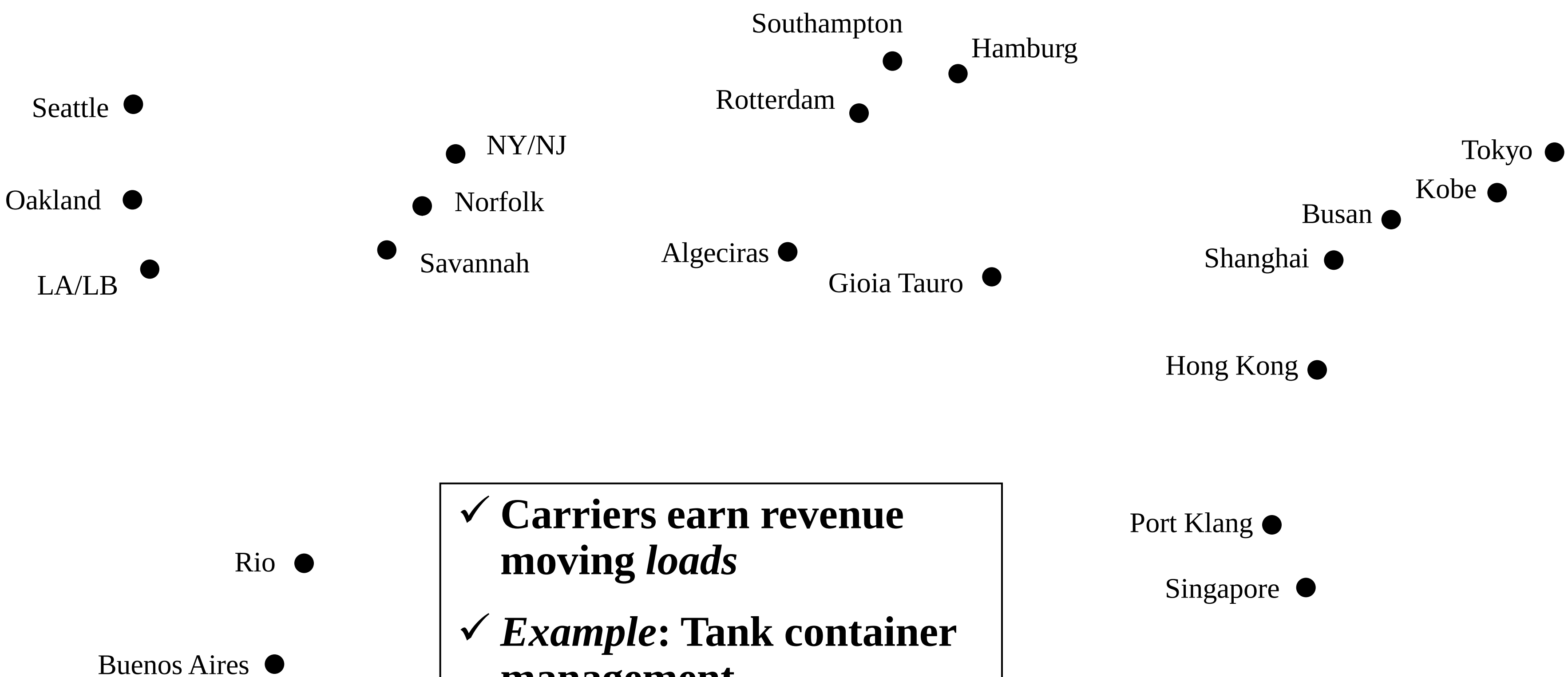
Rolling horizon two-stage approximation

- Specify a planning horizon of a number of time periods
- Partition planning horizon into two stages
  - Stage 1 periods: little to no uncertainty in parameters, and not modeled
  - Stage 2 periods: some parameters modeled with uncertainty
- Solve two-stage model, assuming that all uncertainty is revealed in the second stage
- Implement some decisions, roll horizon forward, and repeat

Two-stage robust repositioning problem

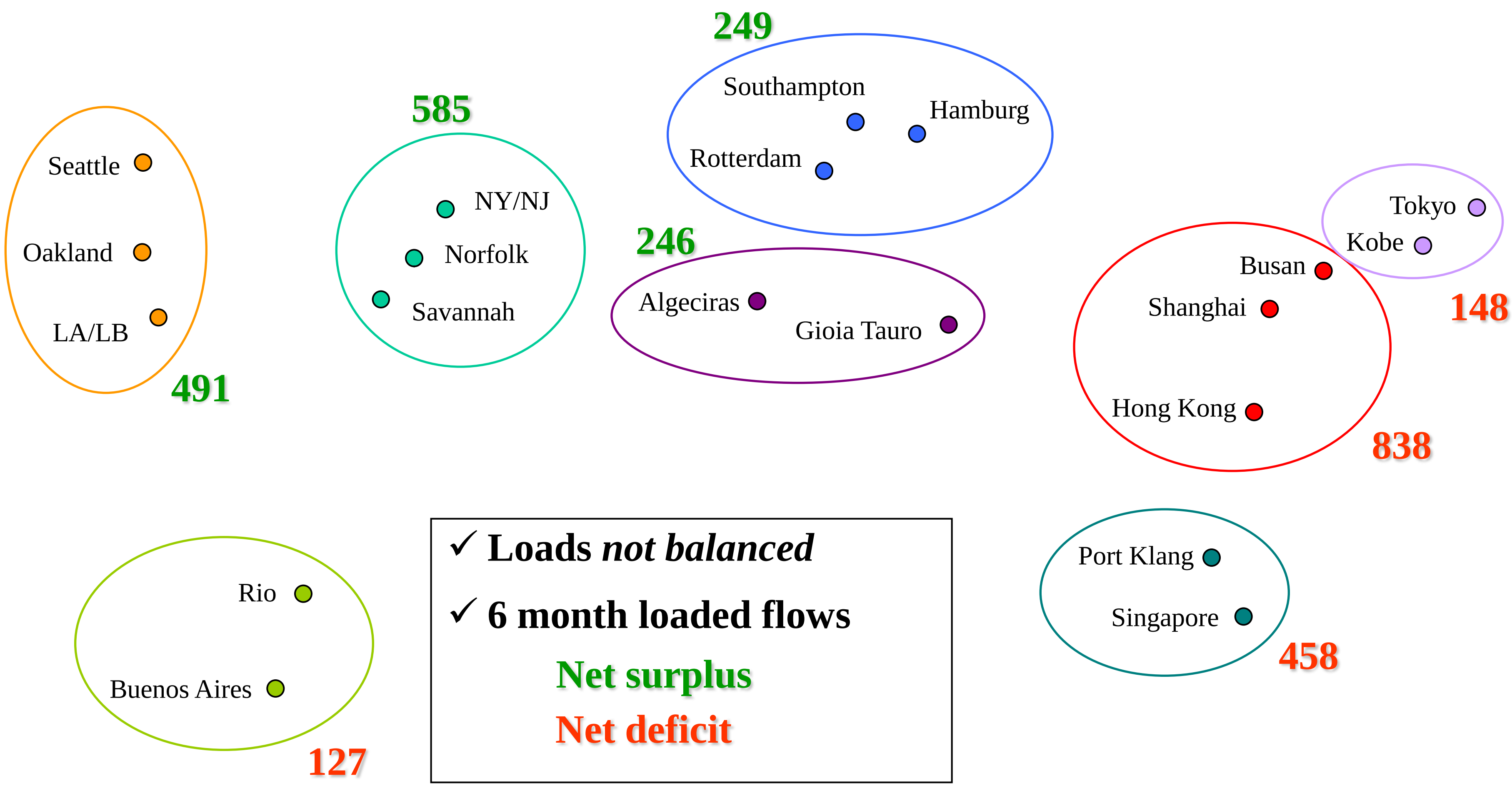
Erera, Morales, Savelsbergh (2009)

Repositioning in transportation



- ✓ **Carriers earn revenue moving *loads***
- ✓ ***Example*: Tank container management**

