Speed-aware network design

A parametric optimisation approach

Ugo Rosolia, Marc Bataillou Almagro, George Iosifidis, Amit Kumar, Georgios Paschos

amazon | science

Outline

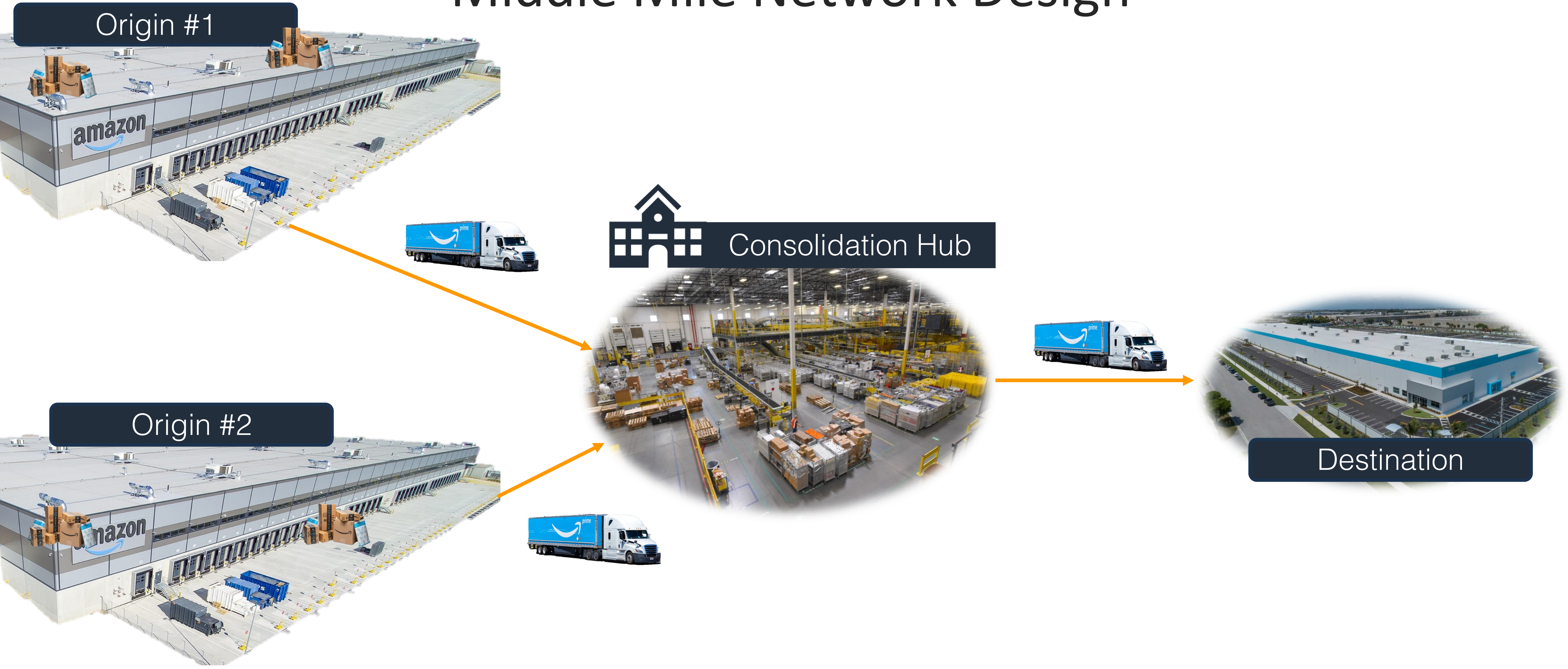
- Motivation
- Problem formulation
- Solution strategy
- Results

Outline

- Motivation
- Problem formulation
- Solution strategy
- Results

Middle Mile Network Design

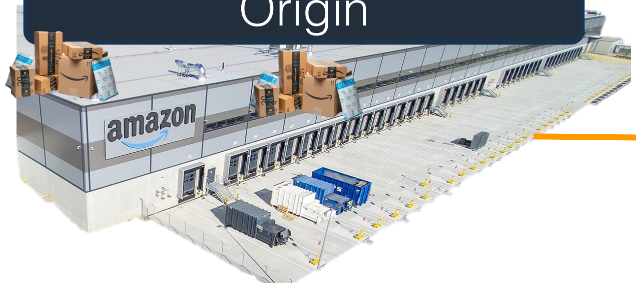
Middle Mile Network Design



amazon | science

Middle Mile Network Design

Origin

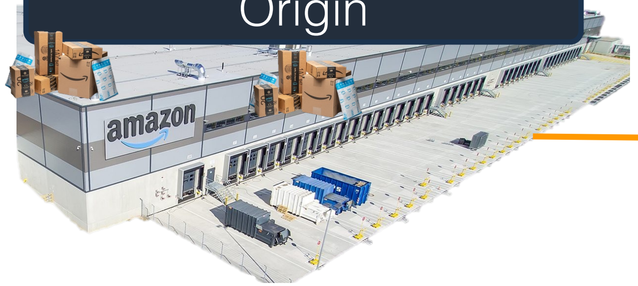


Consolidation Hub

Destination

Middle Mile Network Design

Origin



Consolidation Hub

Destination

Key decisions

1. Connectivity, i.e., buildings to connect.
2. Timing, i.e., trucks departure times.

Middle Mile Network Design

Key decisions

1. Connectivity, i.e., buildings to connect.
2. Timing, i.e., trucks departure times.

Objectives

1. Reduce cost.
2. Minimize carbon emissions.
3. Maximize delivery speed.

Middle Mile Network Design

Key decisions

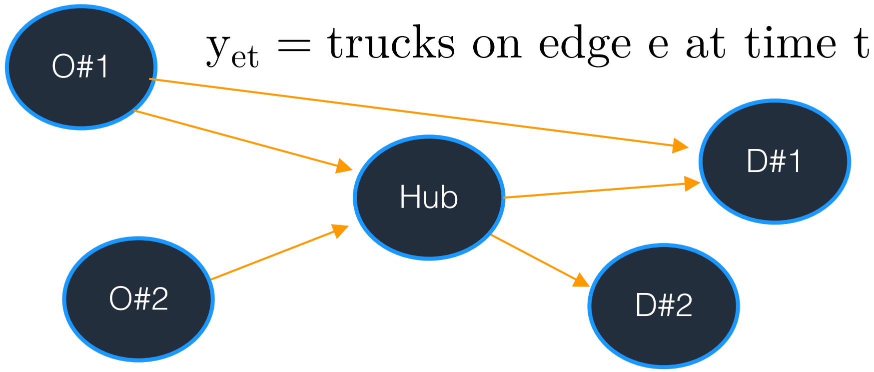
1. Connectivity, i.e., buildings to connect.
2. Timing, i.e., trucks departure times.

Objectives

1. Reduce cost.
2. Minimize carbon emissions.
3. Maximize delivery speed.

Time-expanded network flow problem

$\mathrm{y_{et}} = \mathrm{trucks\ on\ edge\ e\ at\ time\ t}$



Middle Mile Network Design

Key decisions

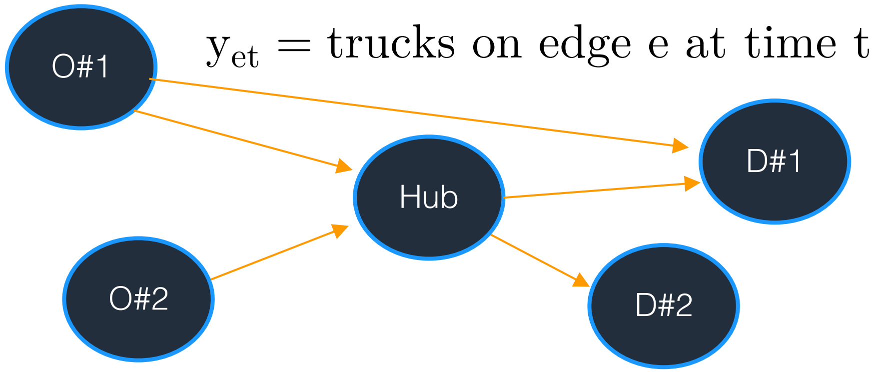
1. Connectivity, i.e., buildings to connect.
2. Timing, i.e., trucks departure times.

Objectives

1. Reduce cost.
2. Minimize carbon emissions.
3. Maximize delivery speed.

Time-expanded network flow problem

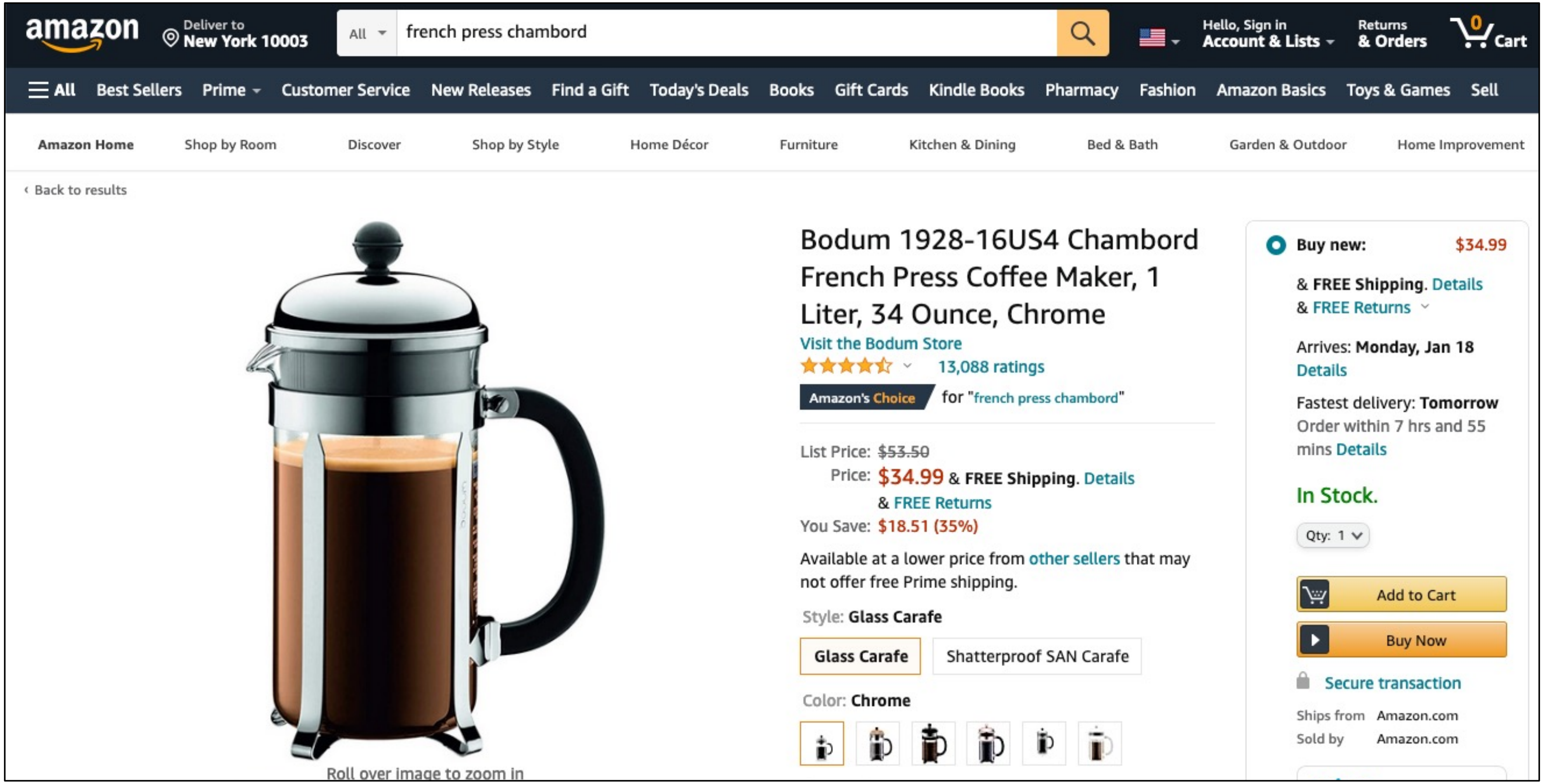
$$y_{et} = \text{trucks on edge e at time t}$$



Intractable at Amazon's scale!

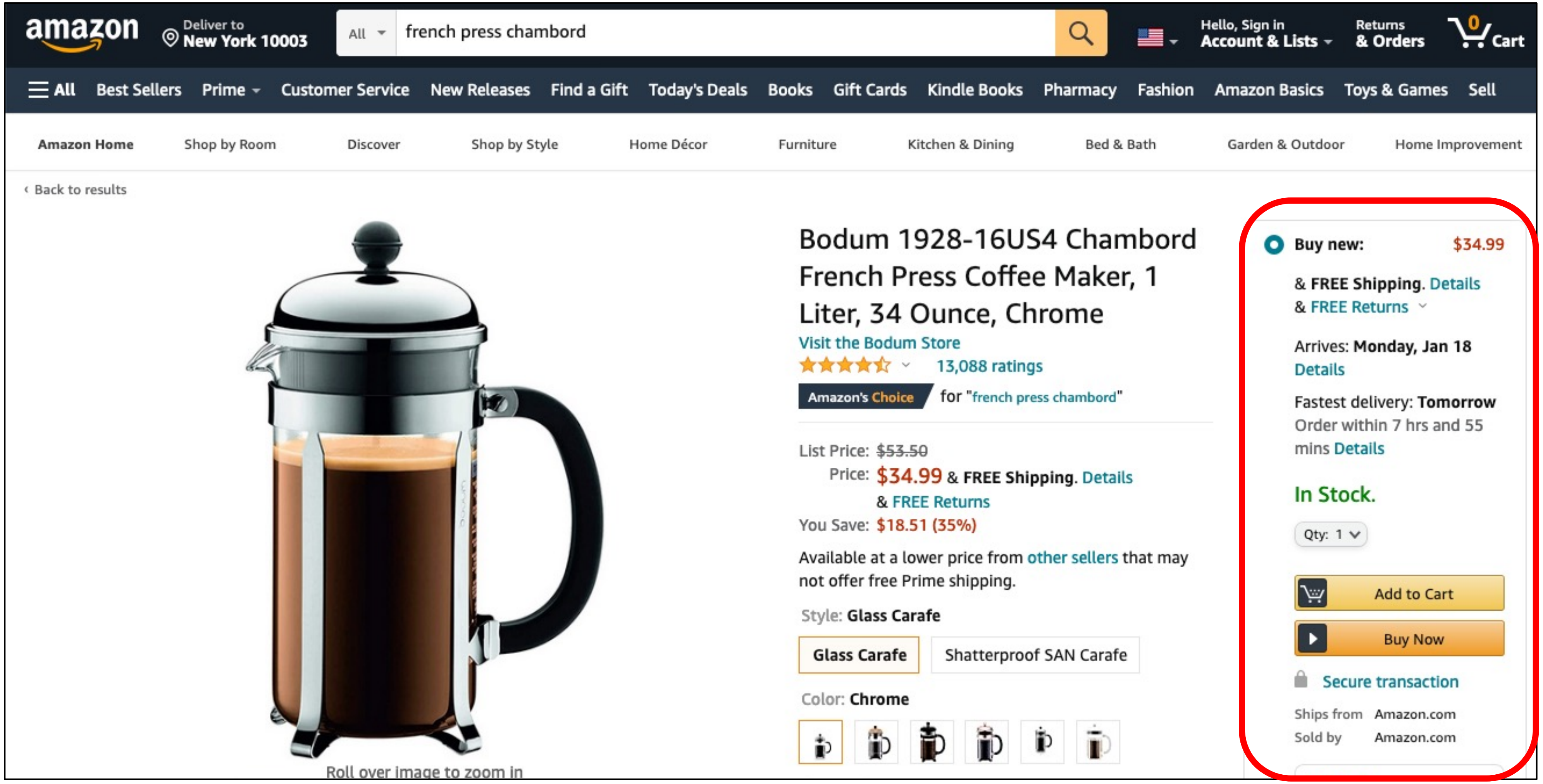
Why considering speed?

