Models for Designing Resilient Supply Chain Networks for Chemicals and Gases

Lawrence V. Snyder Peng Peng

Department of Industrial & Systems Engineering
Center for Value Chain Research
Lehigh University
Bethlehem, PA

EWO Meeting March 12, 2008

Outline

Outline

1. Introduction
2. The Model
3. Example

Introduction

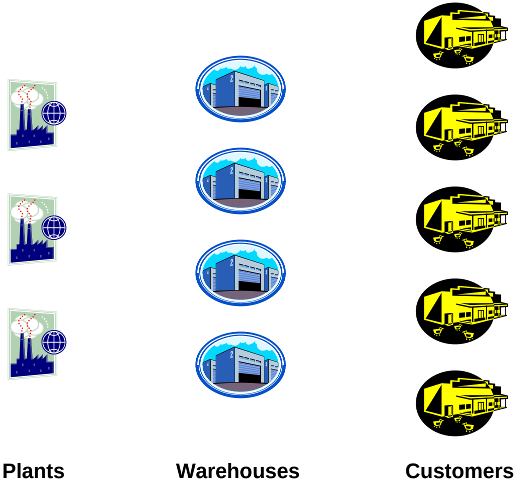
- Supply chain networks are usually designed as though they funcion in normal mode all the time
- But disruptions are a significant factor
- Result in significant cost increases
  - Increased transportation costs
  - Penalty for missed demands

Introduction

- Supply chain networks are usually designed as though they funcion in normal mode all the time
- But disruptions are a significant factor
- Result in significant cost increases
  - Increased transportation costs
  - Penalty for missed demands

- **Objective:** Develop model for designing supply chains that perform well when no disruptions occur but not too badly when disruptions occur.
- Focus on packaged gases, but model is generic

Motivating Example



Decisions

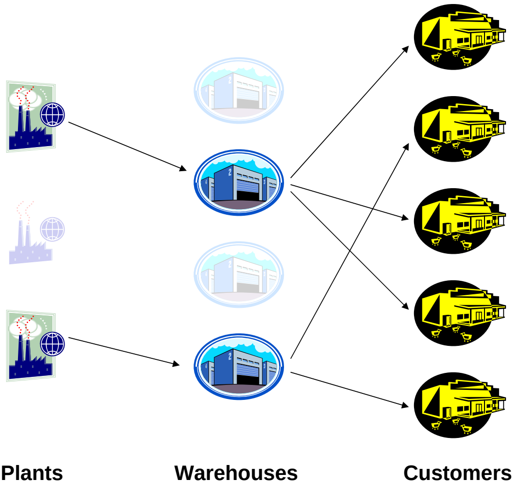
- Need to decide which plants and warehouses to open
- Customer locations are fixed
- Minimize sum of
  - Fixed cost (to build/lease facilities)
  - Transportation cost

Decisions

- Need to decide which plants and warehouses to open
- Customer locations are fixed
- Minimize sum of
  - Fixed cost (to build/lease facilities)
  - Transportation cost
- Suppose we optimize **ignoring disruptions**

"Optimal" Solution

Cost: $521,163

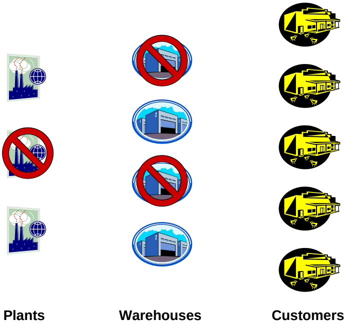


Disruptions

- Now suppose disruptions can occur
- We must design supply chain before we know what facilities will be disrupted
- Possible disruptions are described by **scenarios**

Disruptions

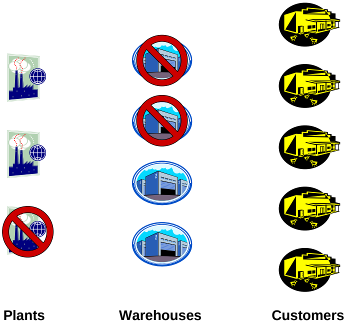
- Now suppose disruptions can occur
- We must design supply chain before we know what facilities will be disrupted
- Possible disruptions are described by **scenarios**