Repositioning in transportation



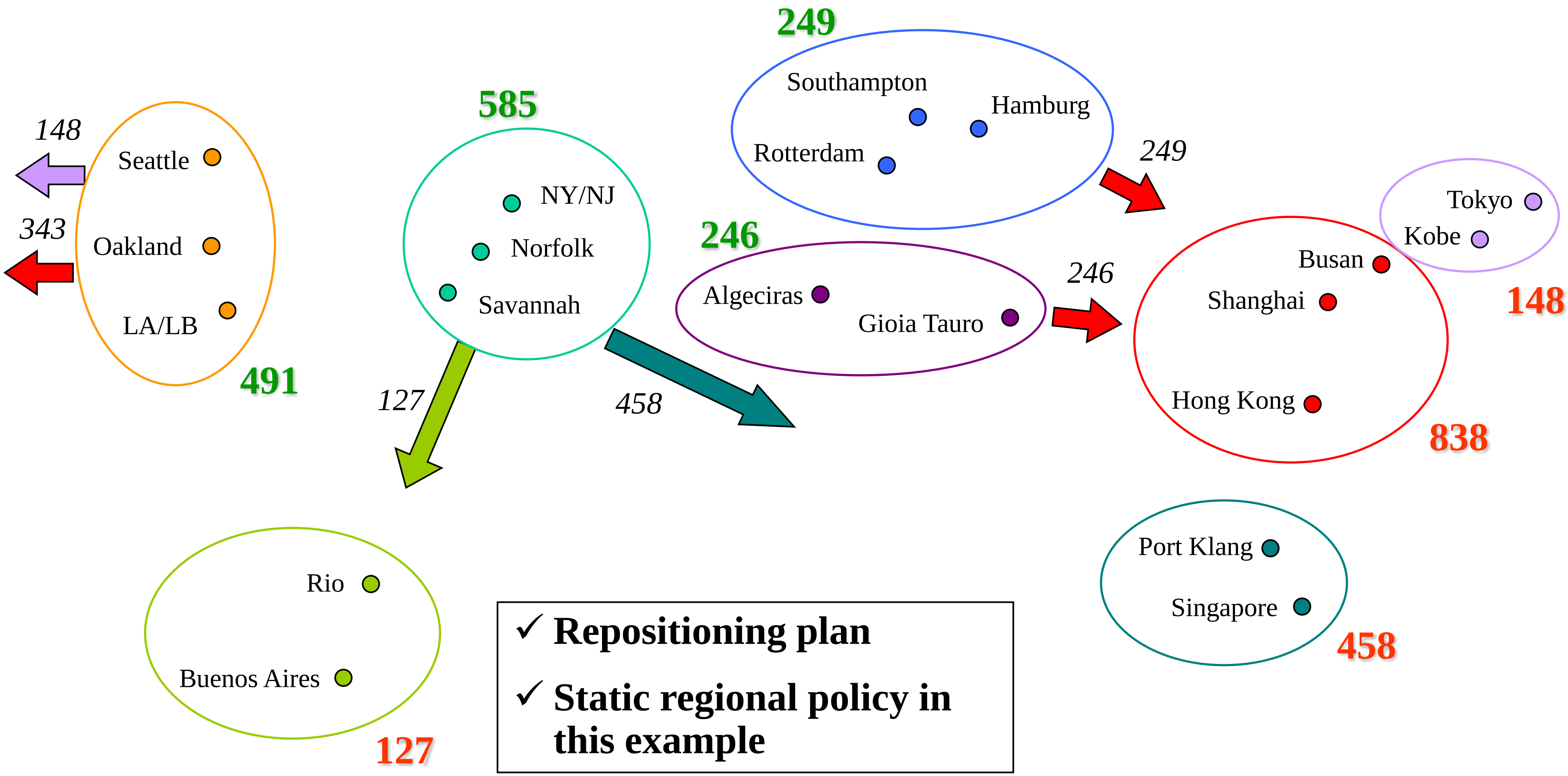
✓ **Loads *not balanced***

✓ **6 month loaded flows**

Net surplus

Net deficit

Repositioning in transportation



- ✓ **Repositioning plan**
- ✓ **Static regional policy in this example**

Repositioning in transportation

- System dynamics…
  - Short-term, long-term seasonality
  - Add new customers, lose old customers
  - Add new demand flows, lose old demand flows
- … and system uncertainties
  - Where and when will customer base change?
  - Where and when will demand flows change?
  - Business cycles

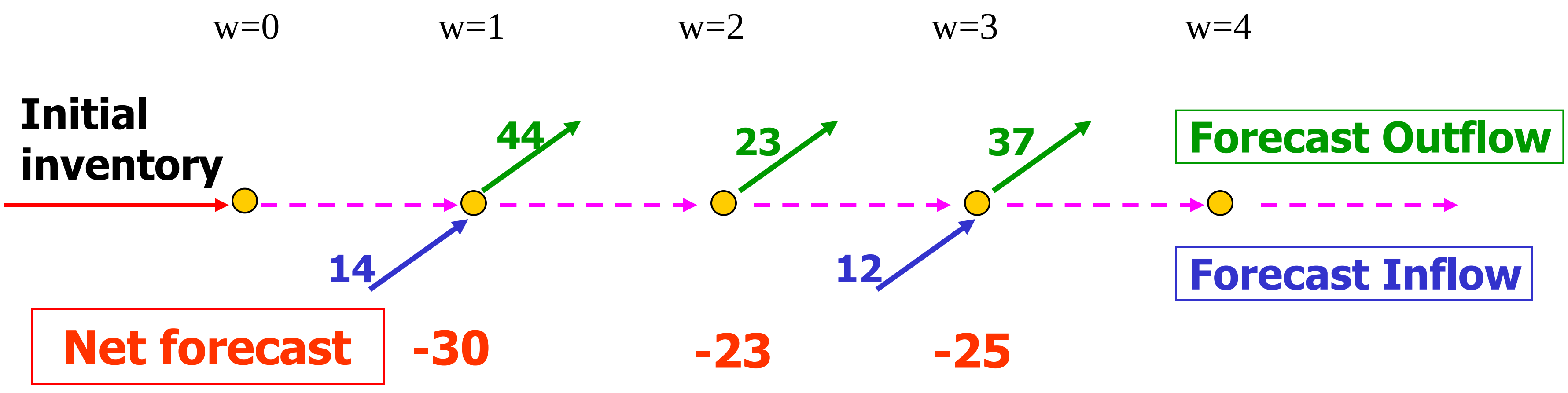
Repositioning in transportation

- Planning question
  - How should I move equipment this period?

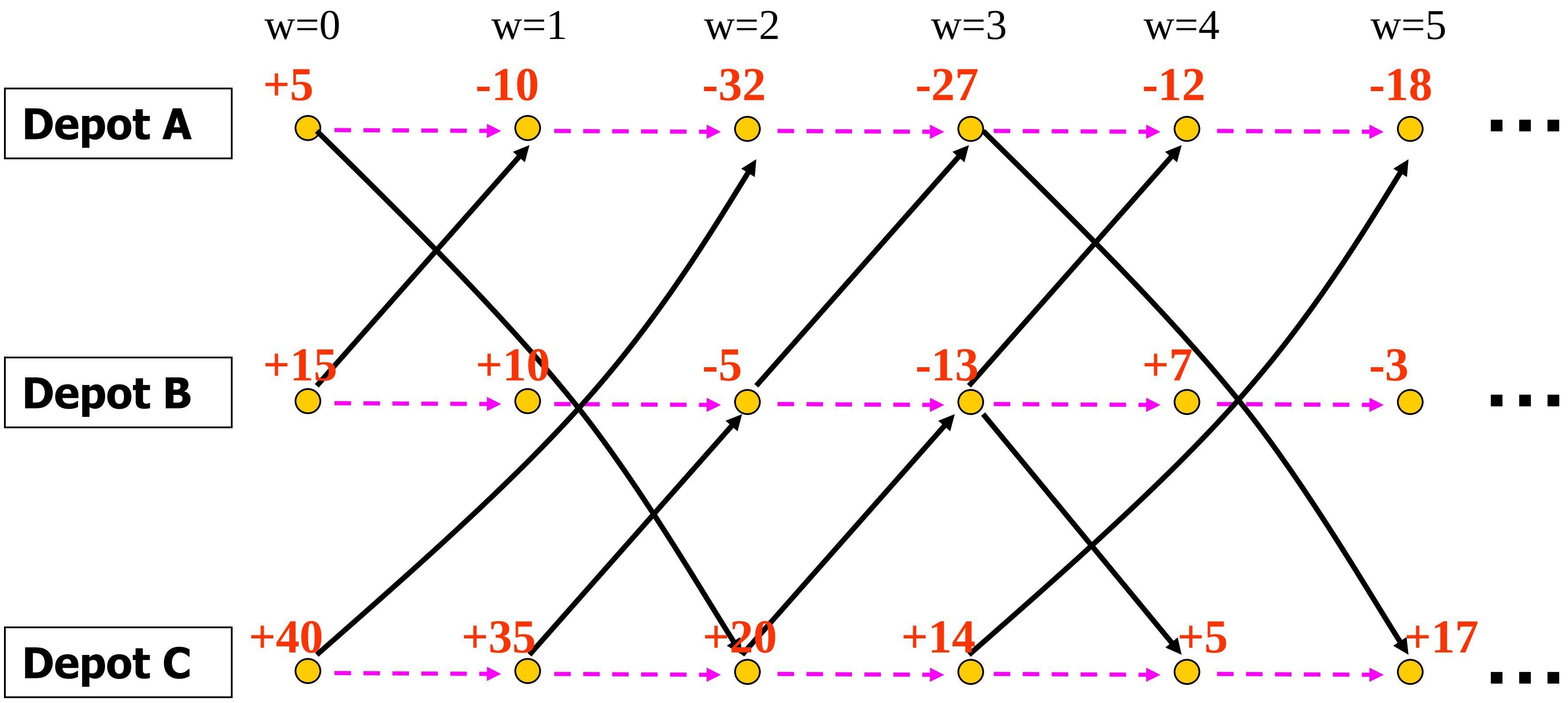
Dynamic repositioning practice

Weekly depot-to-depot via network flow

- *Initial* number of resources at each depot
- *Point forecasts* of the *net* supply of resources at each depot during each week