Why considering speed?

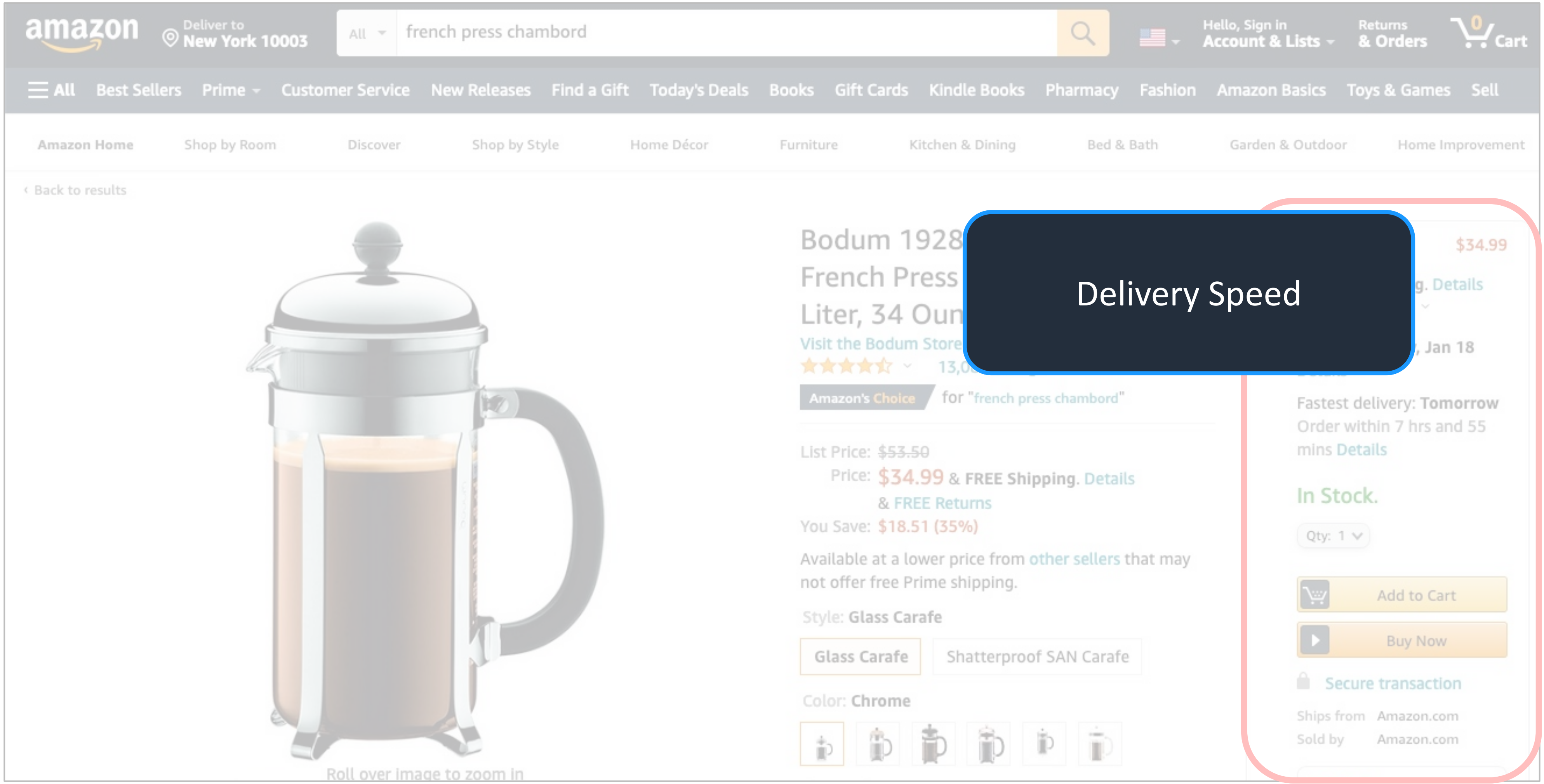


amazon | science

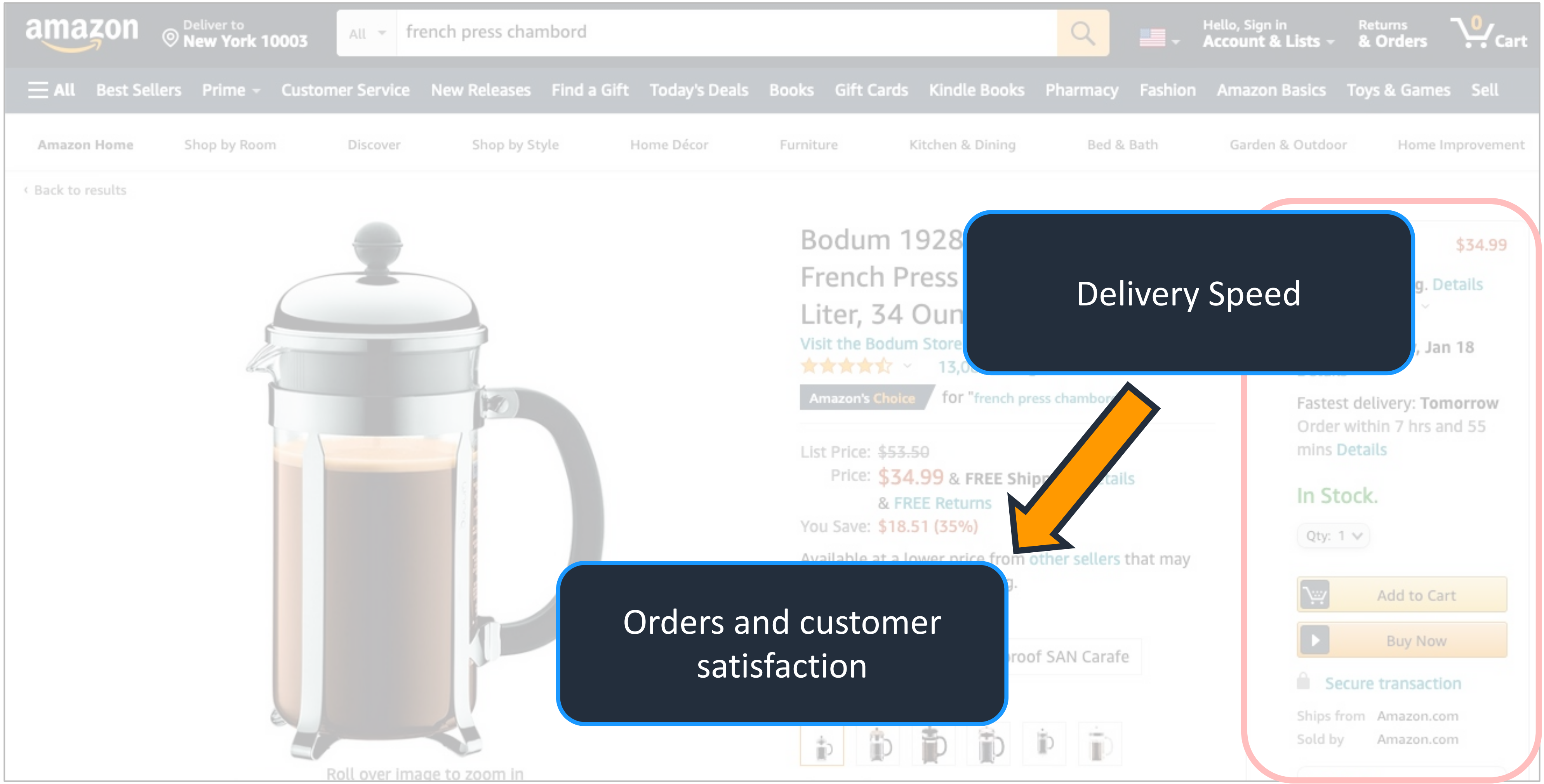
Why considering speed?



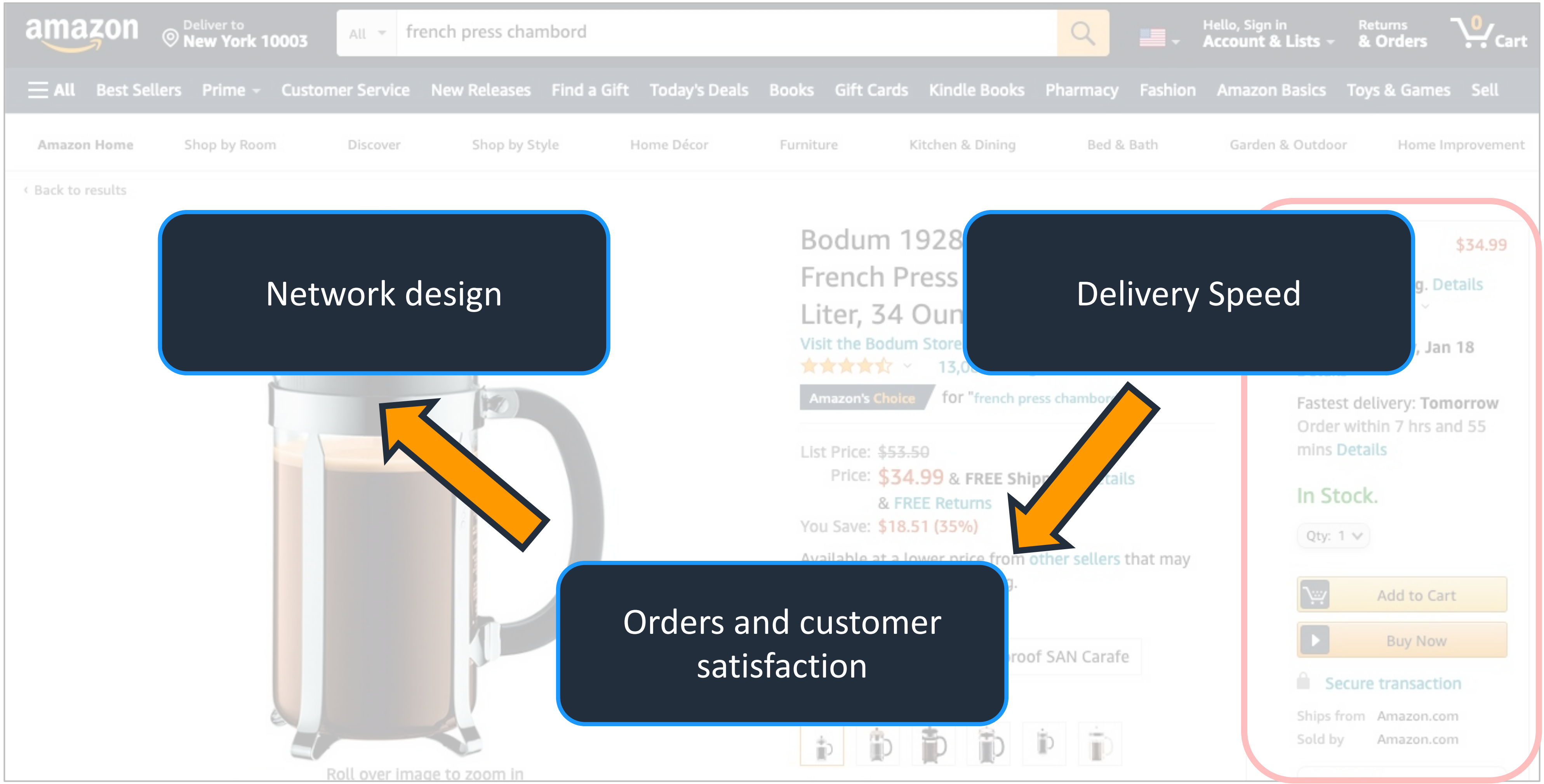
Why considering speed?



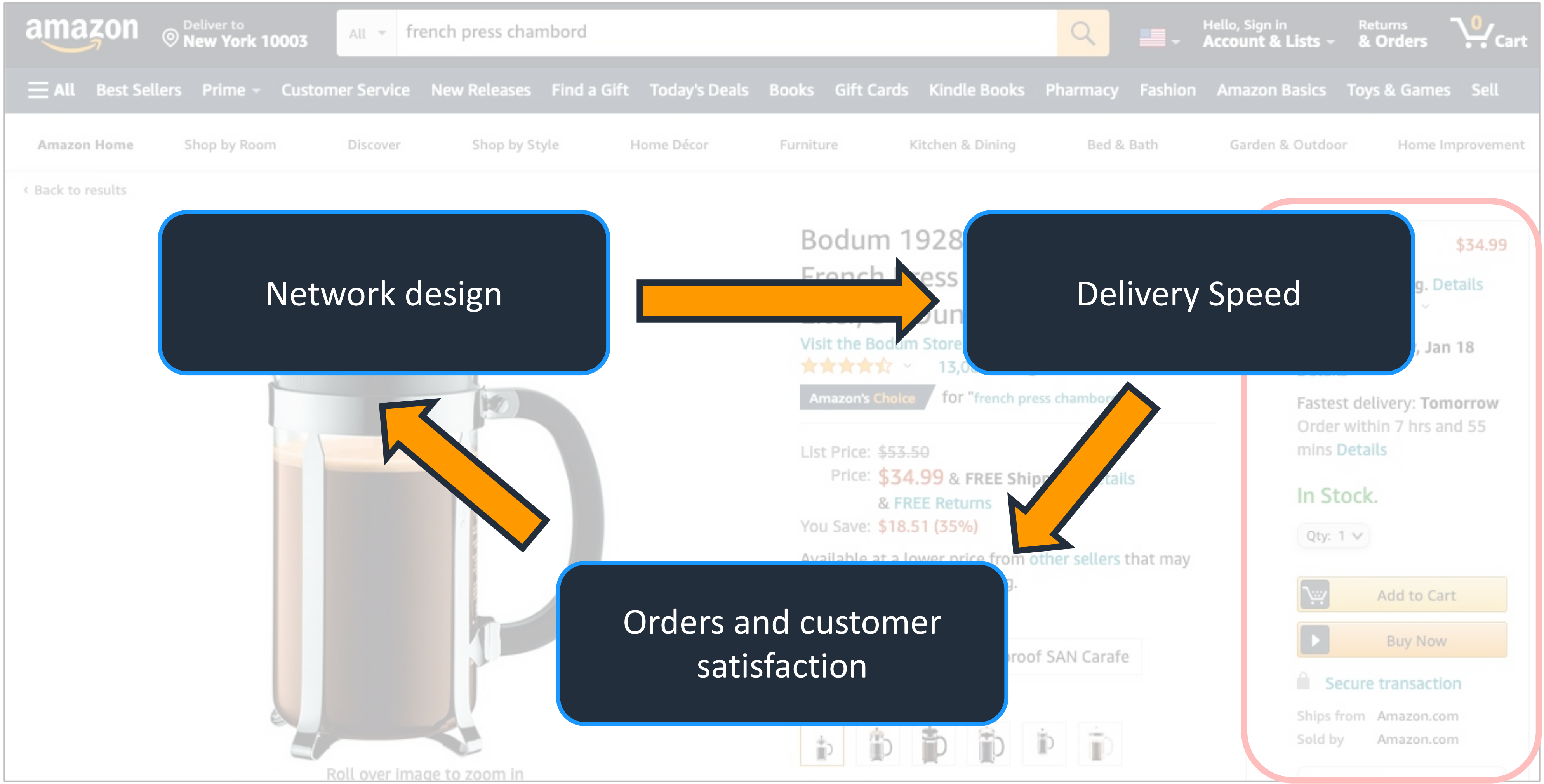
Why considering speed?