Scenario 1

Disruptions

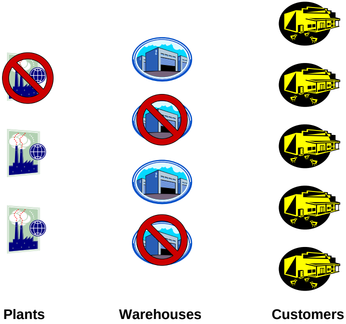
- Now suppose disruptions can occur
- We must design supply chain before we know what facilities will be disrupted
- Possible disruptions are described by **scenarios**



Scenario 2

Disruptions

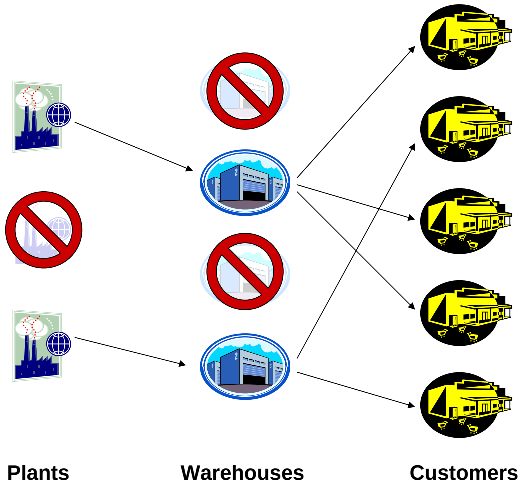
- Now suppose disruptions can occur
- We must design supply chain before we know what facilities will be disrupted
- Possible disruptions are described by **scenarios**



Scenario 3

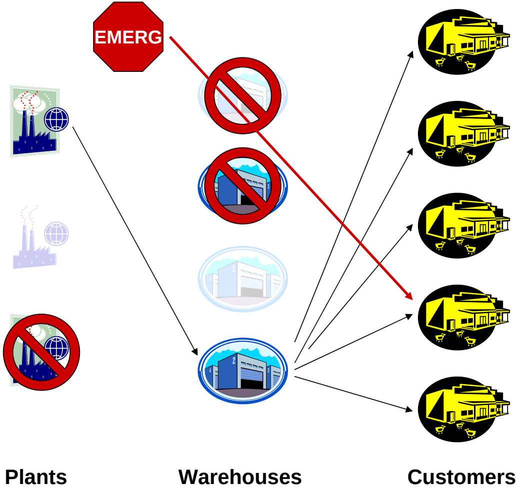
Solution Performance When Disruptions Occur

Scenario 1: Cost $521,163



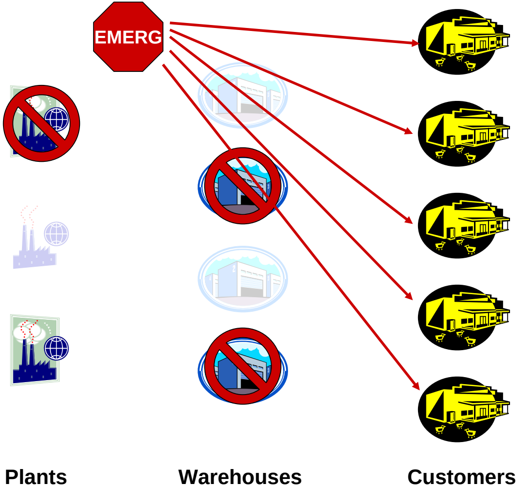
Solution Performance When Disruptions Occur