Network flow problem



$I =$ Inventory arcs

$R =$ Repositioning arcs

$b =$ Net estimated supply

Network flow problem

$$G = (N, A)$$

$$A = I \cup R$$

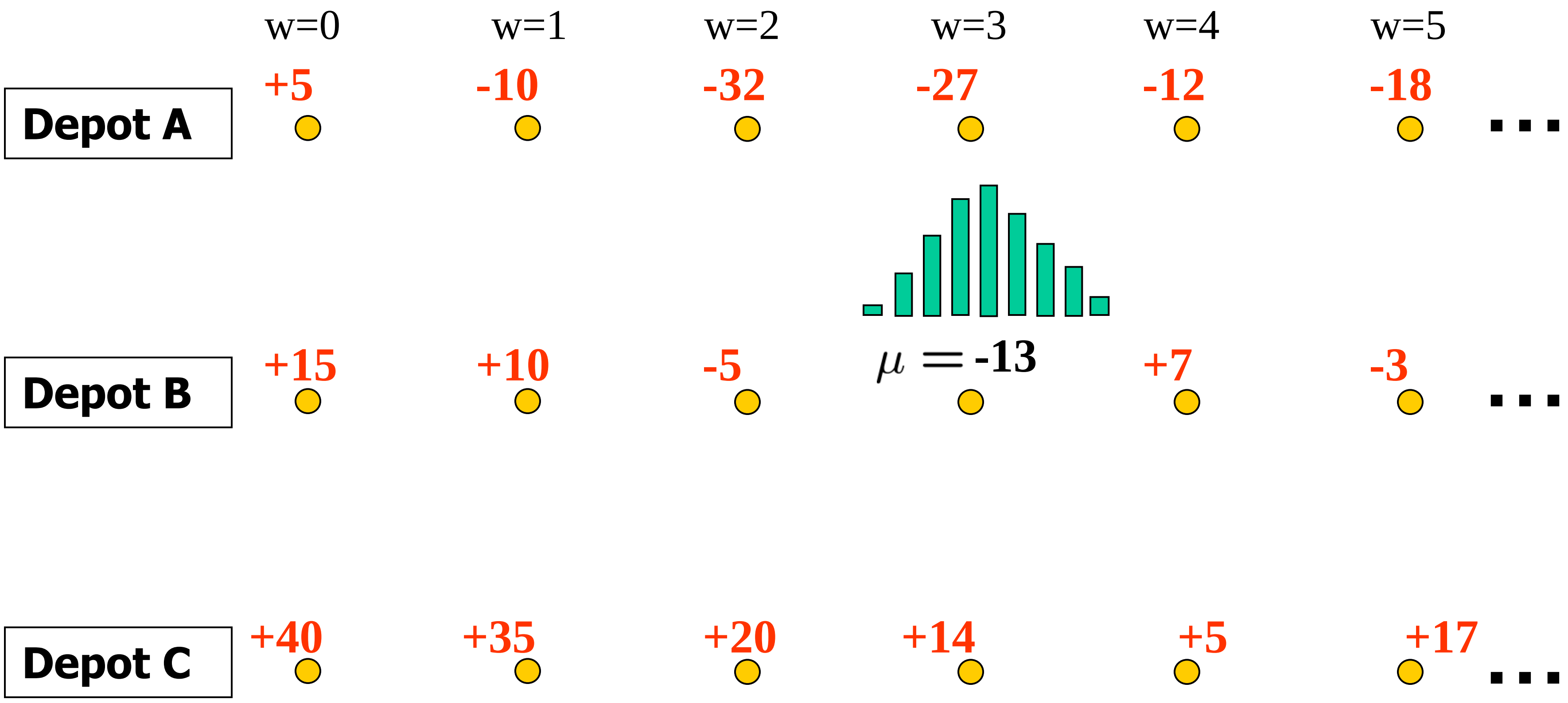
Nominal repositioning problem

NP $\quad \min_{x \in \mathbf{Z}^n} \{cx : \mathbf{A}x = b,\ x \geq 0\}$

$\mathbf{A} =$ node-arc incidence matrix

$b =$ point forecast supply vector, *nominal values*

Distribution forecasts



– Estimation difficulties
  - Blending known and unknown information
  - Signal may be evolving rapidly, or unpredictably

Alternative robust approach

- Expected value minimization limitations
  - Estimation of distribution forecasts
  - Risk-neutrality
  - Computability for large-scale problems

- Robust approach goals
  - Simpler input requirements
  - Computability with off-the-shelf optimization software
  - Focus on service
    - Ensure ability to serve future customer requirements
    - Parametric control of conservatism

Robust optimization

Ben-Tal, *et al*. (2004)

Adjustable robust counterpart

$$\textbf{ARC} \qquad \min_{x \in \mathbf{R}^n} \left\{ cx : \; \forall \, \tilde{b} \in \mathcal{Z} \;\; \exists \, y(\tilde{b}) : \mathbf{A}x + \mathbf{B}y(\tilde{b}) \leq \tilde{b} \right\}$$

Erera, *et al*. (2009)

Transformable robust problem

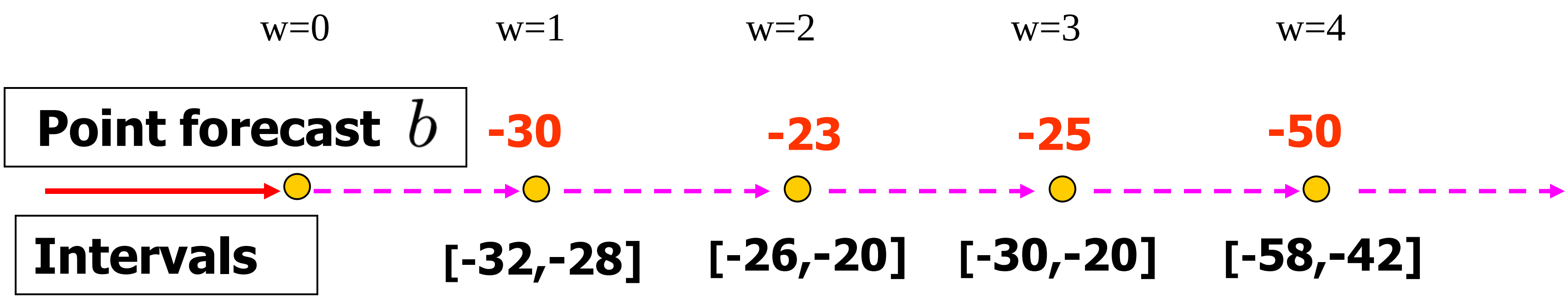
Related work

- Atamturk and Zhang (2007)
  - Two-stage network flow and design with uncertain demand
  - Complexity of separation problem
  - Tractable special cases
    - Lot-sizing problems
- Bertsimas and Sim (2003)
  - Robust network flow with uncertain costs

Robust repositioning framework

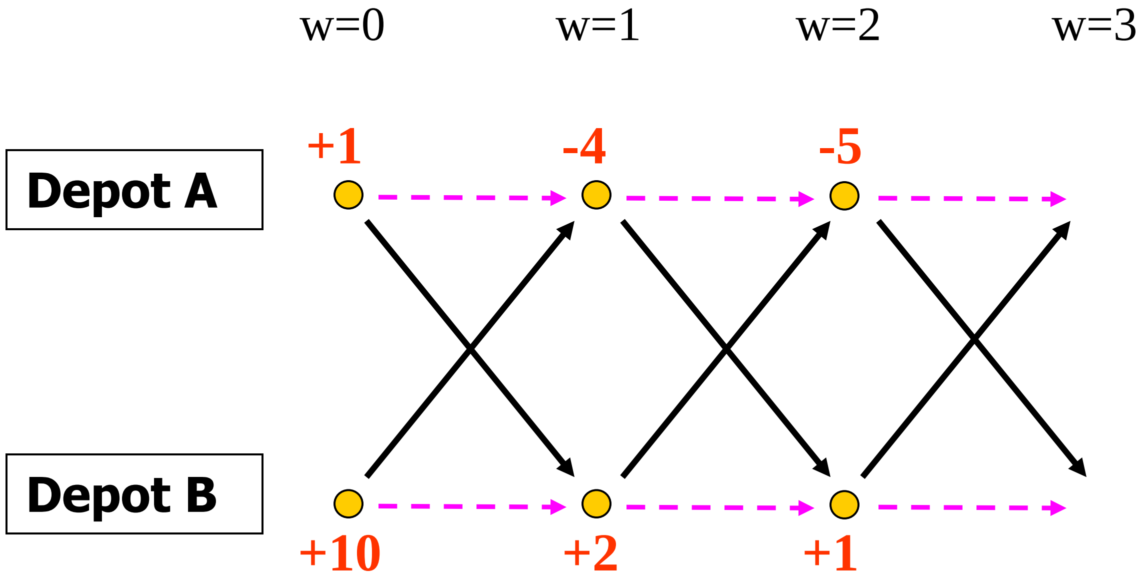
Symmetric interval forecasts

$$\tilde{b} \in [b - \bar{b}, b + \bar{b}] \quad \text{where} \quad \bar{b} \geq 0$$