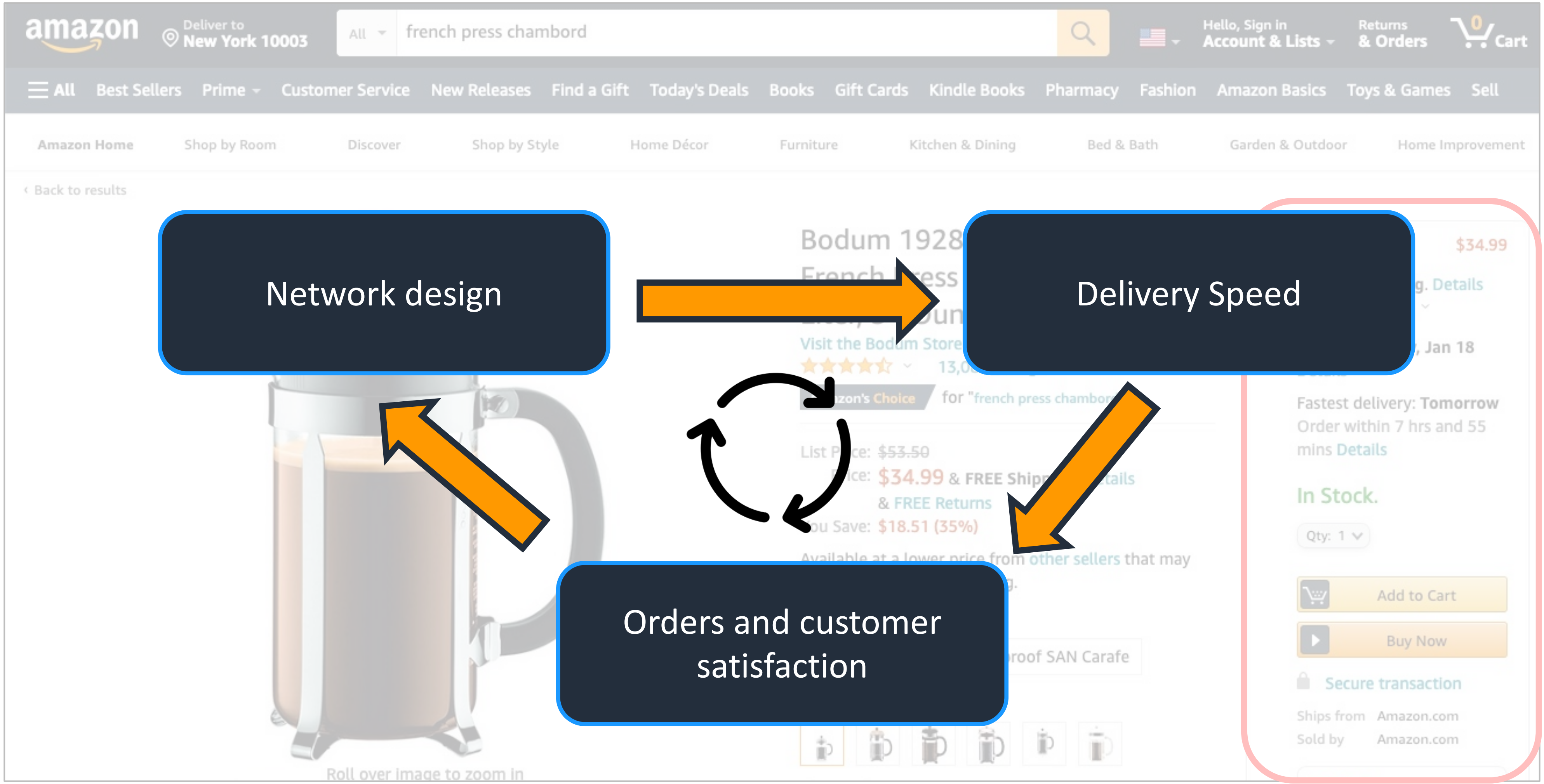
Why considering speed?



Why considering speed?

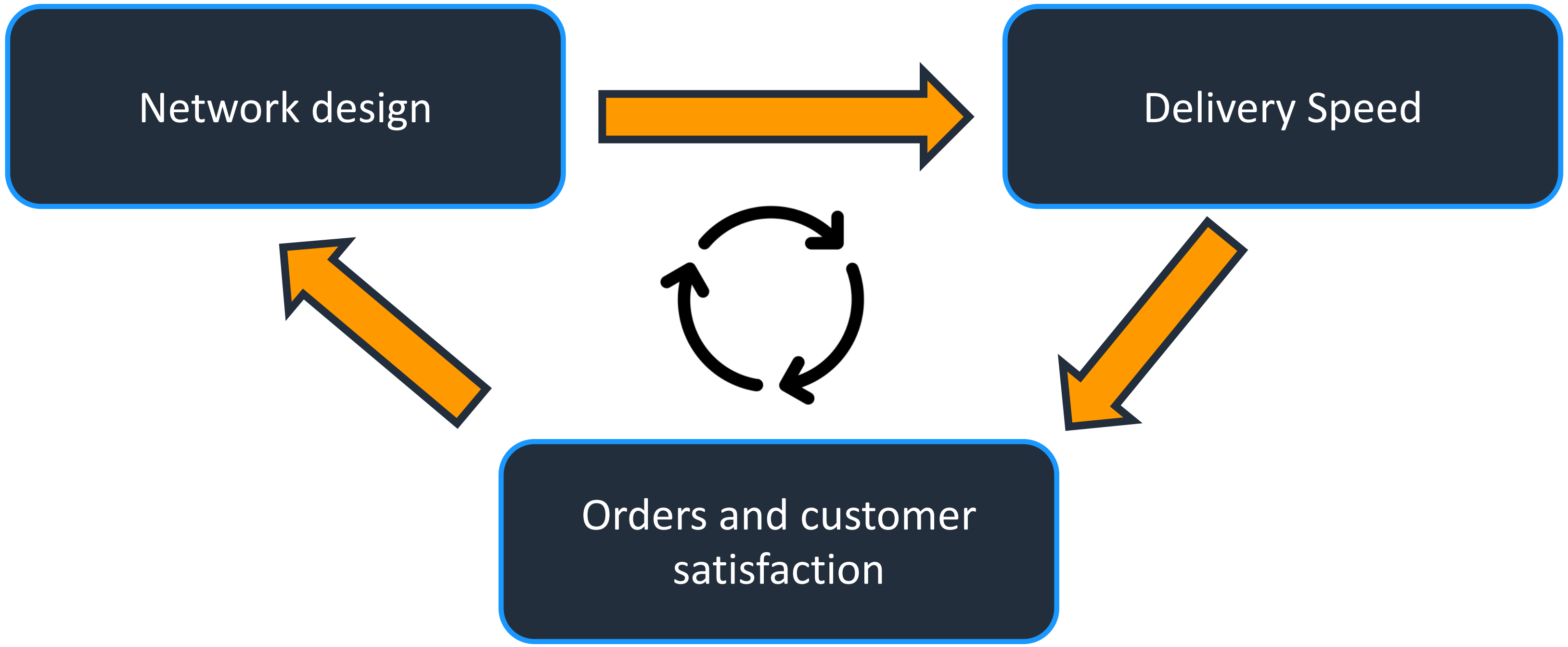


Why considering speed?



Why considering speed?

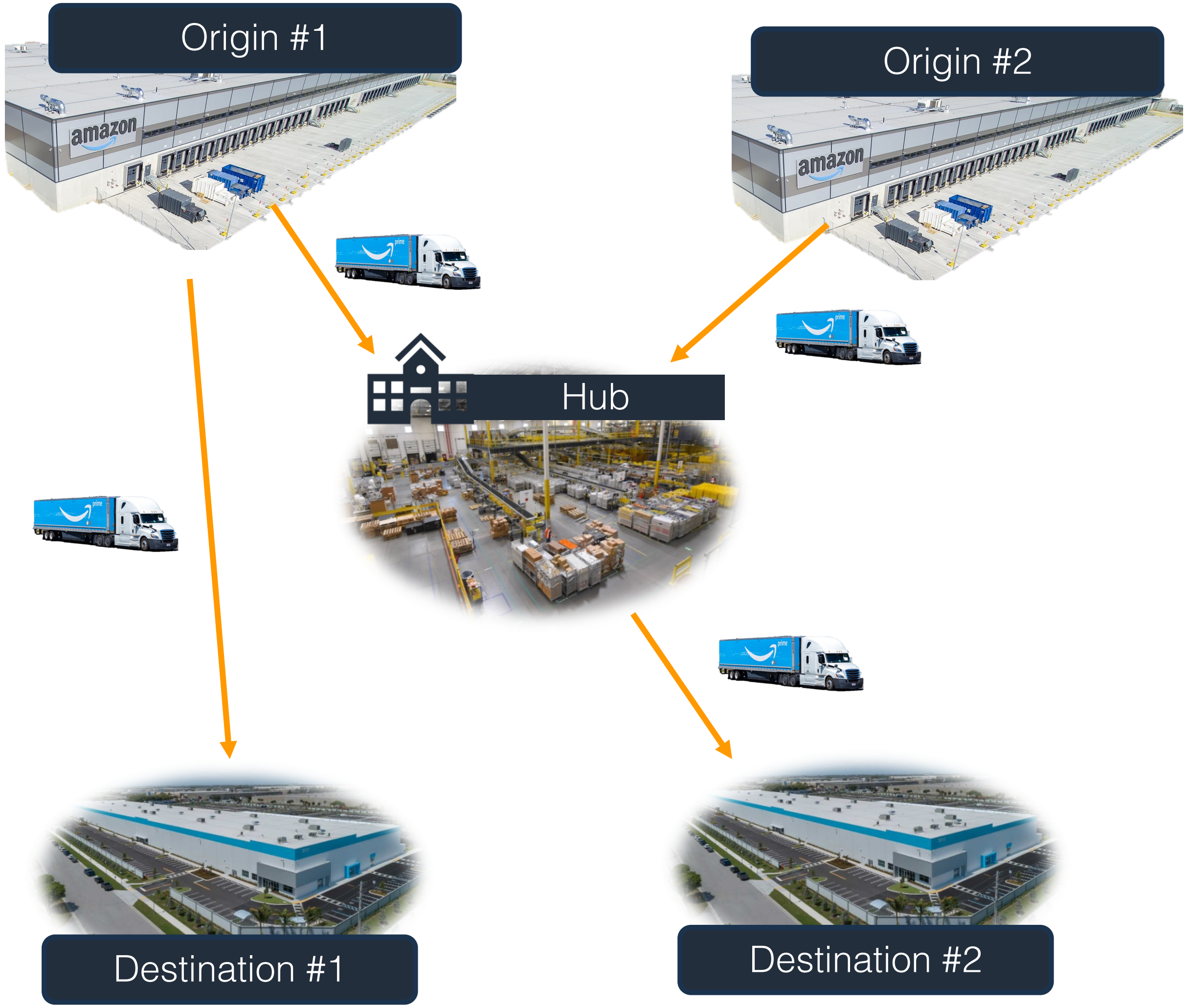
Key objective: Design the <u>cheapest and fastest</u> network for our customers



Outline

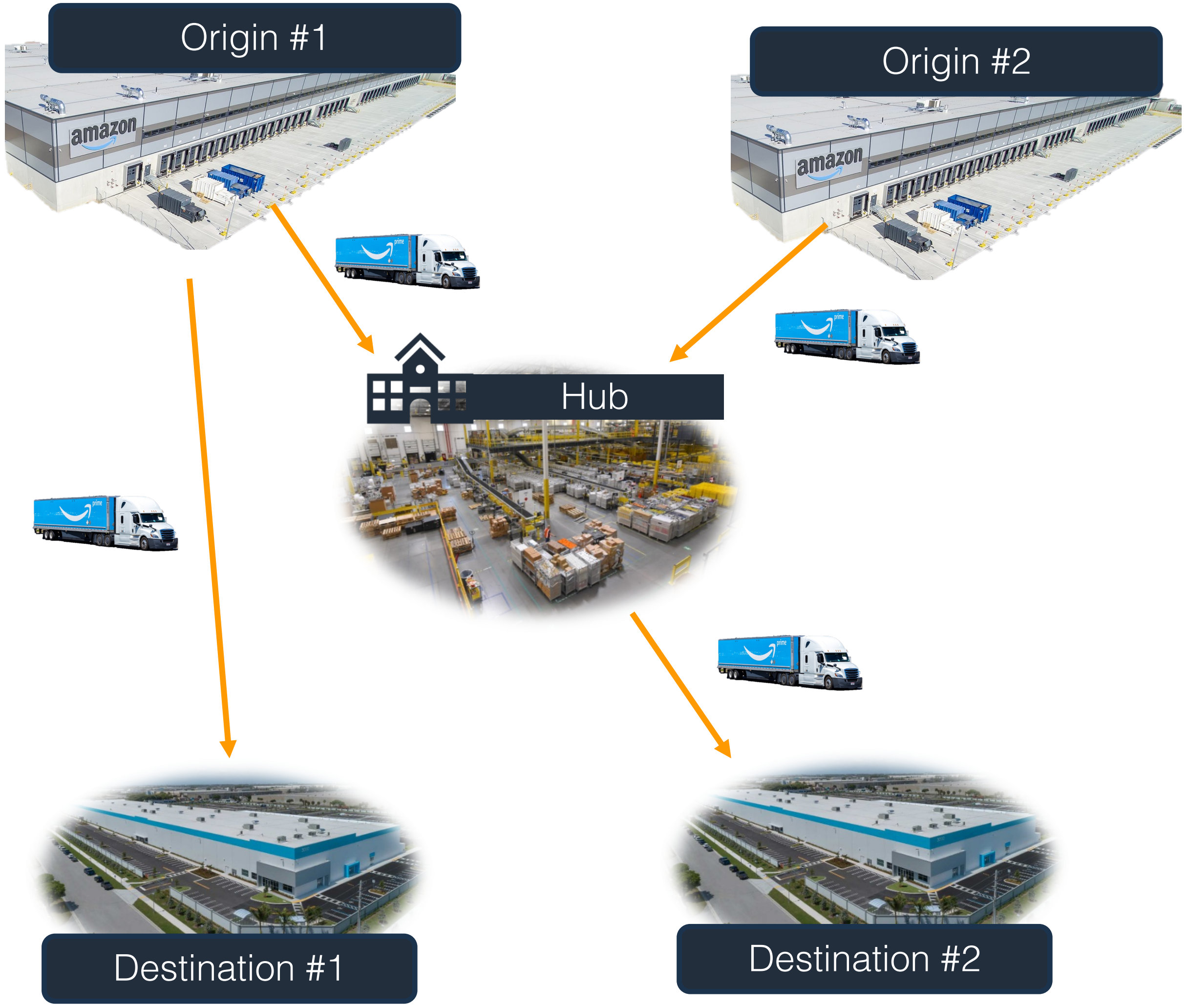
- Motivation
- Problem formulation
- Solution strategy
- Results