Scenario 2: Cost $982,241

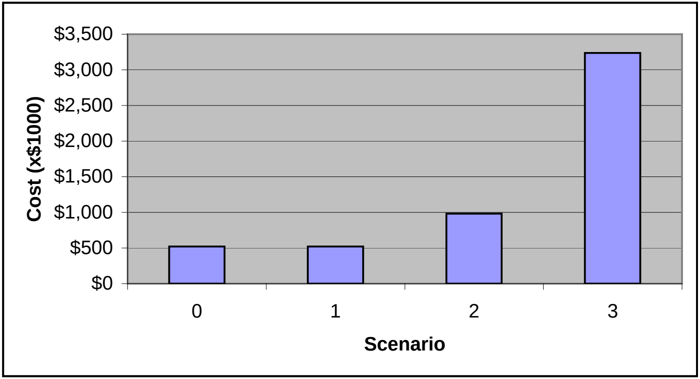


Solution Performance When Disruptions Occur

Scenario 3: Cost $3,237,107



Solution Performance: Summary



Outline

Model Objectives

- Can we improve performance under the disruption scenarios by choosing better facilities?
- "Nominal" cost will increase
- But scenario costs will decrease

Model Objectives

- Can we improve performance under the disruption scenarios by choosing better facilities?
- "Nominal" cost will increase
- But scenario costs will decrease
- Two-stage problem:
  - **Stage 1:** Choose facility locations
  - (scenario occurs)
  - **Stage 2:** Assign flows

Robust Optimization

- This is a **robust optimization** problem
- Prevent performance from being "too bad" in any scenario
- Lots of approaches for robust optimization in the literature
  - Min-max cost
  - Min-max regret
  - CVaR
  - etc.

Robust Optimization

- This is a **robust optimization** problem
- Prevent performance from being “too bad” in any scenario
- Lots of approaches for robust optimization in the literature
  - Min-max cost
  - Min-max regret
  - CVaR
  - etc.
- Our approach:
  - Minimize cost in nominal scenario (no disruptions)
  - Subject to bound on cost in any other scenario
  - **“p-robustness”**

Notation: Nodes and Arcs

- General network (V, A)
  - V = set of nodes
  - A = set of arcs
  - Not necesssarily plants, warehouses, customers
- $V_0 \subseteq V$ = set of non-demand nodes

Notation: Nodes and Arcs

- General network (V, A)
  - V = set of nodes
  - A = set of arcs
  - Not necesssarily plants, warehouses, customers
- $V_0 \subseteq V$ = set of non-demand nodes
- N = number of products
- b_{jn} = supply of product n at node $j \in V$
  - > 0 for supply nodes
  - < 0 for demand nodes
  - $= 0$ for transshipment nodes
- k_{jn} = capacity for product n at node $j \in V_0$
  - Depends on type of product and type of storage (bulk, drum, tote)

Notation: Costs and Disruptions

- f_j = fixed cost to open node $j \in V_0$
- d_{ijn} = cost to transport one unit of product n on arc $(i, j) \in A$

Notation: Costs and Disruptions

- f_j = fixed cost to open node $j \in V_0$
- d_{ijn} = cost to transport one unit of product n on arc $(i, j) \in A$
- S = set of scenarios
  - $s = 0$ is nominal scenario
- a_{js} = 1 if node $j \in V_0$ is disrupted in scenario s
  - Applies to non-demand nodes only
- U_s = maximum allowable cost in scenario $s \in S \setminus \{0\}$

Notation: Decision Variables

- $X_j = 1$ if node $j \in V_0$ is open, 0 otherwise
- Y_{ijns} = flow of product n on arc (i, j) in scenario s

Objective Function

Minimize nominal-scenario (no-disruption) cost:

$$\text{minimize} \quad \sum_{j\in V_0} f_j X_j + \sum_{(i,j)\in A} \sum_{n=1}^{N} d_{ijn} Y_{ijn0} \tag{1}$$

Constraints

Subject to:

$$\sum_{j \in V_0} f_j X_j + \sum_{(i,j) \in A} \sum_{n=1}^{N} d_{ijn} Y_{ijns} \leq U_s \qquad \forall s \tag{2}$$

$$\sum_{(j,i) \in A} Y_{jins} - \sum_{(i,j) \in A} Y_{ijns} = b_{jn} \qquad \forall j, n, s \tag{3}$$

$$\sum_{(j,i) \in A} Y_{jins} \leq (1 - a_{js}) k_{jn} X_j \qquad \forall j, n, s \tag{4}$$

$$X_j \in \{0, 1\} \qquad \forall j \tag{5}$$

$$Y_{ijns} \geq 0 \qquad \forall i, j, n, s \tag{6}$$

In Words

minimize cost in nominal scenario

subject to cost in scenario $s \leq U_s \quad \forall s$

flow balance constraints $\quad \forall j, n, s$

flow can't exceed capacity, or 0 if disrupted $\quad \forall j, n, s$

X_j integer

Y_{ijns} non-negative

An Even More Compact Description

$$\begin{aligned} \text{minimize } & c_0(X, Y) \\ \text{subject to } & c_s(X, Y) \le U_s \qquad \forall s \\ & (X, Y) \in \Omega \end{aligned}$$