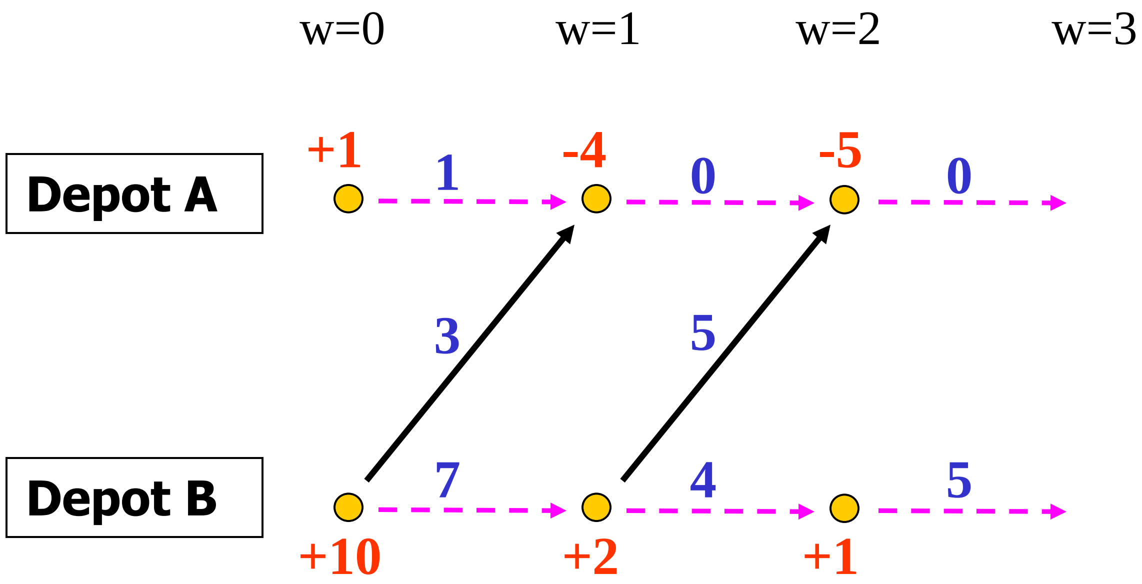
No distribution for $\tilde{b}$ assumed

Robust repositioning framework



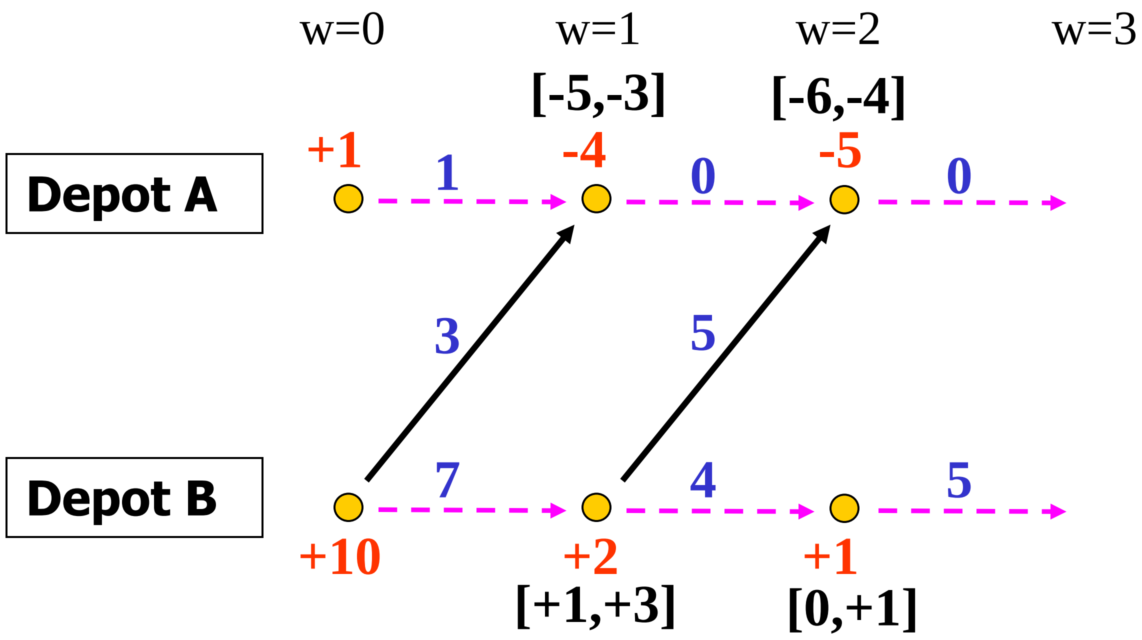
Nominal repositioning problem

Robust repositioning framework



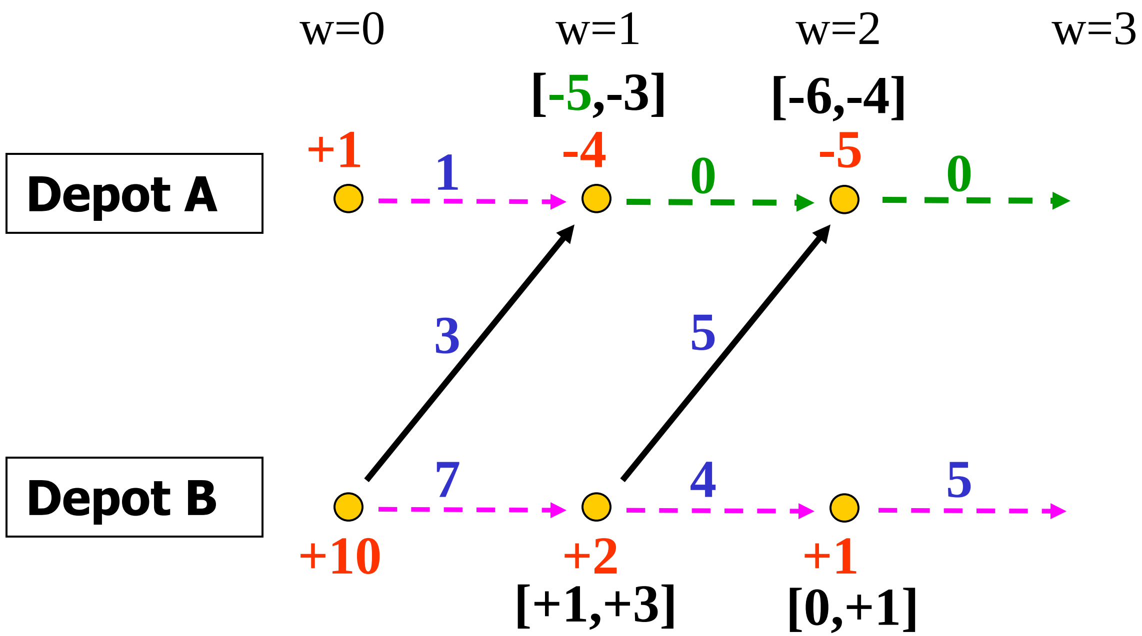
An optimal solution for the nominal problem

Robust repositioning framework



Problem with uncertainty intervals

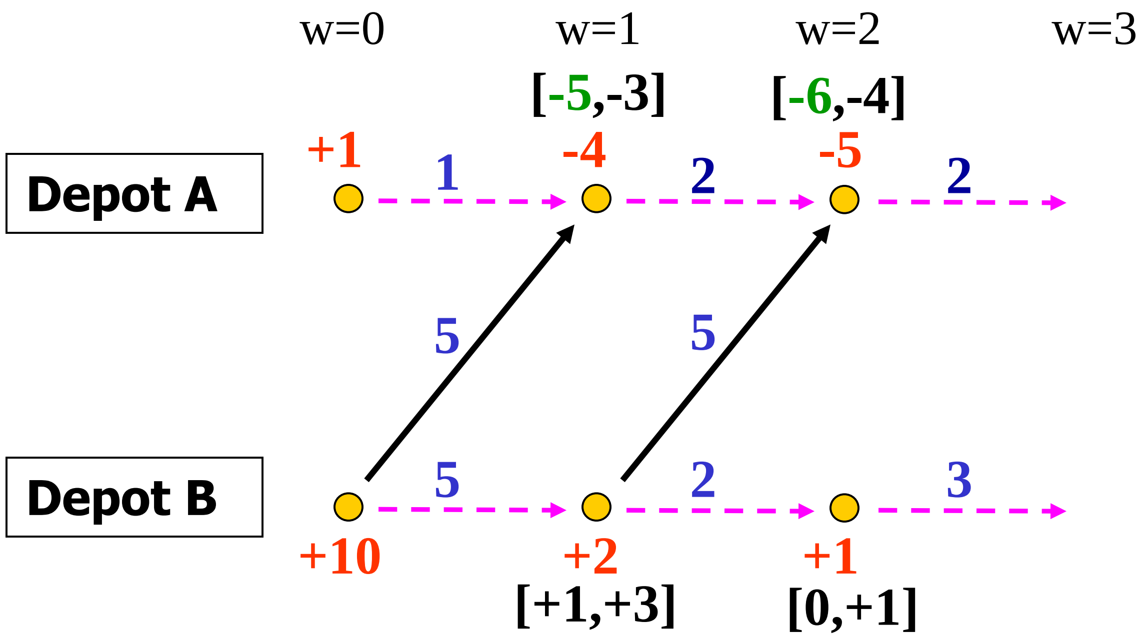
Robust repositioning framework



Problem with uncertainty intervals

Risk of a stock out

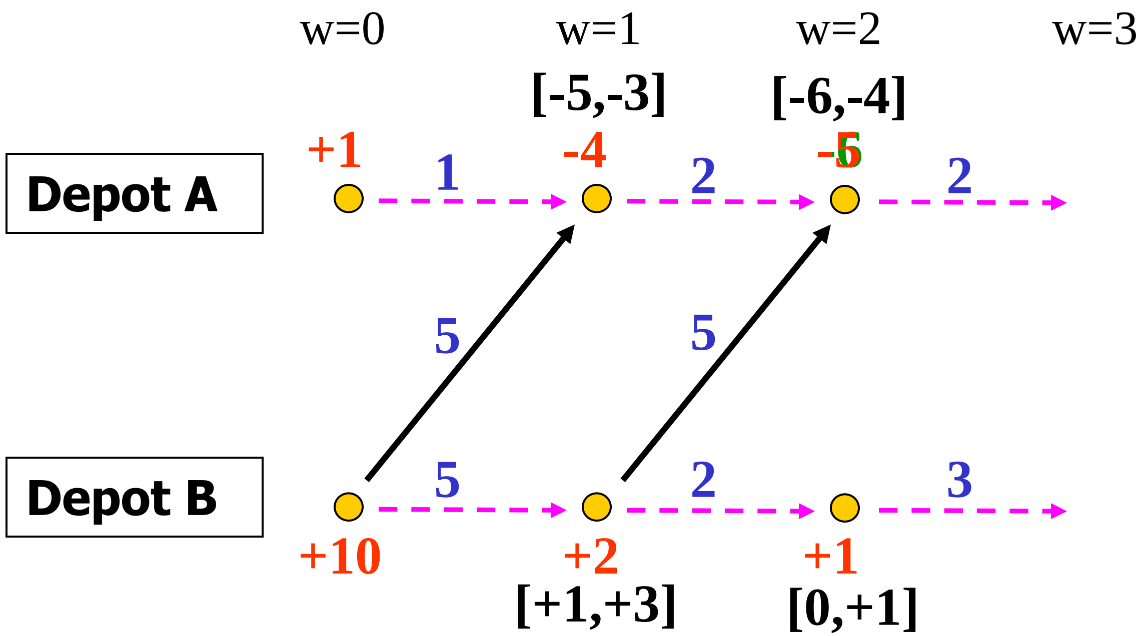
Robust repositioning framework



Alternative feasible solution for
the nominal problem

Higher nominal cost, but no stockout risk

Robust repositioning framework



Given realization, nominal solution *not feasible*

Allow variable *adjustments*
to recover feasibility

Robust repositioning framework

NP $\min_{x \in \mathbf{Z}^n} \{cx : \mathbf{A}x = b,\ x \geq 0\}$

Net supply realization: $b + \delta$

$$\delta \in \Gamma = \left\{\delta \in \mathbf{Z}^m : -\bar{b} \leq \delta \leq \bar{b}\right\}$$

Adjustable decisions: $y(\delta) \in W$

$$\begin{aligned} A(x + y(\delta)) &= b + \delta \\ x + y(\delta) &\geq 0 \end{aligned}$$

Allowable adjustments

$$y(\delta) \in W$$

- Inventory flow adjustments
  - Assume no capacity limitations
  - Homogeneous inventory carrying cost
- Local repositioning flow adjustments
  - Allow sharing between neighbors
  - Only allow flow increases