Network Design: Connectivity



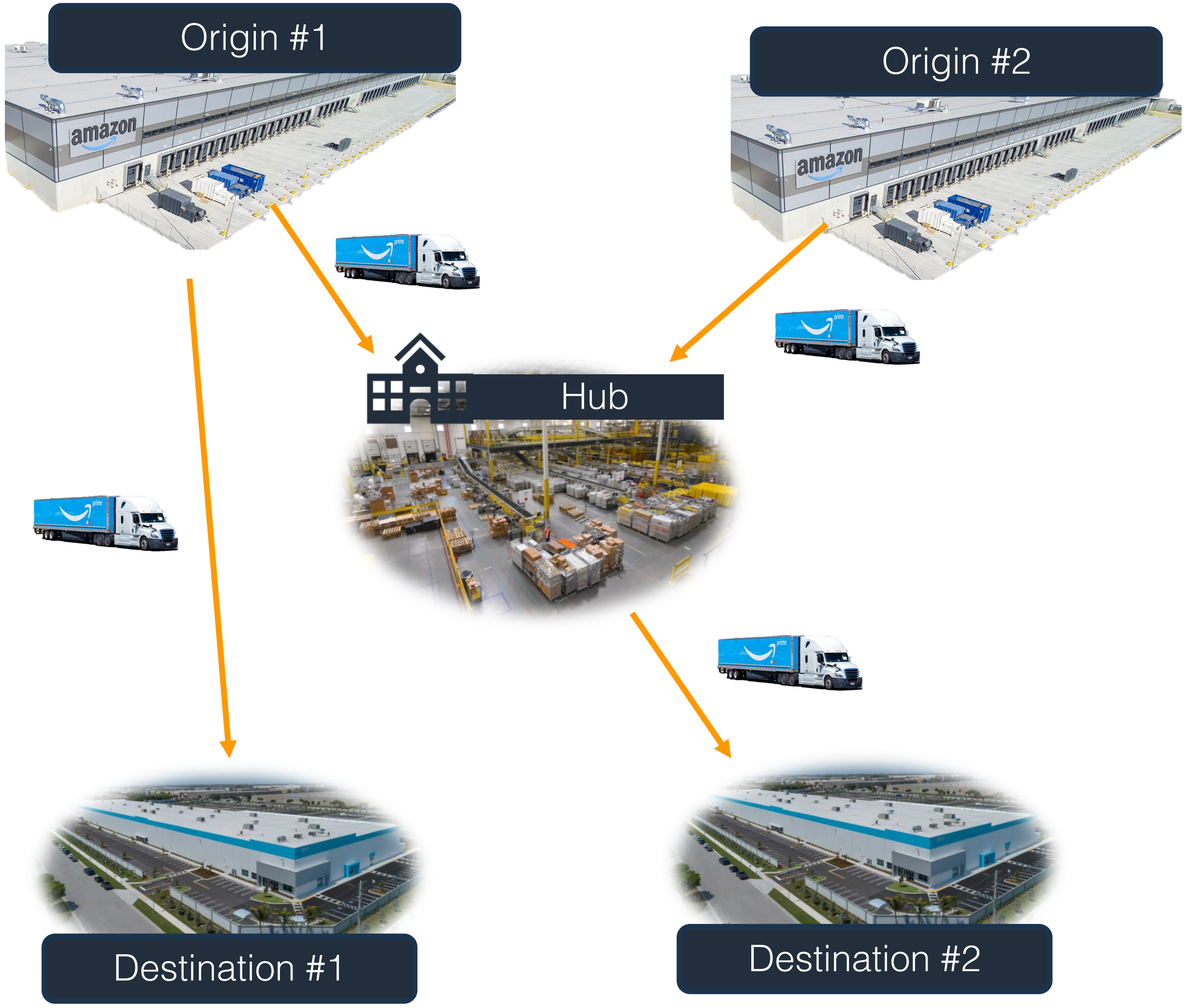
amazon | science

Network Design: Connectivity



$$\min_{\boldsymbol{p},\boldsymbol{v}} \quad \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y})$$
$$\text{s.t.} \quad (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork}$$

Network Design: Connectivity



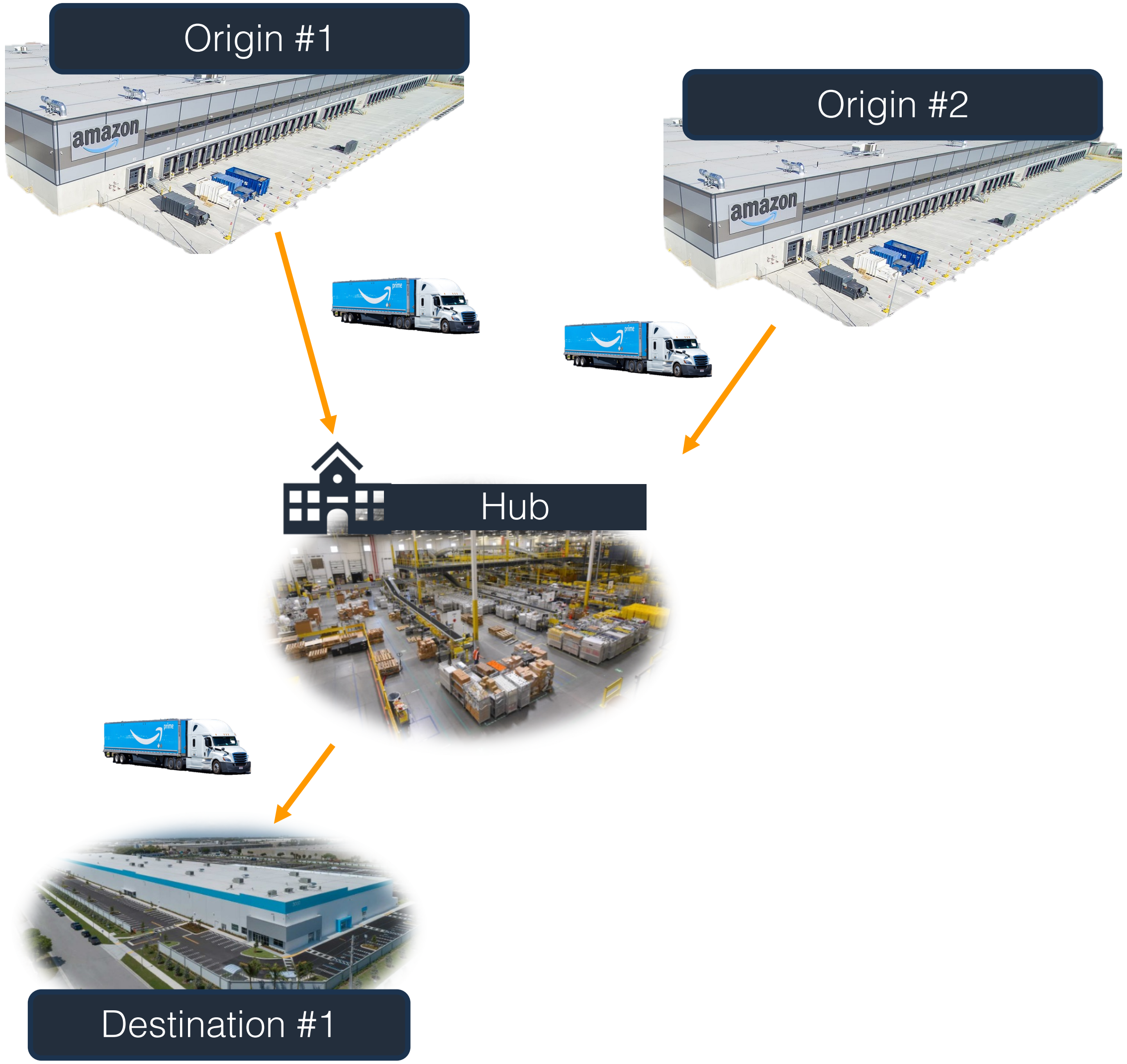
$$\min_{\boldsymbol{p},\boldsymbol{v}} \quad \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y})$$
$$\text{s.t.} \quad (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork}$$