Solution Methods

- For now, we solve using an off-the-shelf MIP solver
- Run times generally $<$5 minutes for problems with
  - 40 plants
  - 40 warehouses
  - 500 customers
  - 5 scenarios
  - 1 product
- For bigger problems, we'll need customized algorithms
  - Future research

Outline

Example

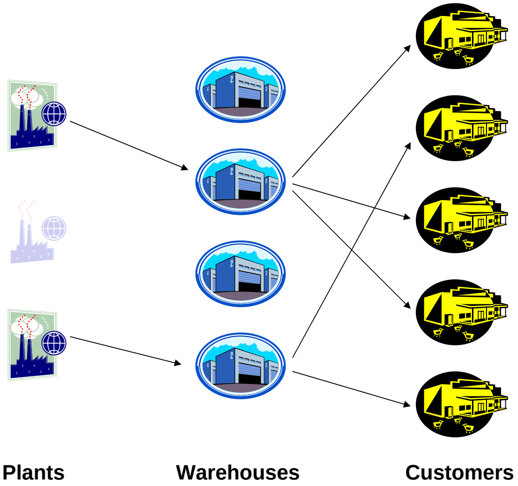
- Single product
- 3 plants, each with capacity 280
- 4 warehouses, each with capacity 280
- 10 customers, average demand 60
- 4 scenarios
    - Nominal scenario
    - 3 disruption scenarios

Example

- Single product
- 3 plants, each with capacity 280
- 4 warehouses, each with capacity 280
- 10 customers, average demand 60
- 4 scenarios
  - Nominal scenario
  - 3 disruption scenarios
- Costs
  - $f_j = 120{,}000$ on average for plants
  - $f_j = 25{,}000$ on average for warehouses
  - $d_{ijn} = 500$ on average
  - Penalty cost for unmet demand $= 10{,}000$
- Scenario upper-bounds $= 800{,}000$

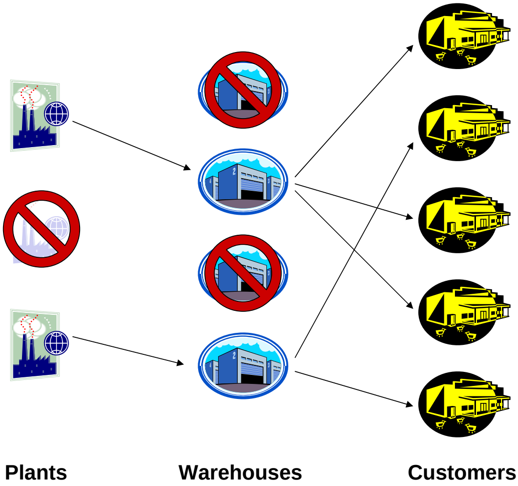
New Solution

Cost: $580,758 (vs. $521,163)



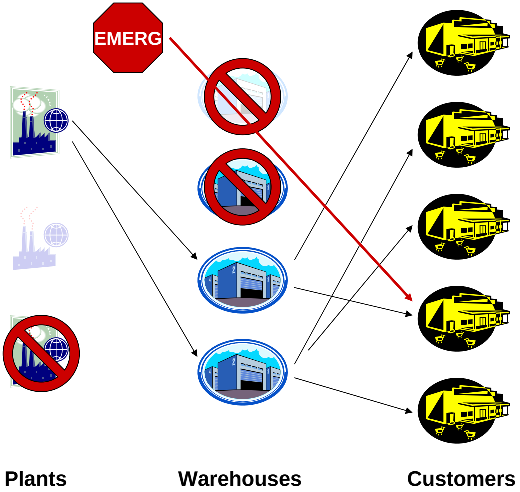
Solution Performance When Disruptions Occur