Robust repositioning framework

Transformable robust optimization problem

$\mathbf{RRP}(W, \Gamma)$:

$$\min_{x \in \mathbf{Z}^n} \quad cx$$

$$\begin{aligned} \mathbf{A}x &= b \\ x &\geq 0 \end{aligned}$$

$$\left.\begin{aligned} \mathbf{A}(x + y(\delta)) &= b + \delta \\ x + y(\delta) &\geq 0 \\ y(\delta) &\in W \end{aligned}\right\} \quad \forall\, \delta \in \Gamma$$

Robust repositioning framework

Transformable robust optimization problem

$\textbf{RRP}(W, \Gamma)$:

$$\min_{x \in \mathbf{Z}^n} \quad cx$$

$$\mathbf{A}x = b$$

$$x \geq 0$$

$$x \in H(W, \delta) \qquad \forall\ \delta \in \Gamma$$

Solving $\textsf{RRP}(W, \varphi_k)$

- Develop sufficient constraint sets that guarantee $x \in H(W, \delta)$ for all $\delta \in \varphi_k$

- Constraint sets vary given feasible adjustments: W

Parametric conservatism

$$\Gamma = \left\{ \delta \in \mathbf{Z}^{|N|} : -\bar{b} \leq \delta \leq \bar{b} \right\}$$

A robust solution with respect to Γ may be *too conservative*

$$\varphi_k = \left\{ \delta \in \Gamma : \delta_v = z_v \bar{b}_v, |z_v| \leq 1, \sum_{v \in N} |z_v| \leq k \right\}$$

At most *k* time-space demands can *simultaneously* realize worst case value

Inventory-only problem

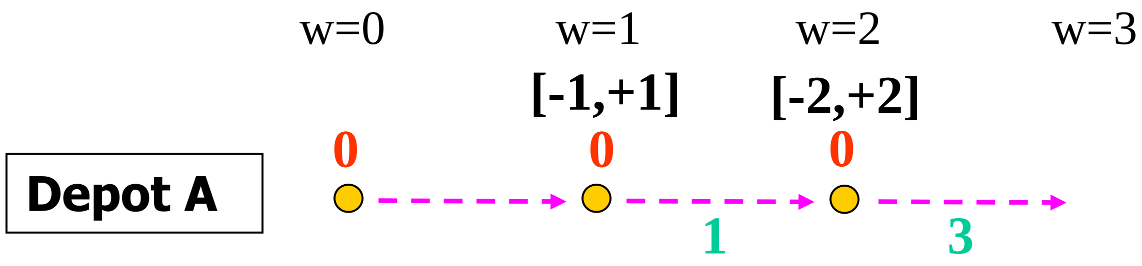
$\textbf{RRP}(W_1, \varphi_k)$

where W_1 restricts adjustable variables s.t.

$$y_a = 0 \quad \forall\, a \in R$$

Nominal flows on repositioning arcs cannot be adjusted, therefore each depot must hedge against uncertainty with only its own inventory

Inventory-only problem

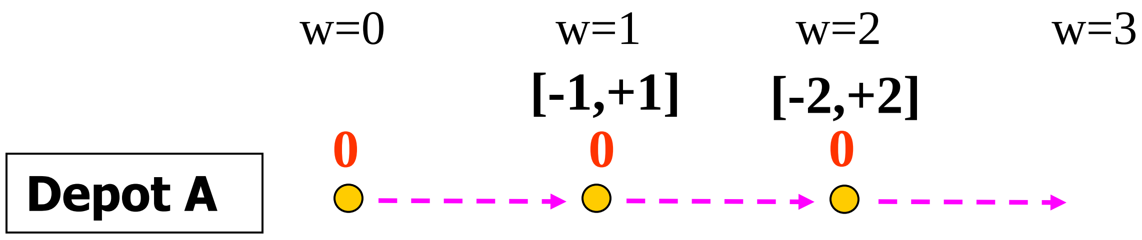


Maximum vulnerability $\vartheta(a)$

$$N\left(a = (v_t^j, v_{t+1}^j)\right) = \left\{v_\ell^j : \ell \leq t\right\}$$

$$\vartheta(a) = \sum_{v \in N_a} \bar{b}_v$$

Inventory-only problem

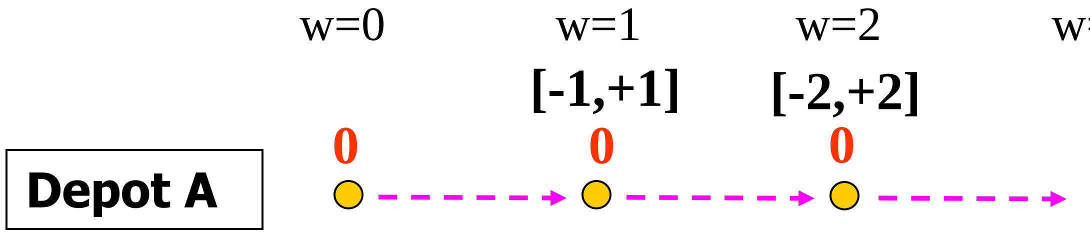


Bounded vulnerability $\upsilon(N,k)$

Simple knapsack problem

$$\upsilon(N,k) = \sum (k \text{ largest } \overline{b_v},\ v \in N)$$

Inventory-only problem



$$\upsilon(\cdot,1)=1 \qquad \upsilon(\cdot,1)=2$$

$$\upsilon(\cdot,2)=1 \qquad \upsilon(\cdot,2)=3$$

Inventory-only problem

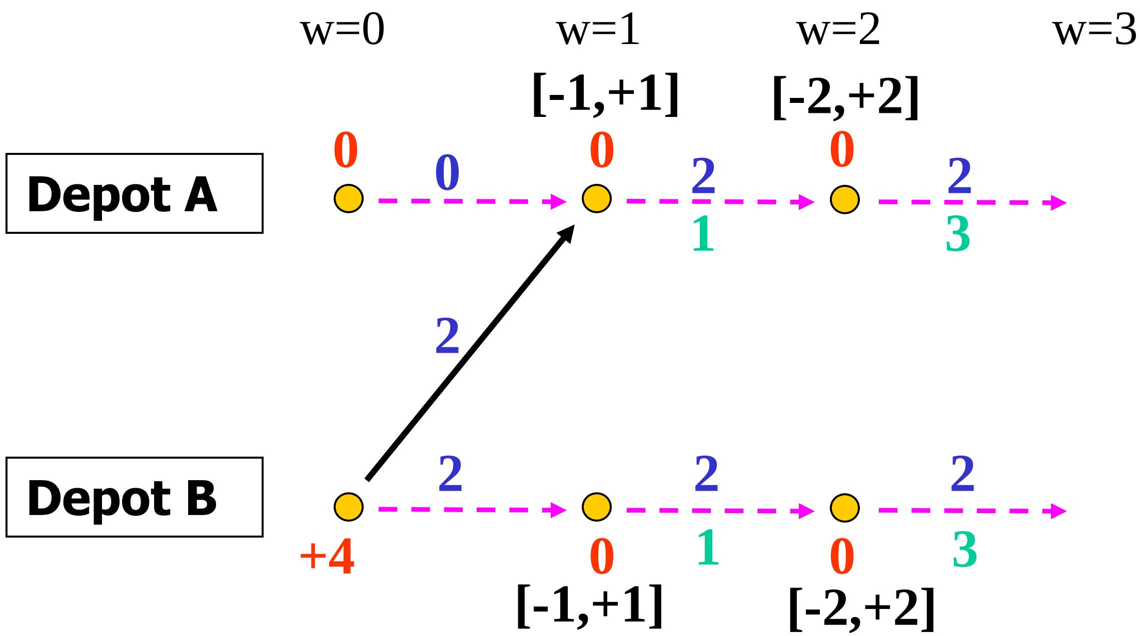
Theorem

A feasible solution x of the nominal problem is *k*-robust inventory feasible *if and only if* for all inventory arcs *a*

$$x_a \geq \upsilon(N_a, k)$$

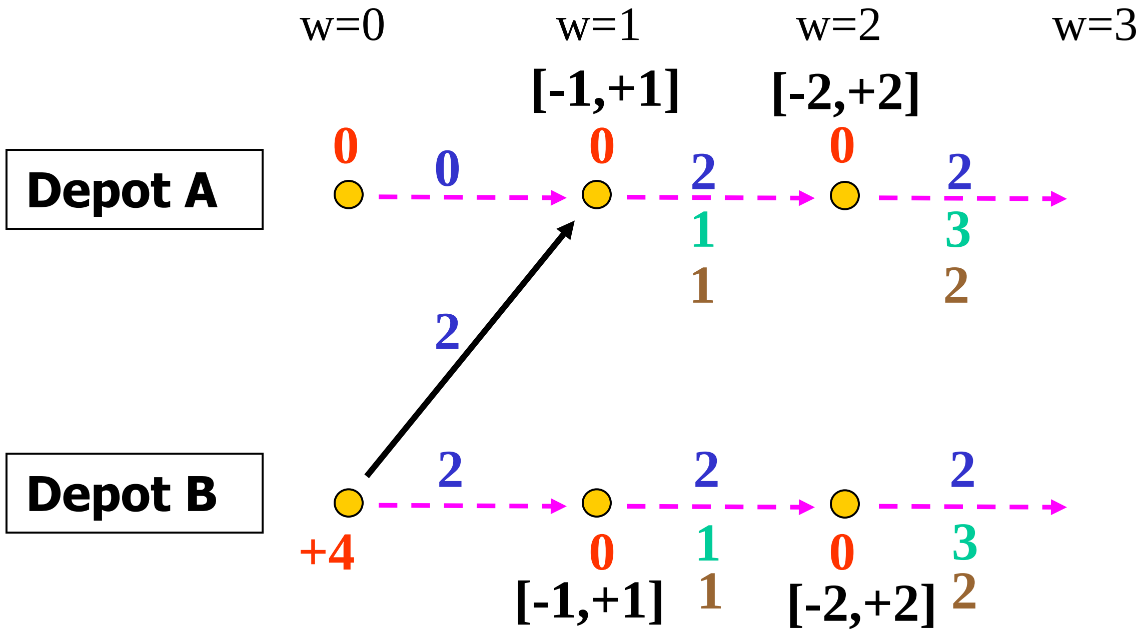
Solvable in polynomial time by adding pre-computed lower-bounds on inventory arcs to the nominal problem