Path vector variable

Trucks vector variable

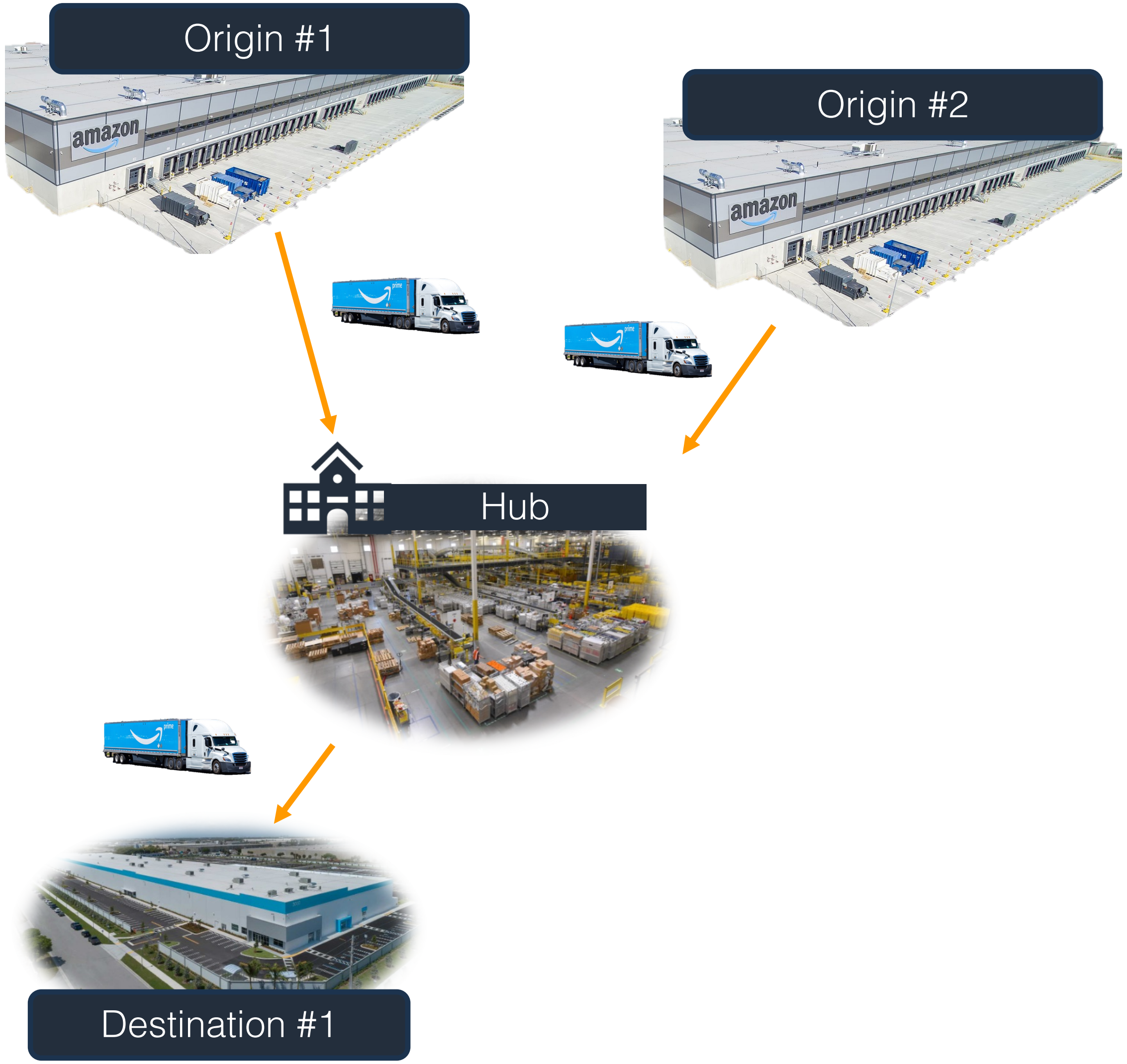
amazon | science

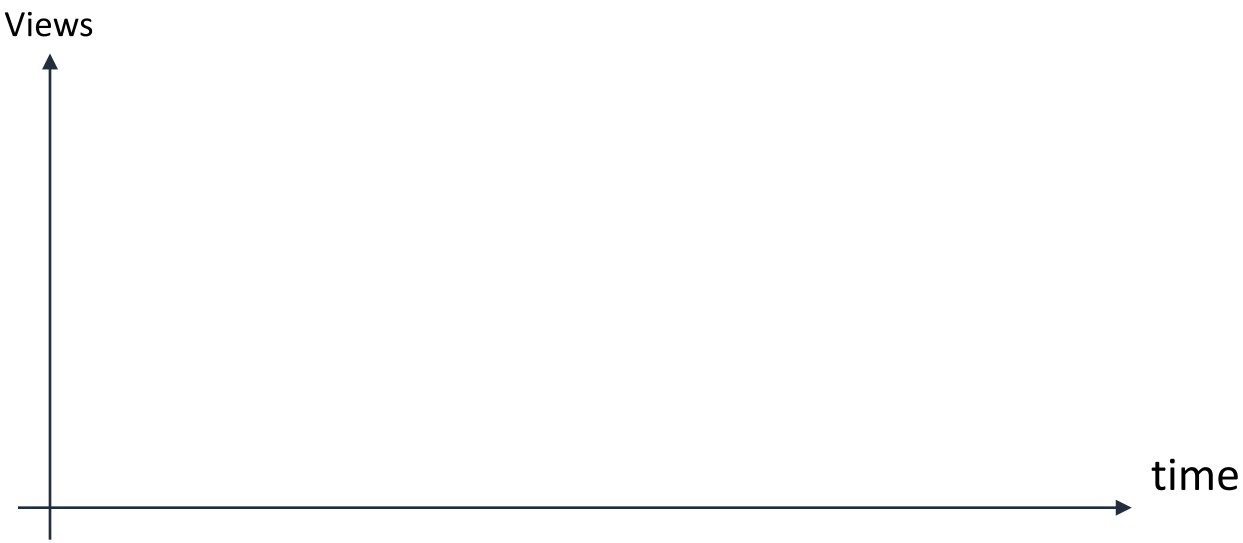
Network Design: Timing



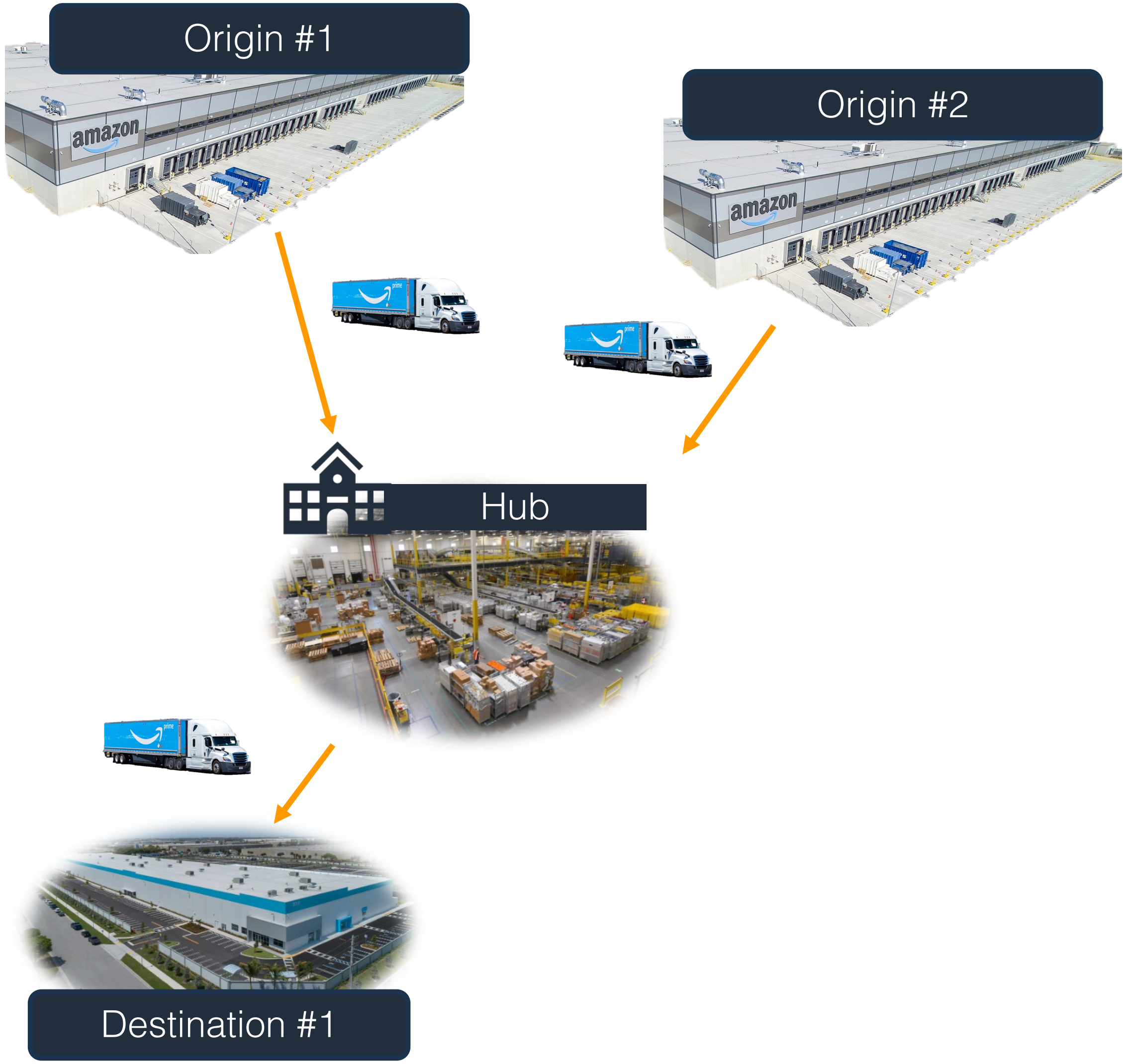
amazon | science

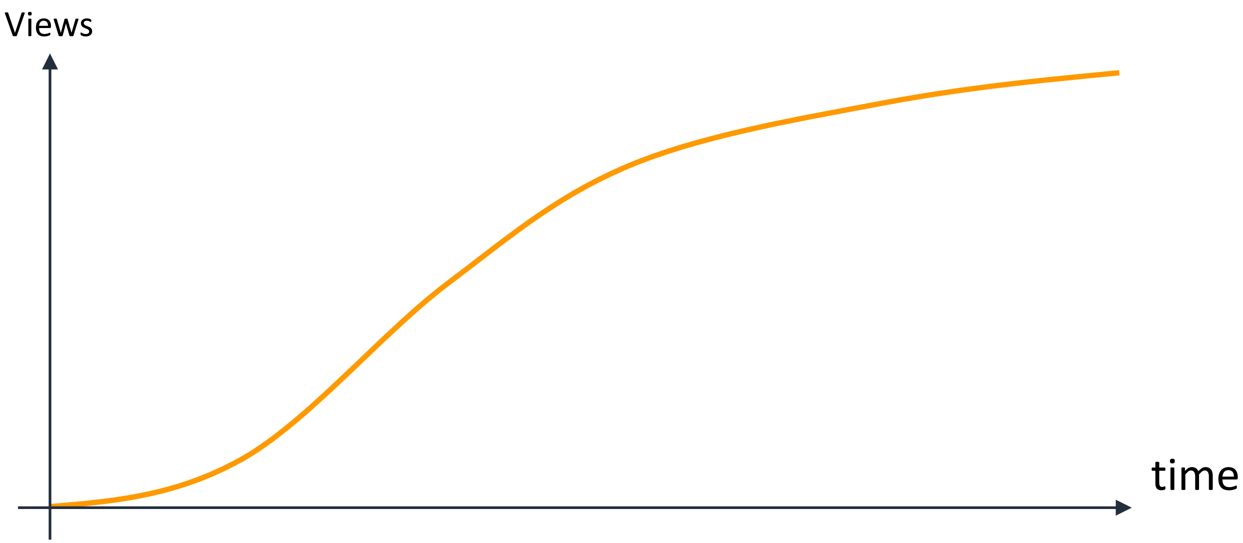
Network Design: Timing



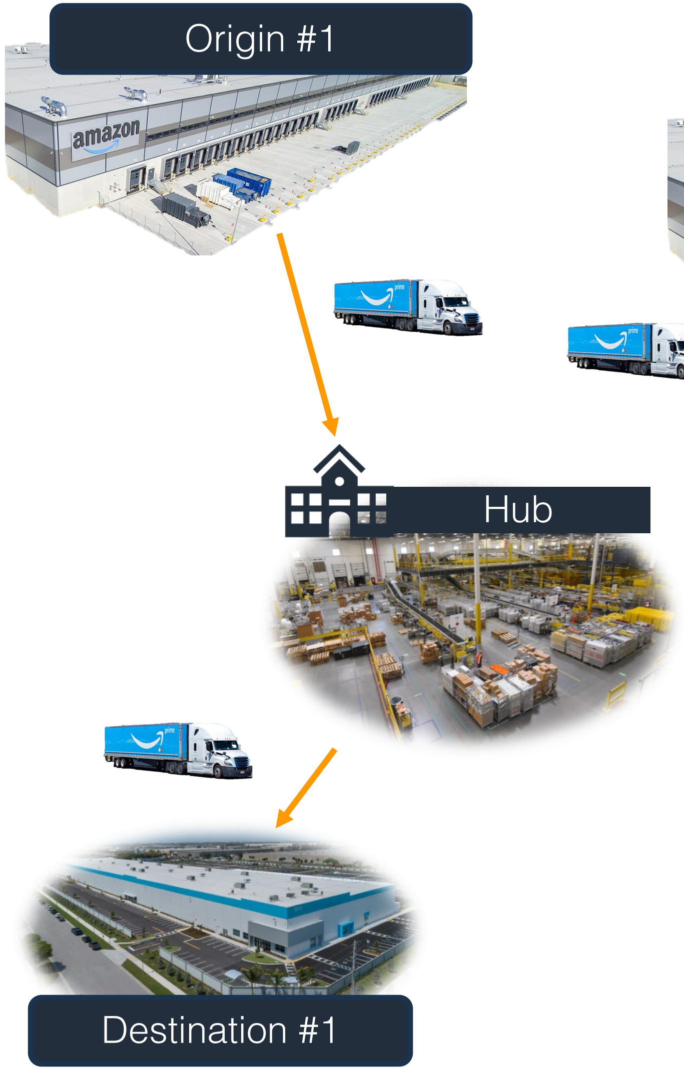


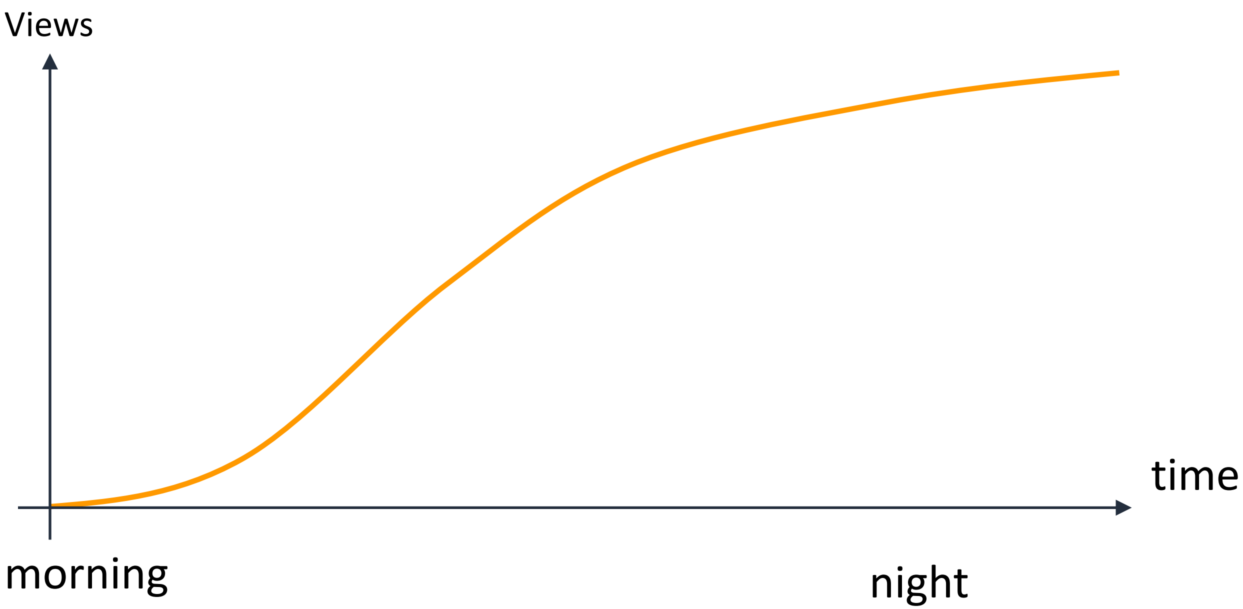
Network Design: Timing





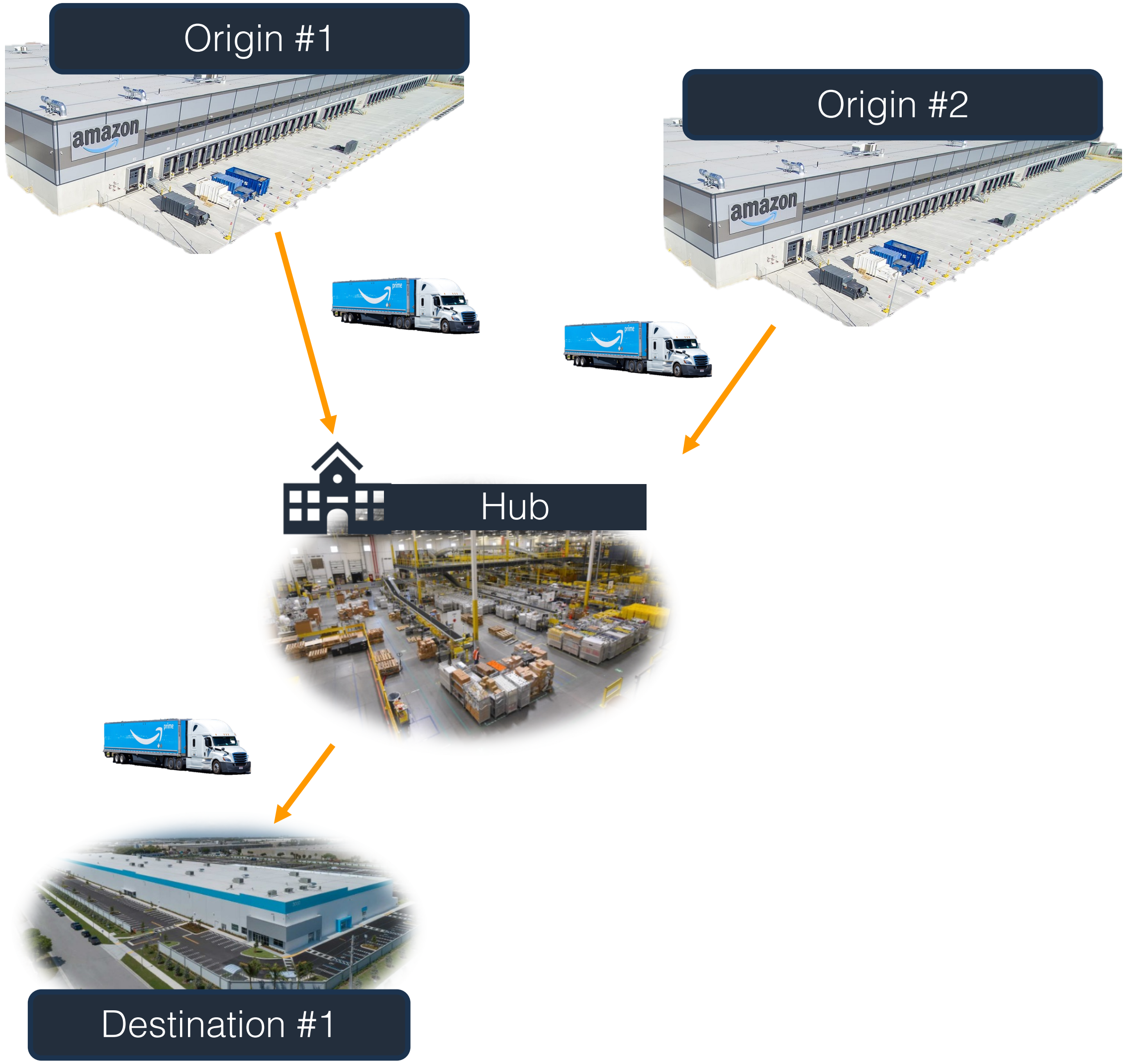
Network Design: Timing