Scenario 1: Cost \$580,758 (vs. \$521,163)



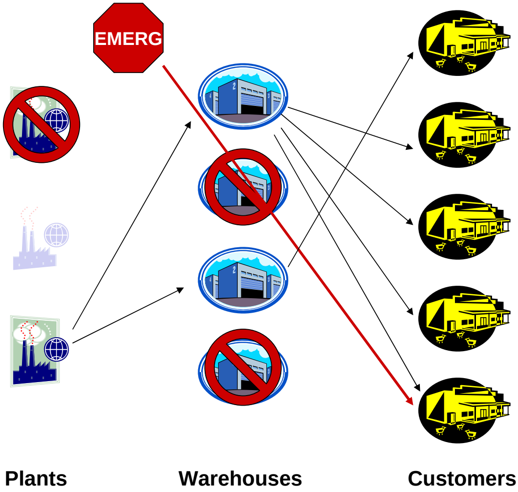
Solution Performance When Disruptions Occur

Scenario 2: Cost $800,000 (vs. $982,241)

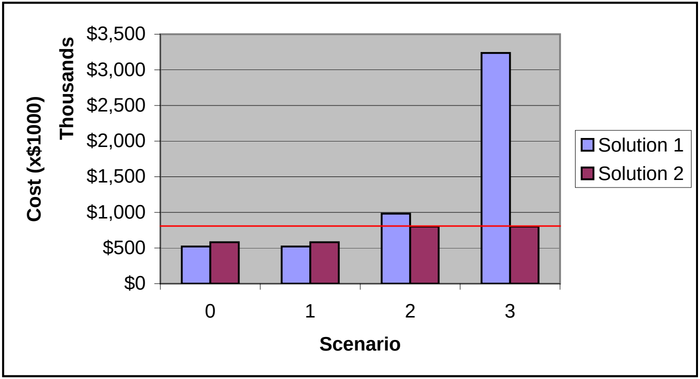


Solution Performance When Disruptions Occur

Scenario 3: Cost $800,000 (vs. $3,237,107)



Solution Performance: Summary



Scenario Costs Equal Upper Bounds

- Recall:

$$\begin{aligned} \text{minimize } & c_0(X, Y) \\ \text{subject to } & c_s(X, Y) \le U_s \qquad \forall s \\ & (X, Y) \in \Omega \end{aligned}$$

Scenario Costs Equal Upper Bounds

- Recall:

$$\begin{aligned} \text{minimize } & c_0(X, Y) \\ \text{subject to } & c_s(X, Y) \leq U_s \qquad \forall s \\ & (X, Y) \in \Omega \end{aligned}$$

- Nothing to require **optimization** in scenarios
  - Only achieve sufficiently low cost

Possible Approaches

- Include scenario costs in objective function

$$\text{minimize} \quad c_0(X, Y) + \sum_s \gamma_s c_s(X, Y)$$

Possible Approaches

- Include scenario costs in objective function

$$\text{minimize} \quad c_0(X,Y) + \sum_s \gamma_s c_s(X,Y)$$

  - Downside: scaling issues

Possible Approaches

- Include scenario costs in objective function

$$\text{minimize} \quad c_0(X, Y) + \sum_s \gamma_s c_s(X, Y)$$

  - Downside: scaling issues
- Post-optimize scenarios
  - Solve model, fix X
  - Then find optimal Y for each scenario

Possible Approaches

- Include scenario costs in objective function

$$\text{minimize} \quad c_0(X, Y) + \sum_s \gamma_s c_s(X, Y)$$

  - Downside: scaling issues
- Post-optimize scenarios
  - Solve model, fix X
  - Then find optimal Y for each scenario
  - Downside: cumbersome
- Currently investigating other approaches

Conclusions

- Supply chain network design model that accounts for disruptions
- Minimize nominal cost, subject to bound on cost in each scenario
- Next steps:
  - Resolve sub-optimization in scenarios
  - Design algorithm

Questions?

larry.snyder@lehigh.edu