Inventory-only problem



Maximum vulnerabilities $\vartheta(v_t^j)$

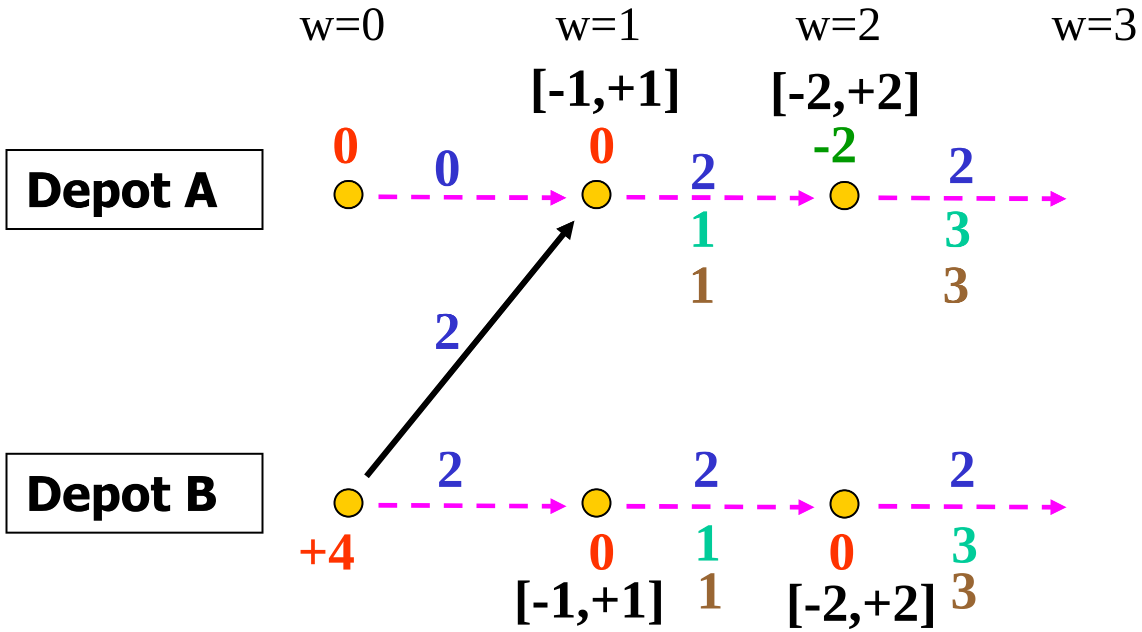
k-robust inventory example



Bounded vulnerabilities $\upsilon(N_a, 1)$

The solution is 1-robust

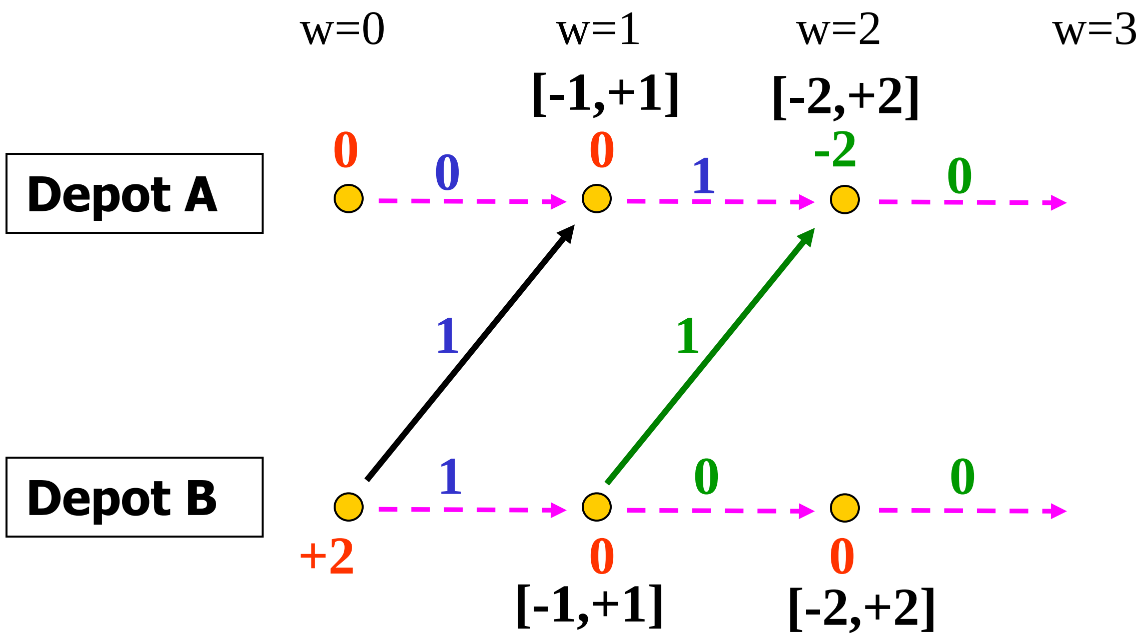
k-robust inventory example



Bounded vulnerabilities $\upsilon(N_a, 2)$

The solution is not 2-robust

Inventory pooling



If we allow depot A and B to reactively reposition containers between them, the solution is 2-robust

ROP for reactive repositioning

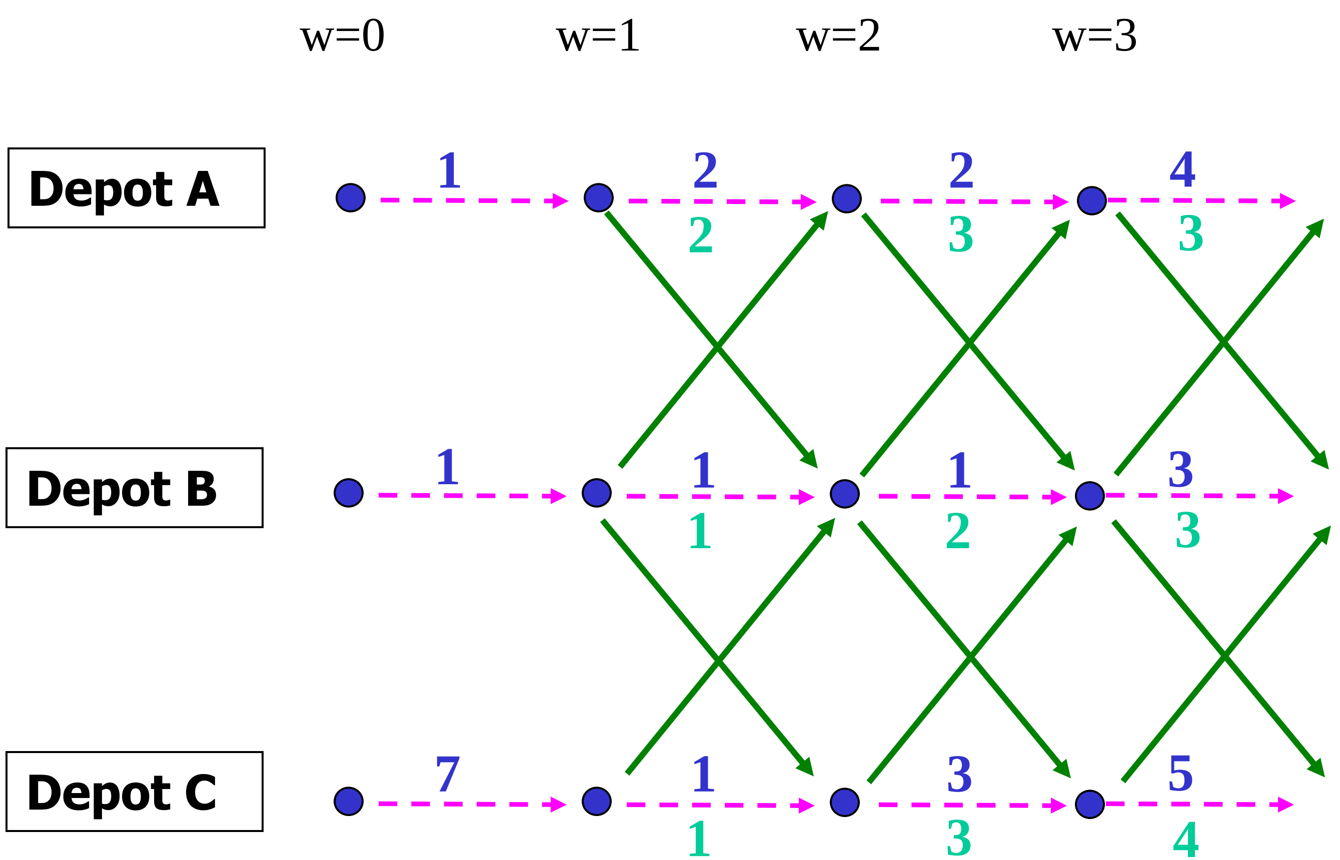
$\textbf{ROP}(W_2, \varphi_k)$:

$$\min_x \{cx : Ax = b,\ x \in H(W_2, \delta)\ \forall\ \delta \in \varphi_k\}$$

W_2 : [Non-negative changes to repositioning arcs]
[Any integer change to inventory arcs]
[No changes to first period repo arcs]