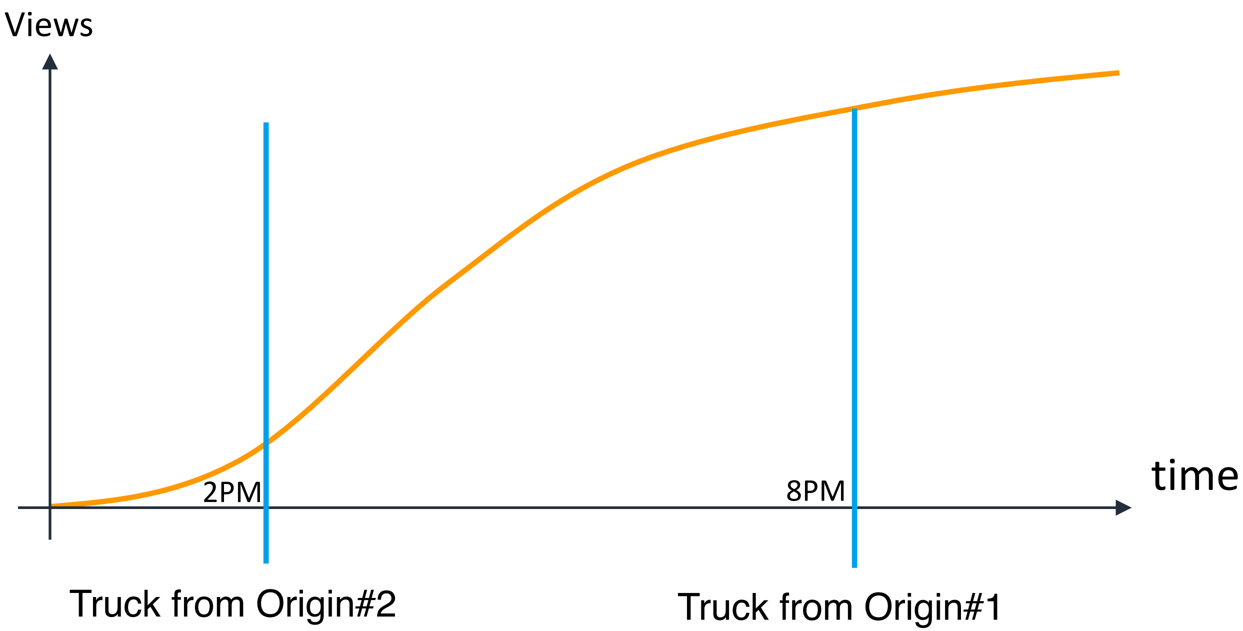




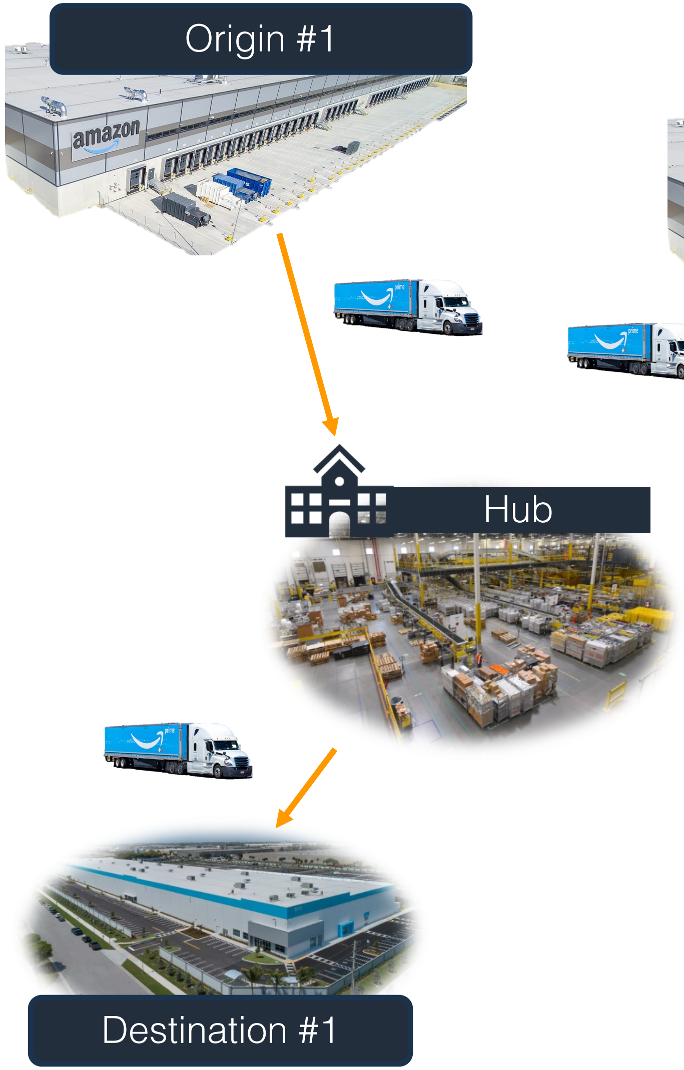
amazon | science

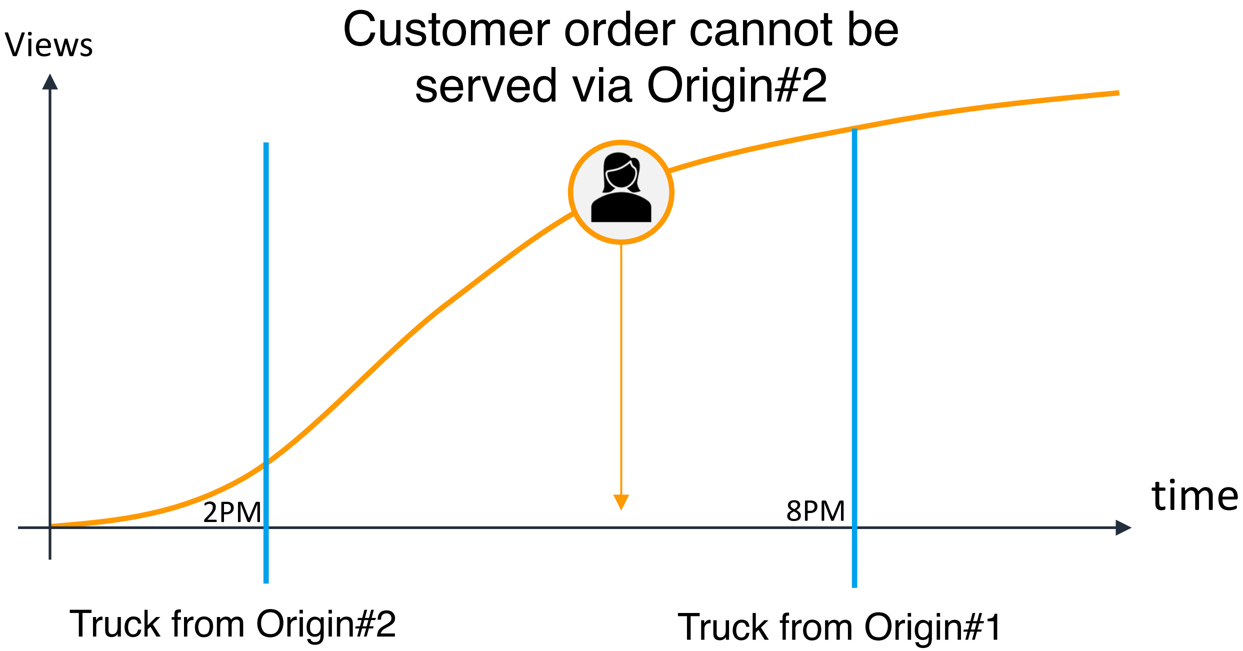
Network Design: Timing





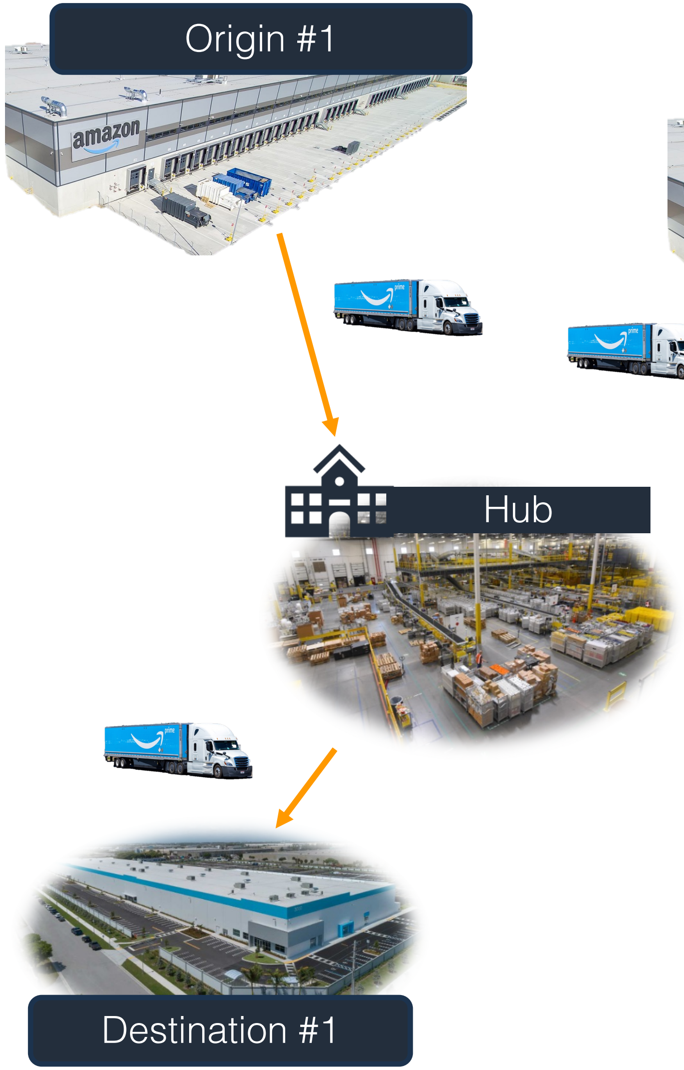
Network Design: Timing

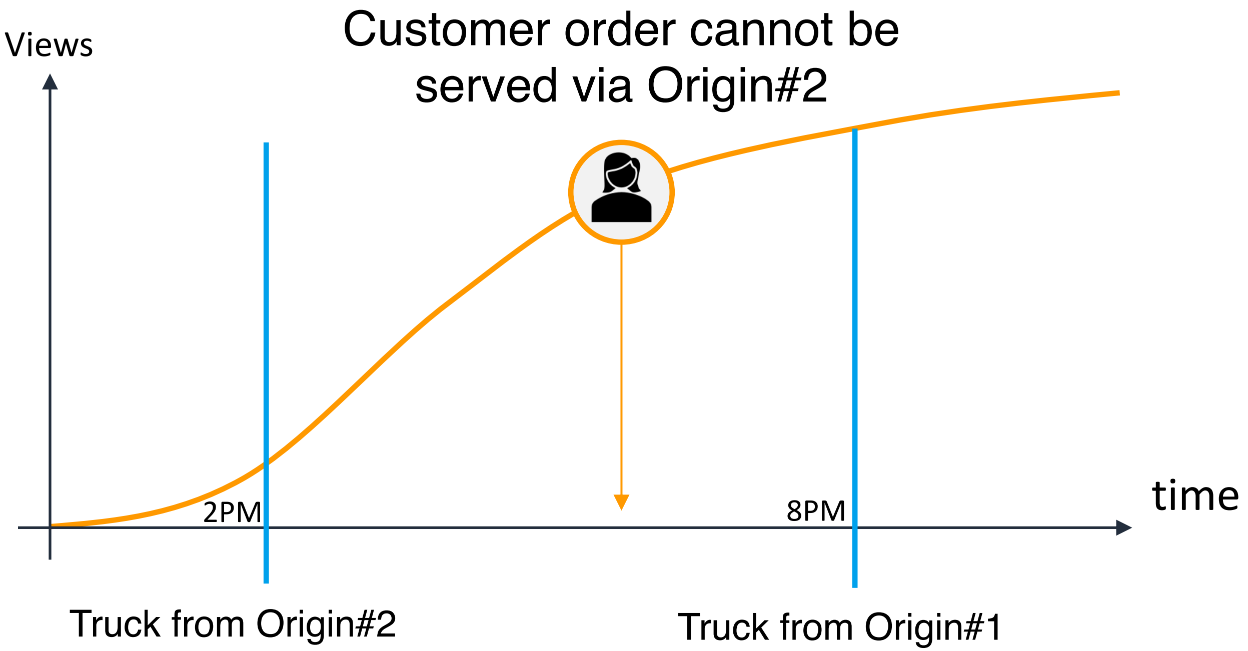




amazon | science

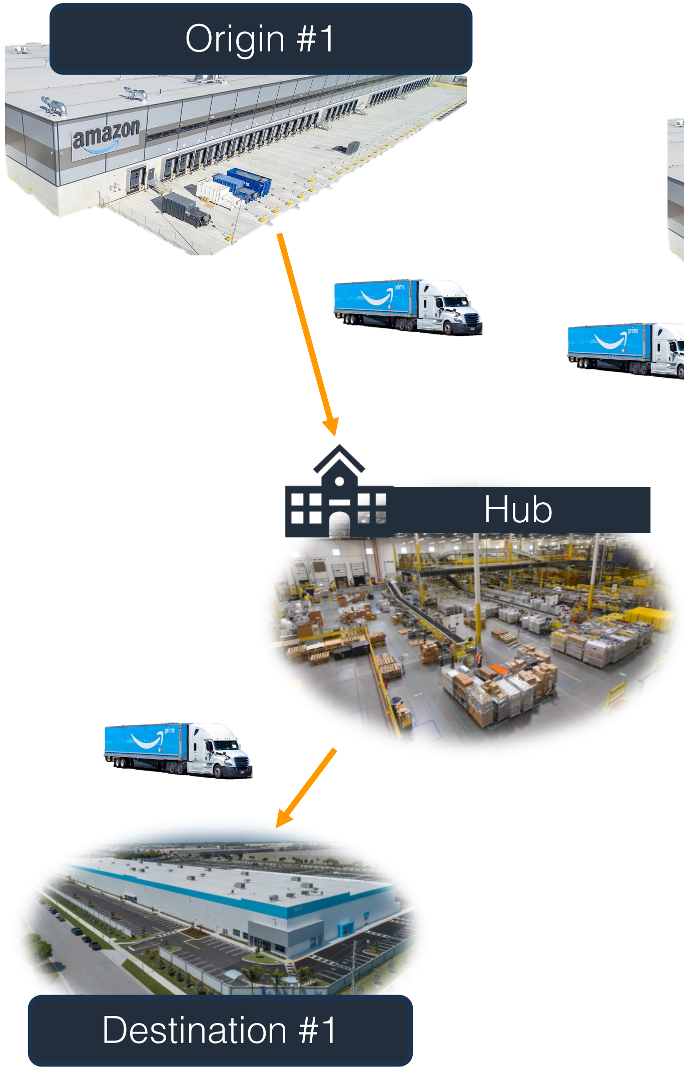
Network Design: Timing

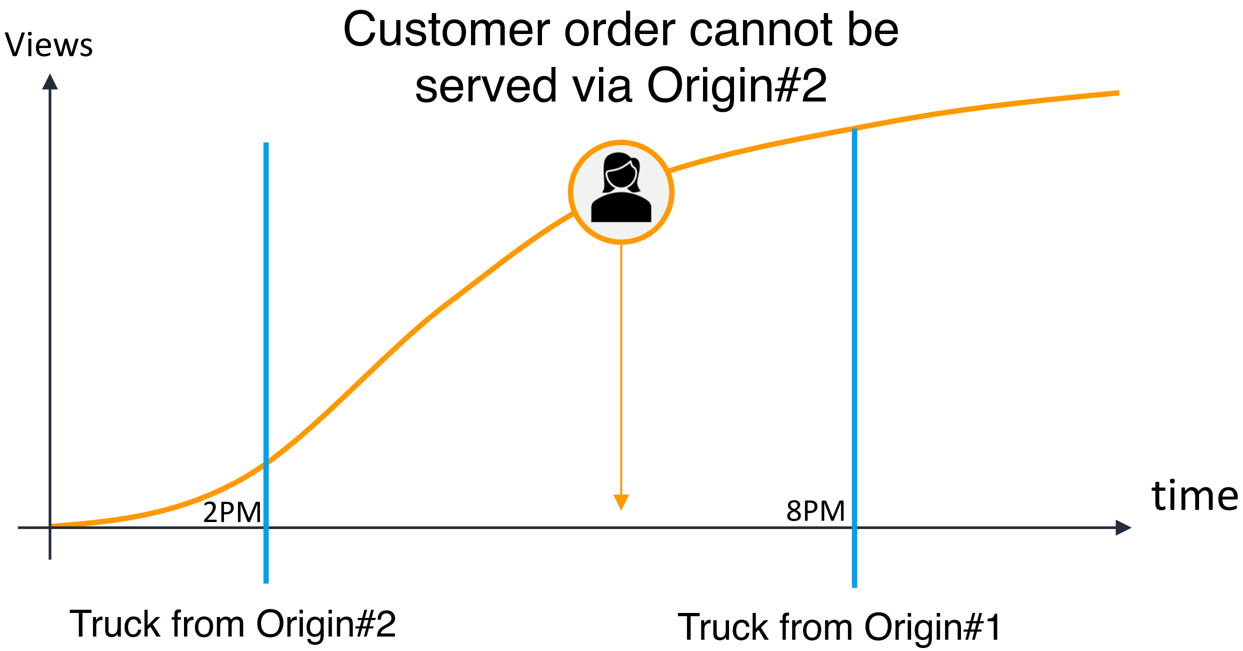




$$\max_{z} \quad \mathtt{Speed}(z)$$
$$\text{s.t.} \quad z \in \mathtt{FeasibleSchedule}(p)$$

Network Design: Timing





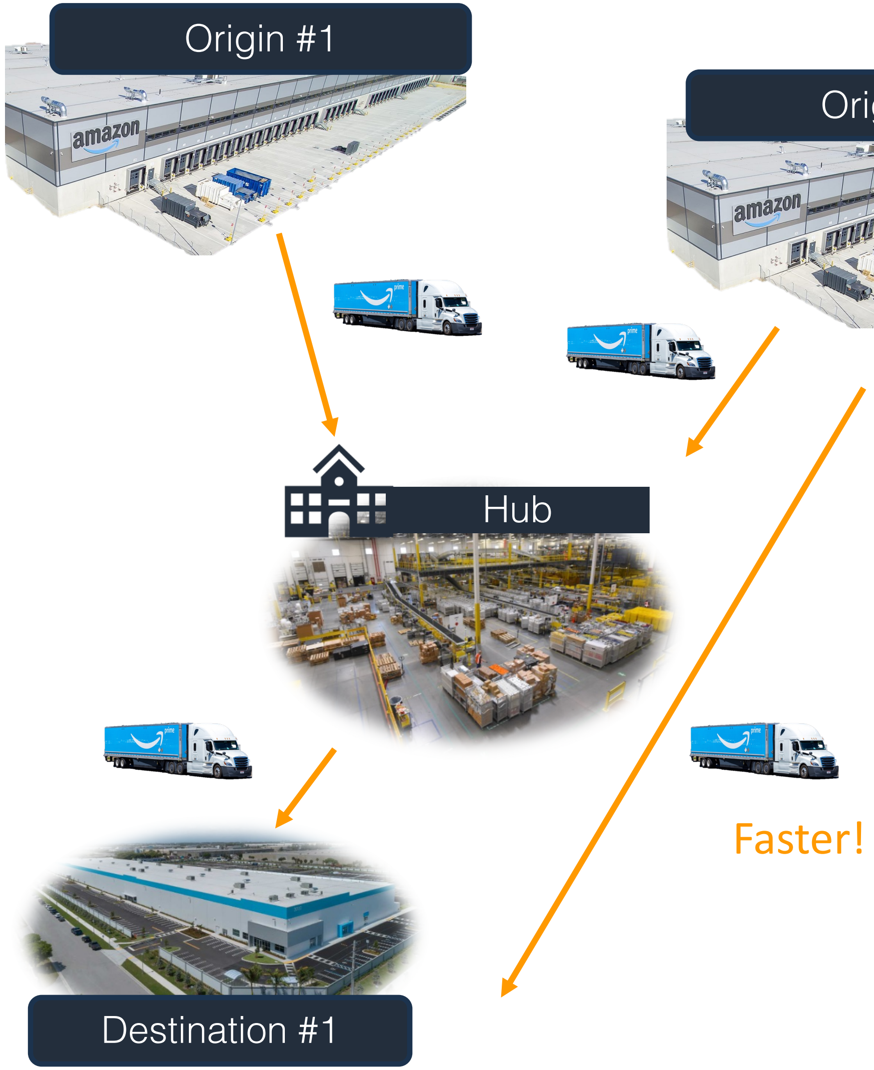
$$\max_{\boldsymbol{z}} \quad \texttt{Speed}(\boldsymbol{z})$$
$$\text{s.t.} \quad \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p})$$

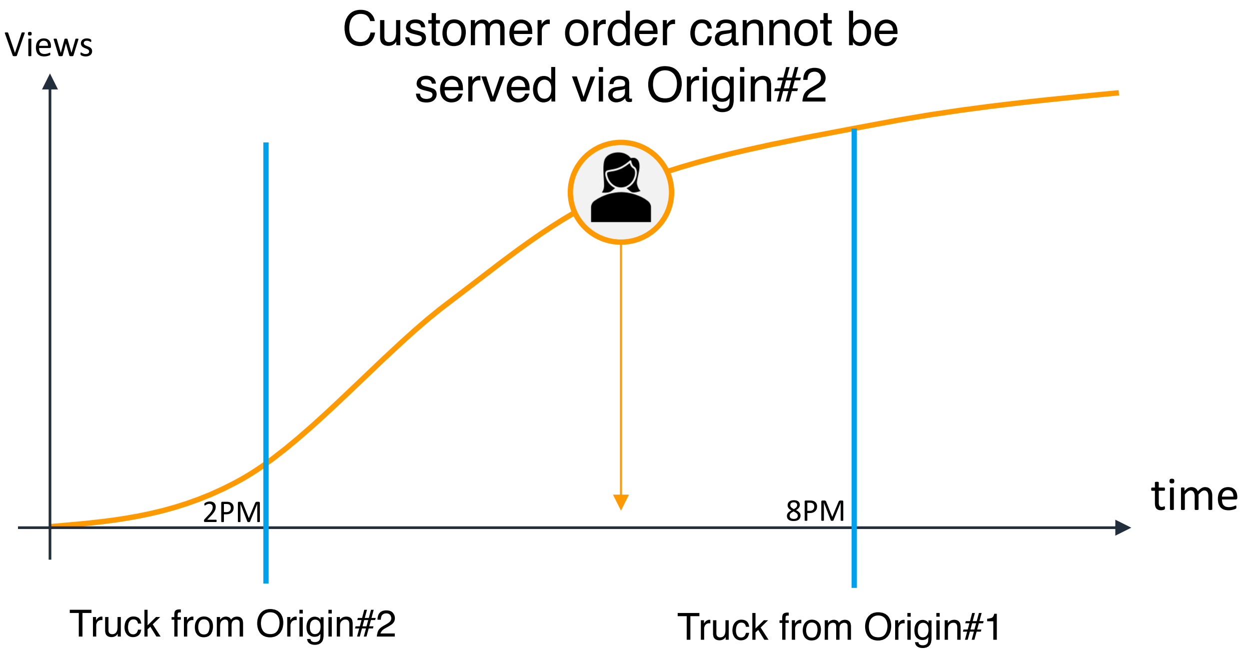
Truck schedule vector variable

Dependency on connectivity

amazon | science

Network Design: Timing





$$
\begin{aligned}
\max_{\boldsymbol{z}} \quad & \texttt{Speed}(\boldsymbol{z}) \\
\text{s.t.} \quad & \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p})
\end{aligned}
$$

Truck schedule vector variable

Dependency on connectivity

Network Design: The joint problem

$$\begin{aligned} \min_{\boldsymbol{p},\boldsymbol{y},\boldsymbol{z}} \quad & \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y}) - \texttt{Speed}(\boldsymbol{z}) \\ \text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork} \\ & \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p}) \end{aligned}$$

Network Design: The joint problem

$$\begin{aligned} \min_{\boldsymbol{p},\boldsymbol{y},\boldsymbol{z}} \quad & \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y}) - \texttt{Speed}(\boldsymbol{z}) \\ \text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork} \\ & \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p}) \end{aligned}$$

Why is this problem hard to solve?

Network Design: The joint problem

$$\begin{aligned} \min_{\boldsymbol{p},\boldsymbol{y},\boldsymbol{z}} \quad & \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y}) - \texttt{Speed}(\boldsymbol{z}) \\ \text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork} \\ & \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p}) \end{aligned}$$

Why is this problem hard to solve?

(a) Feasibility: We must consider granular non-convex operational constraints, e.g., site opening hours.

Network Design: The joint problem

$$\begin{aligned} \min_{\boldsymbol{p},\boldsymbol{y},\boldsymbol{z}} \quad & \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y}) - \texttt{Speed}(\boldsymbol{z}) \\ \text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork} \\ & \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p}) \end{aligned}$$

Why is this problem hard to solve?

(a) Feasibility: We must consider granular non-convex operational constraints, e.g., site opening hours.

(b) Speed objective: Inventory at origins impacts the speed given by expensive fast connections.

Network Design: The joint problem

$$
\begin{aligned}
\min_{\boldsymbol{p},\boldsymbol{y},\boldsymbol{z}} \quad & \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y}) - \texttt{Speed}(\boldsymbol{z}) \\
\text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork} \\
& \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p})
\end{aligned}
$$

Why is this problem hard to solve?

(a) Feasibility: We must consider granular non-convex operational constraints, e.g., site opening hours.

(b) Speed objective: Inventory at origins impacts the speed given by expensive fast connections.

(c) Scale: Billions of variables to model hourly decisions, e.g., when a truck should depart.

Network Design: The joint problem

$$\begin{aligned}
\min_{\boldsymbol{p},\boldsymbol{y},\boldsymbol{z}} \quad & \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y}) - \texttt{Speed}(\boldsymbol{z}) \\
\text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork} \\
& \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p})
\end{aligned}$$

Why is this problem hard to solve?

(a) Feasibility: We must consider granular non-convex operational constraints, e.g., site opening hours.

(b) Speed objective: Inventory at origins impacts the speed given by expensive fast connections.

(c) Scale: Billions of variables to model hourly decisions, e.g., when a truck should depart.

(d) Uncertainty: Customers' demand is uncertain, thus we should minimize the expected cost.

Network Design: The joint problem

$$\begin{aligned} \min_{\boldsymbol{p},\boldsymbol{y},\boldsymbol{z}} \quad & \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y}) - \texttt{Speed}(\boldsymbol{z}) \\ \text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork} \\ & \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p}) \end{aligned}$$

Why is this problem hard to solve?

(a) Feasibility: We must consider granular non-convex operational constraints, e.g., site opening hours.

(b) Speed objective: Inventory at origins impacts the speed given by expensive fast connections.

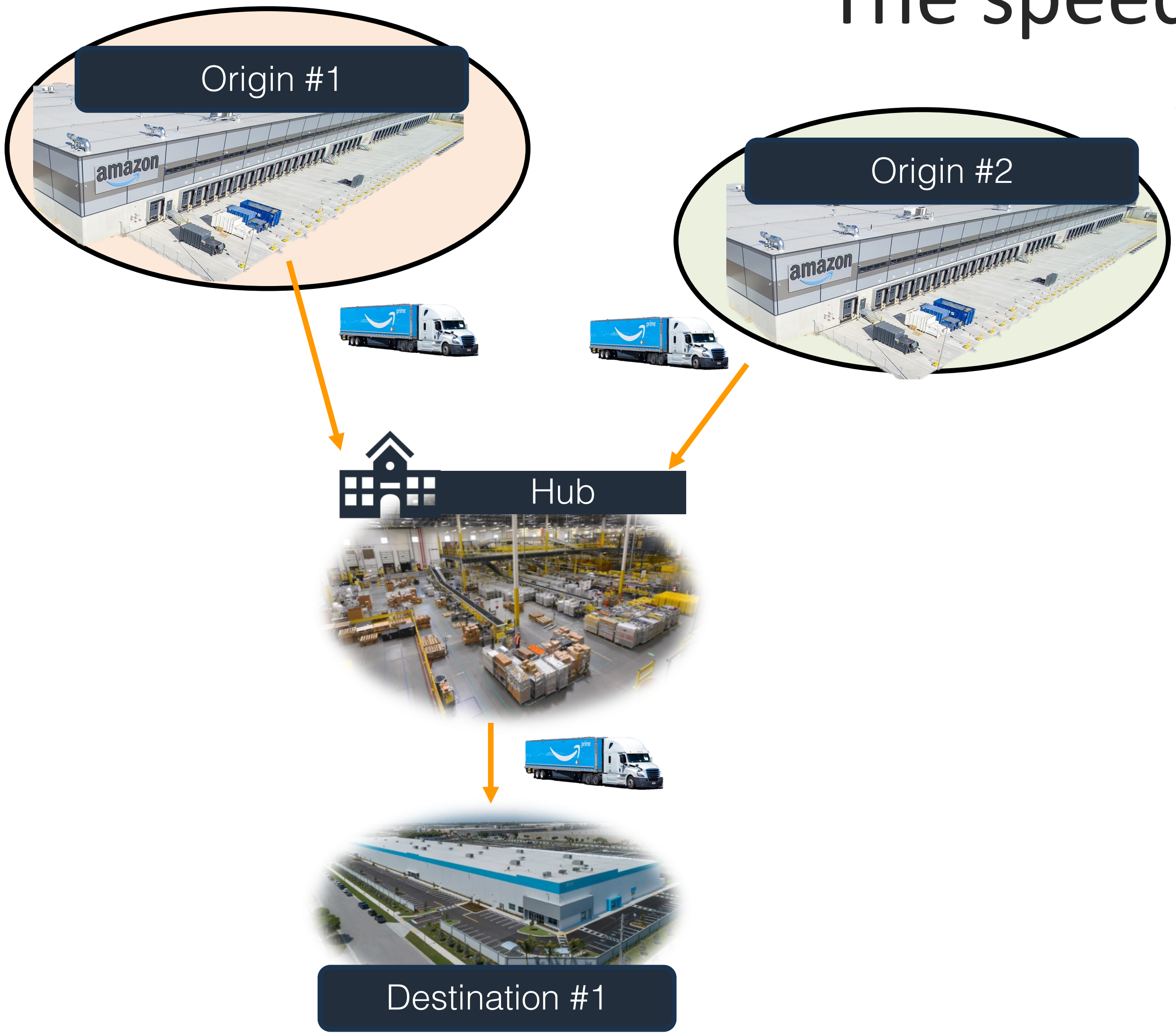
(c) Scale: Billions of variables to model hourly decisions, e.g., when a truck should depart.

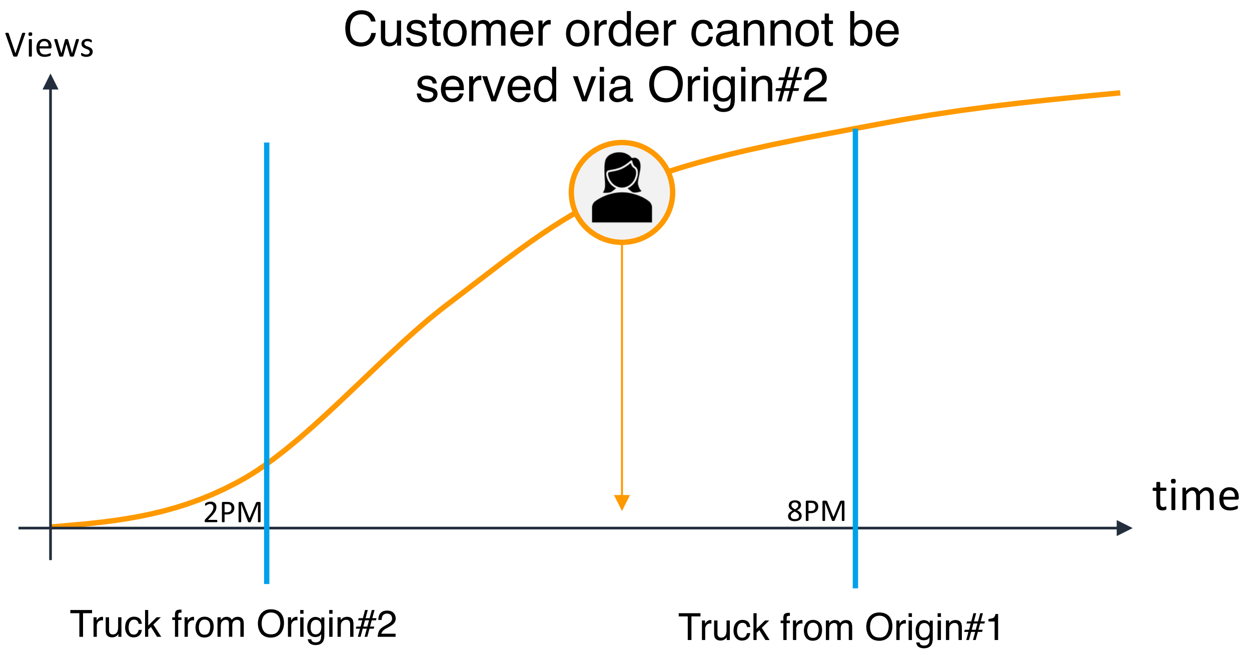
(d) Uncertainty: Customers' demand is uncertain, thus we should minimize the expected cost.