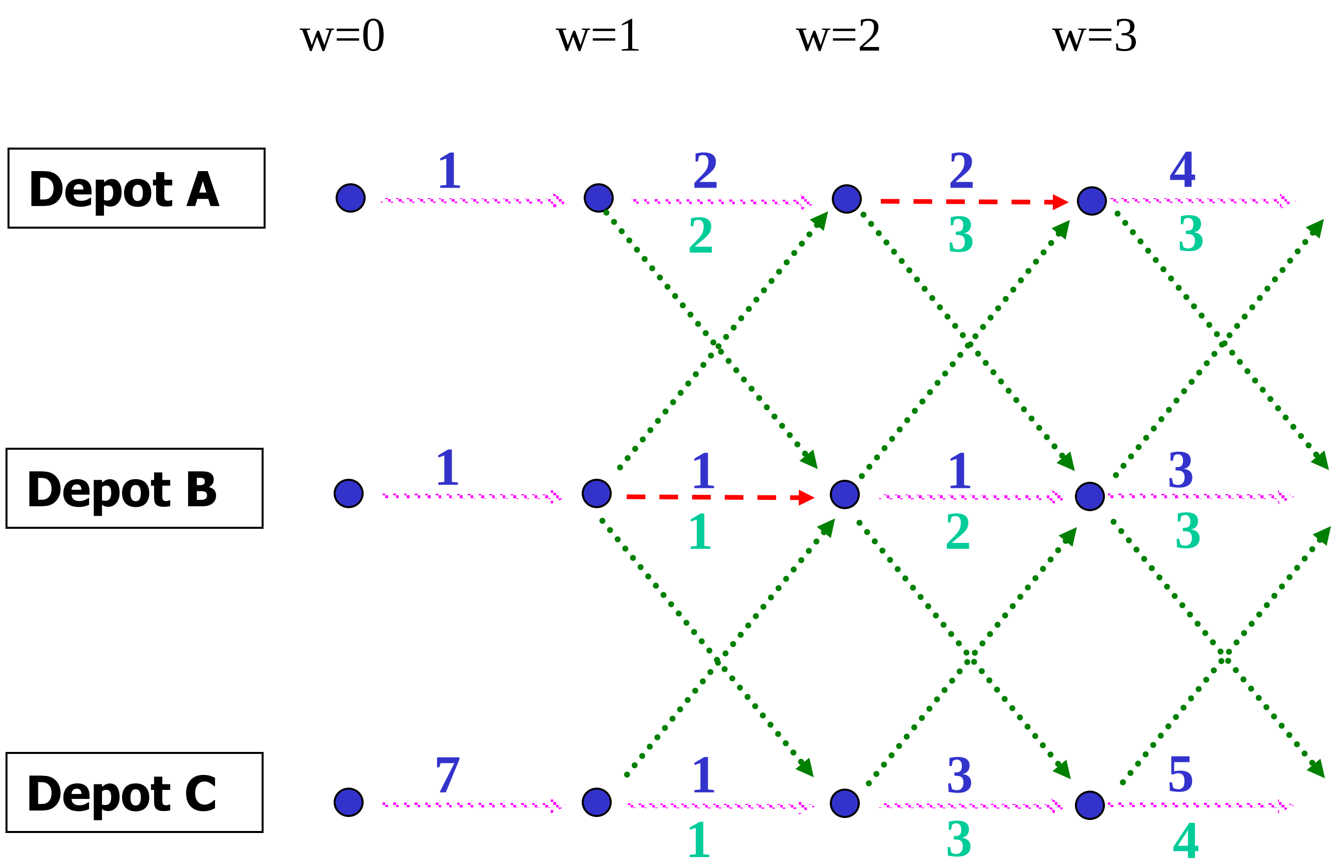
Feasibility for reactive repositioning

Depots supply/receive



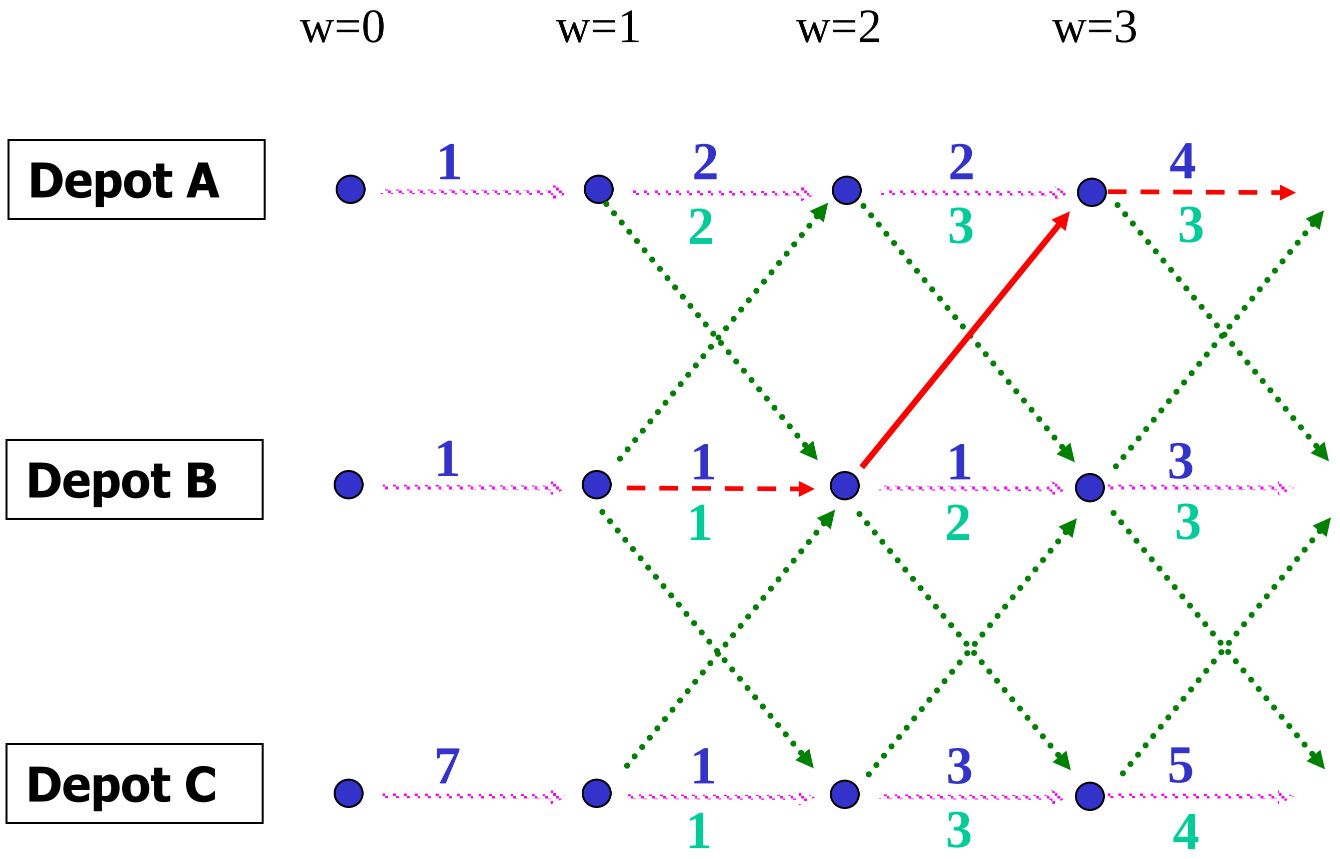
Feasibility for reactive repositioning

Competing arcs: no reactive flow path



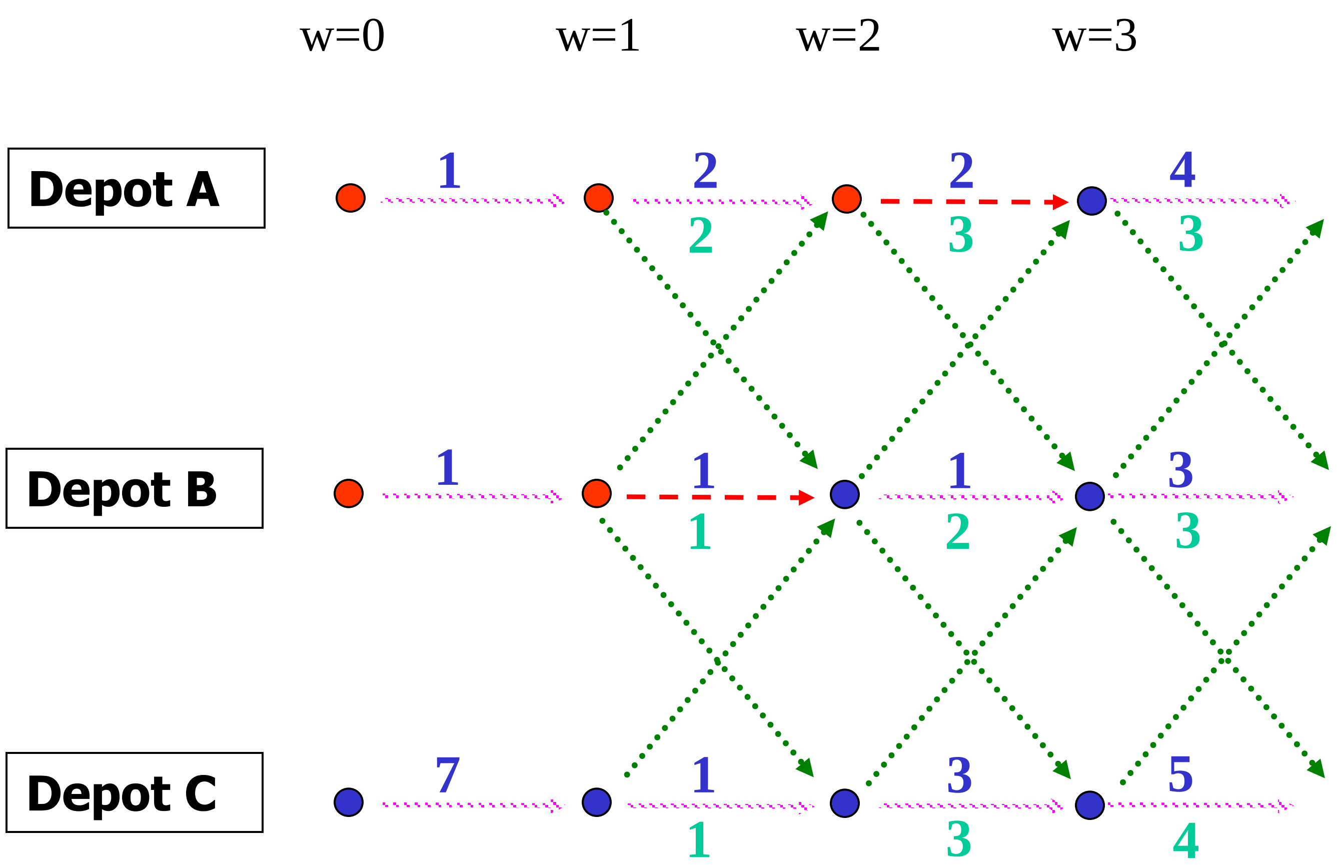
Feasibility for reactive repositioning

Not competing



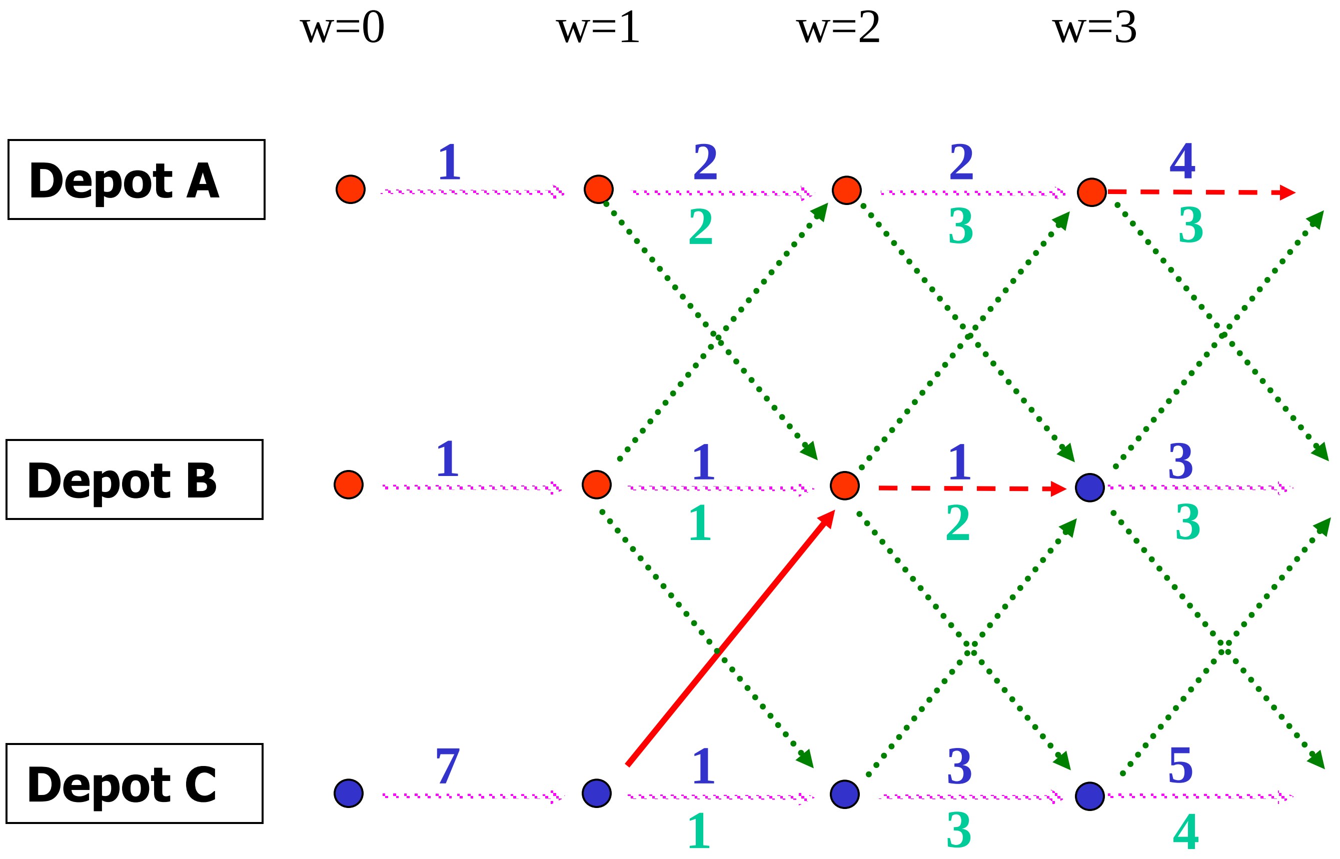
Feasibility for reactive repositioning

Inbound-closed nodes: no inbound reactive arcs



Feasibility for reactive repositioning

Not inbound-closed nodes

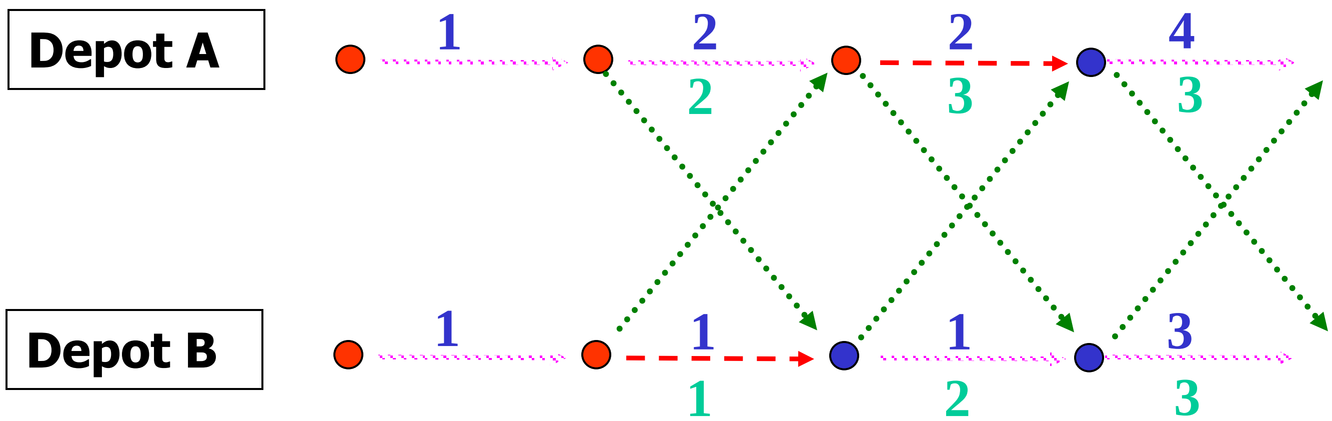


Feasibility for reactive repositioning

Theorem

A solution x to the nominal problem is feasible for the reactive repositioning robust repositioning problem if and only if for every set of competing arcs K defining an inbound closed node set U:

$$\sum_{a \in K} x_a \geq \upsilon(U, k)$$



Arc set satisfies conditions for all $k \leq 3$

Feasibility for reactive repositioning

Theorem

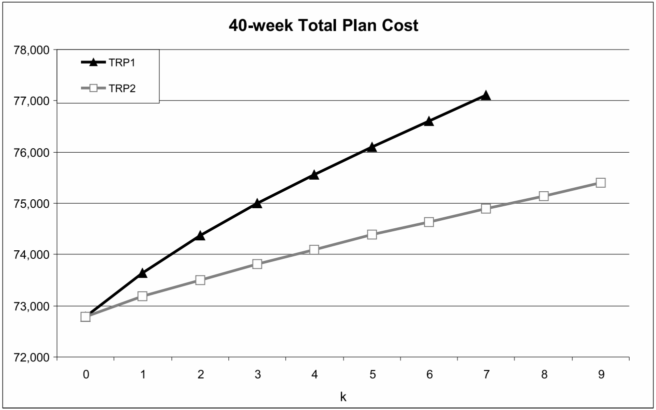
A solution x to the nominal problem is feasible for the reactive repositioning robust repositioning problem if and only if for every set of competing arcs K defining an inbound closed node set U:

$$\sum_{a \in K} x_a \geq \upsilon(U, k)$$

- Potentially large number of constraints
- Resulting formulation requires IP (not LP)
- **Constraint set size independent of uncertain outcome space size!!**

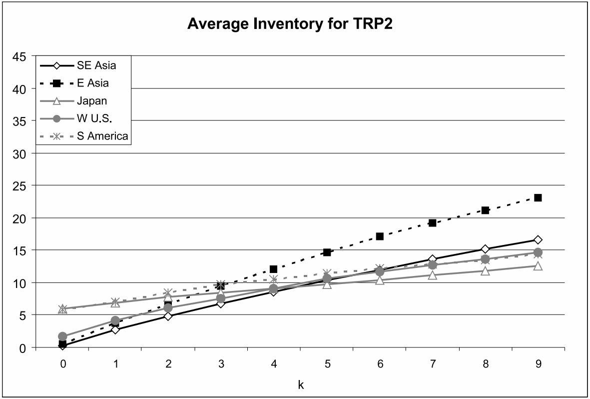
Return…

Robust repositioning results



20 depots, 8 regions

Robust repositioning results



Inventory builds to hedge against uncertainty

What to remember

1. Stochastic and robust optimization are for *dynamic* decision planning problems

What to remember

1. Stochastic and robust optimization are for *dynamic* decision planning problems
2. Many ways to effectively incorporate parameter uncertainty in logistics optimization

What to remember

1. Stochastic and robust optimization are for *dynamic* decision planning problems
2. Many ways to effectively incorporate parameter uncertainty in logistics optimization
3. Modeling and treatment of *recourse* especially critical

What to remember

1. Stochastic and robust optimization are for *dynamic* decision planning problems
2. Many ways to effectively incorporate parameter uncertainty in logistics optimization
3. Modeling and treatment of *recourse* especially critical
4. Ensure that your model is useful (and interesting), then solve

References

1. A. Ak and A.L. Erera, "A Paired-Vehicle Recourse Strategy for the Vehicle Routing Problem with Stochastic Demands," Transportation Science, 2007.
2. A.L Erera, J.C. Morales, and M.W.P. Savelsbergh, "Robust Optimization for Empty Repositioning Problems," Operations Research, 2009.
3. A.L. Erera, M.W.P. Savelsbergh, and E. Uyar, "Fixed Routes with Backup Vehicles for Stochastic Vehicle Routing Problems with Time Constraints," Networks, 2009.
4. A.L Erera, J.C. Morales, and M.W.P. Savelsbergh, "The Vehicle Routing Problem with Stochastic Demands and Duration Constraints," to appear, Transportation Science, 2010.