Outline

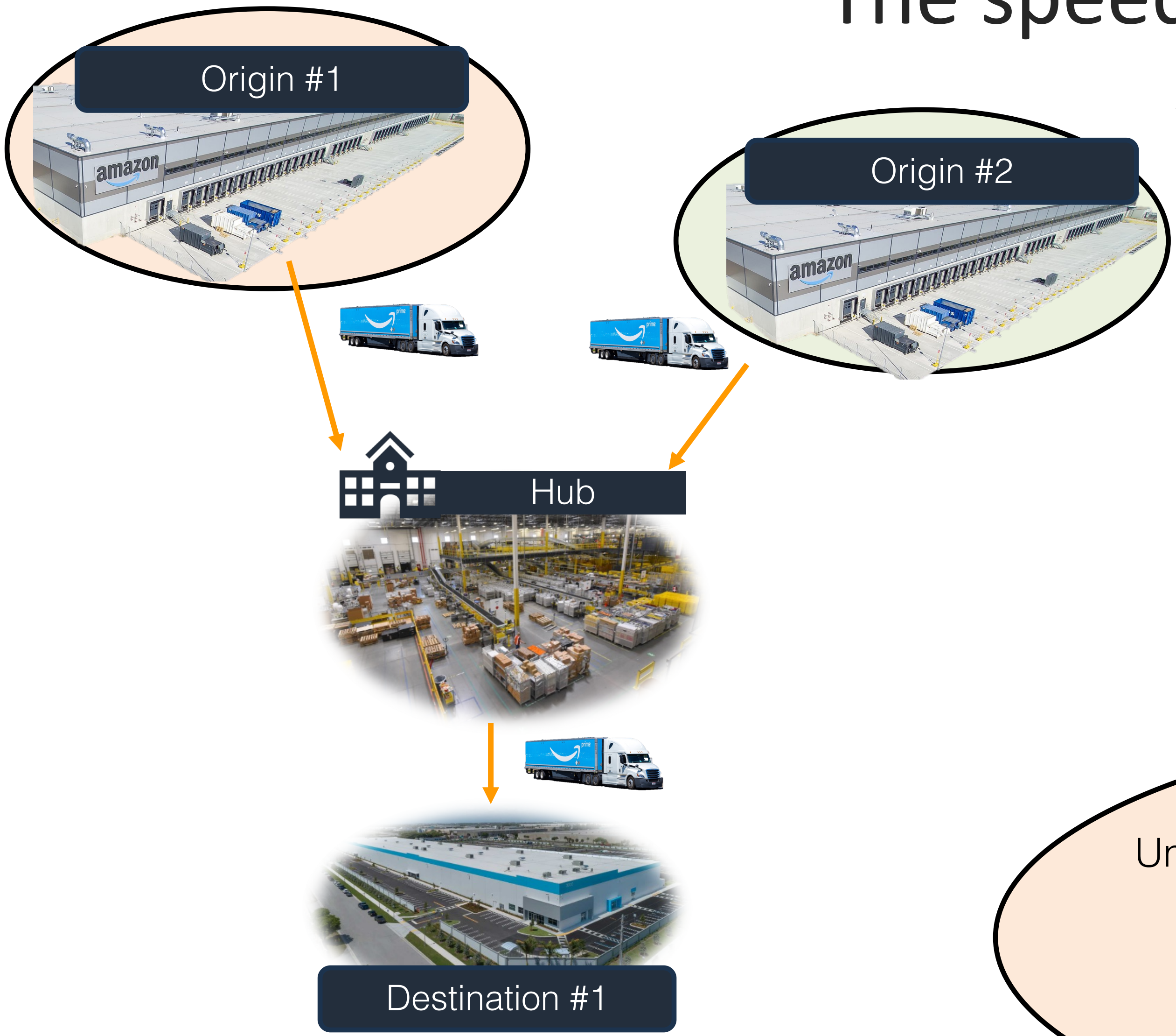
- Motivation
- Problem formulation
- Solution strategy
- Results

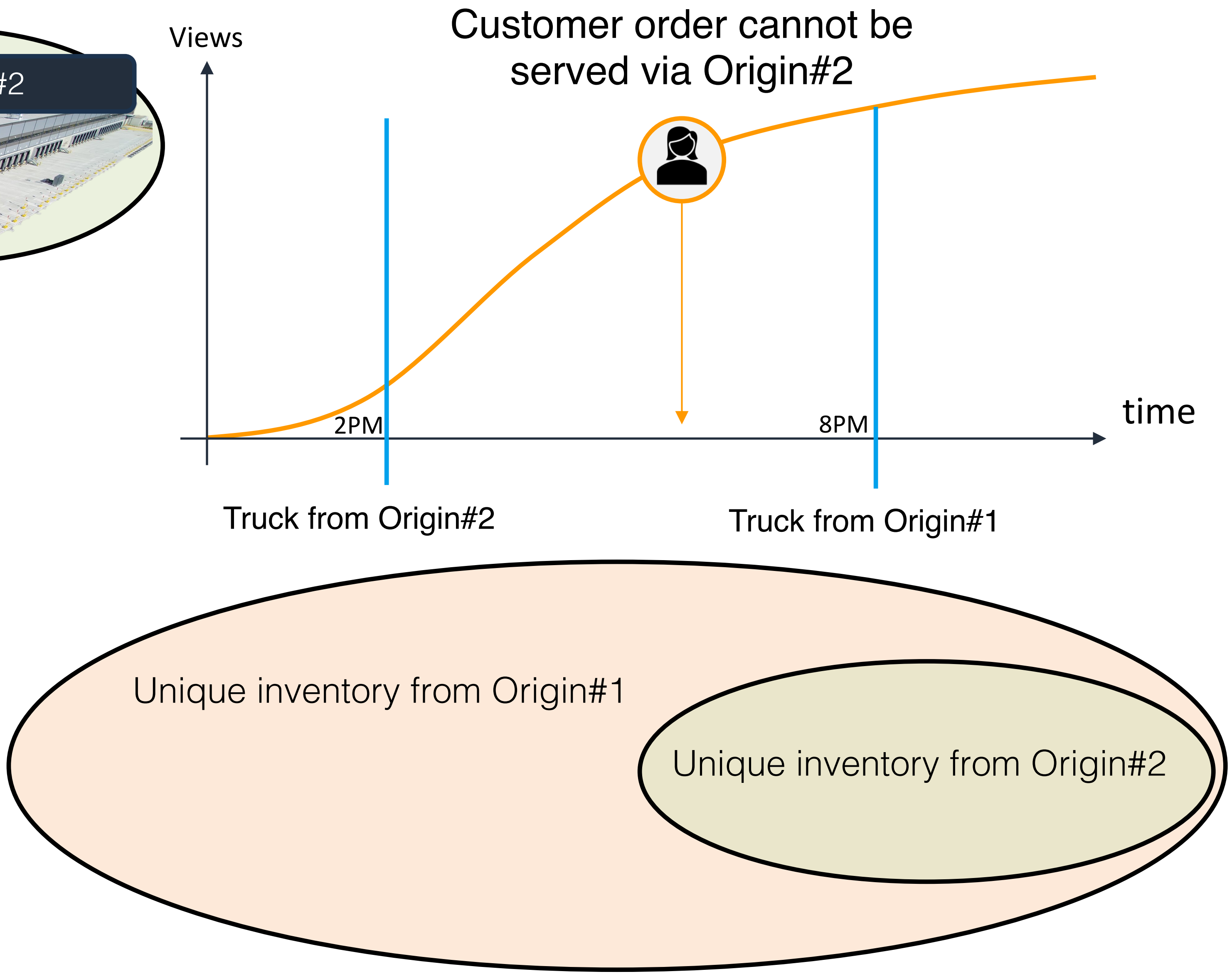
The speed objective





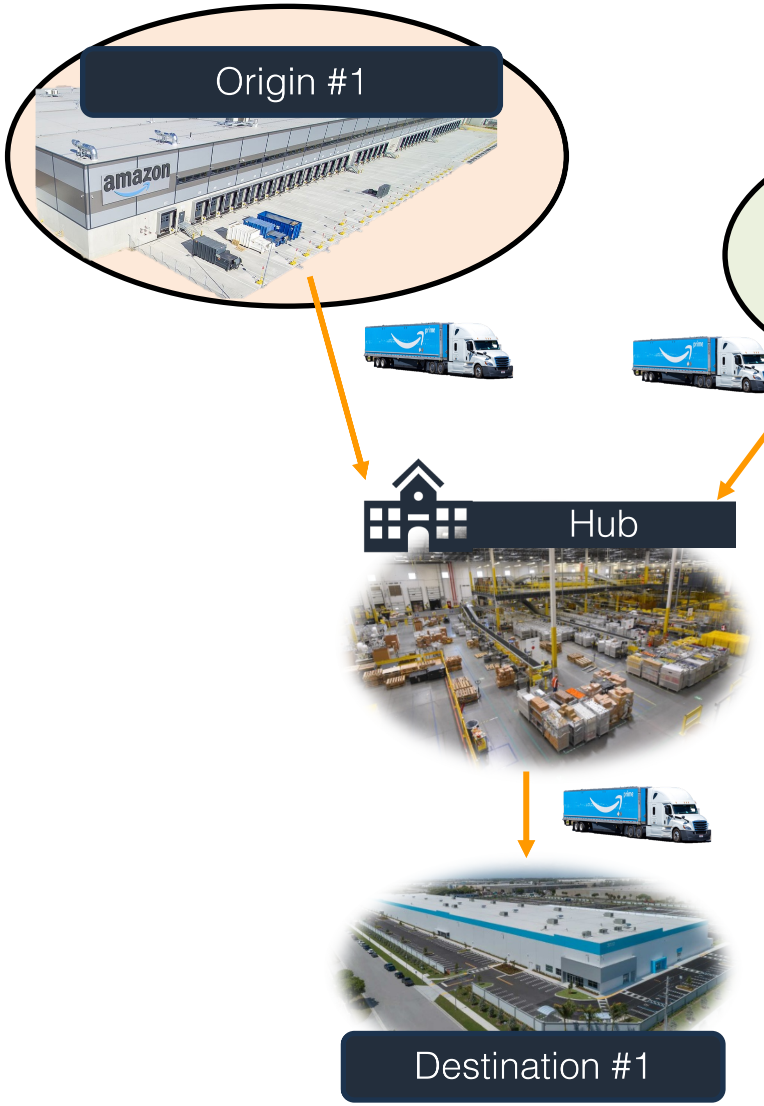
The speed objective

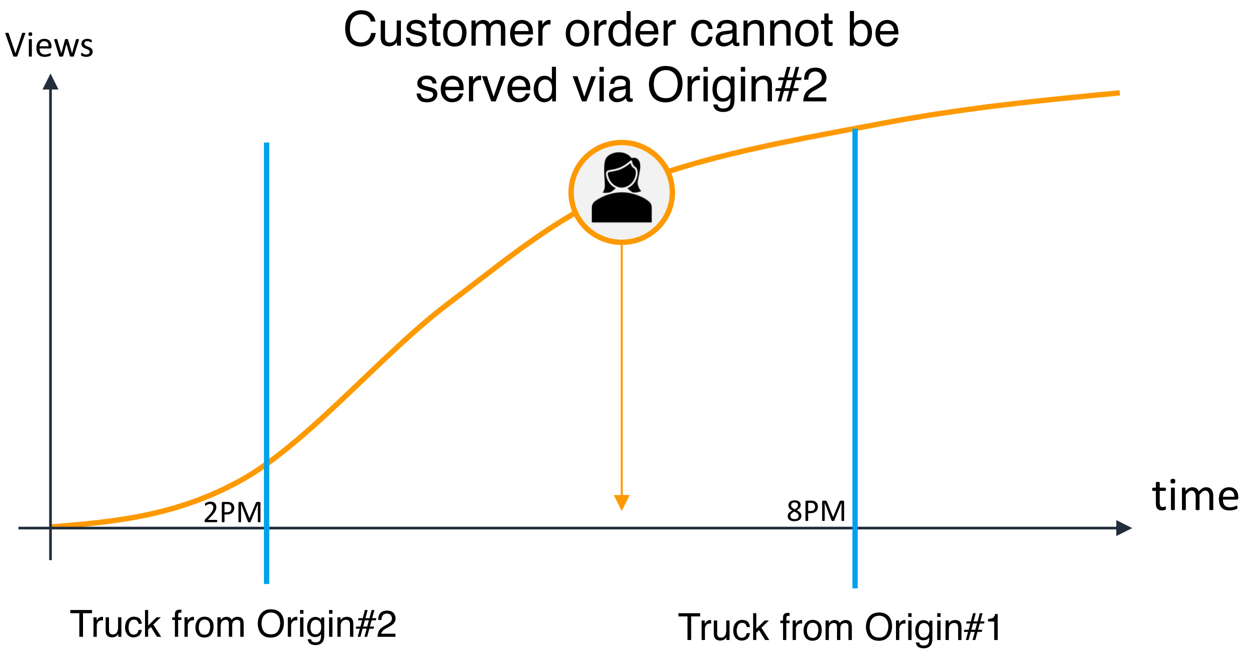


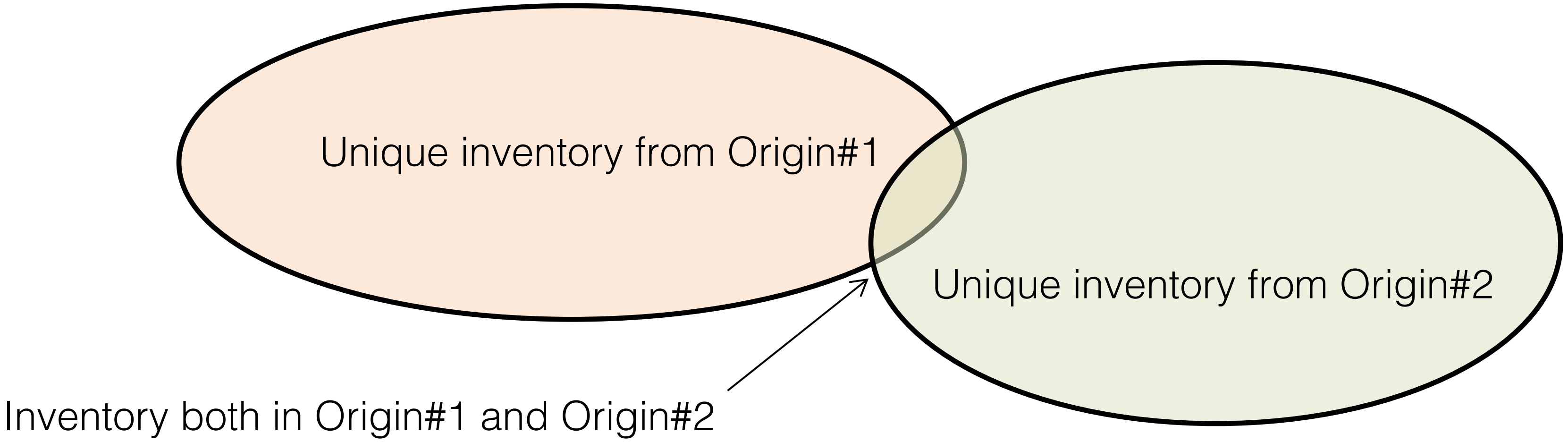


amazon | science

The speed objective

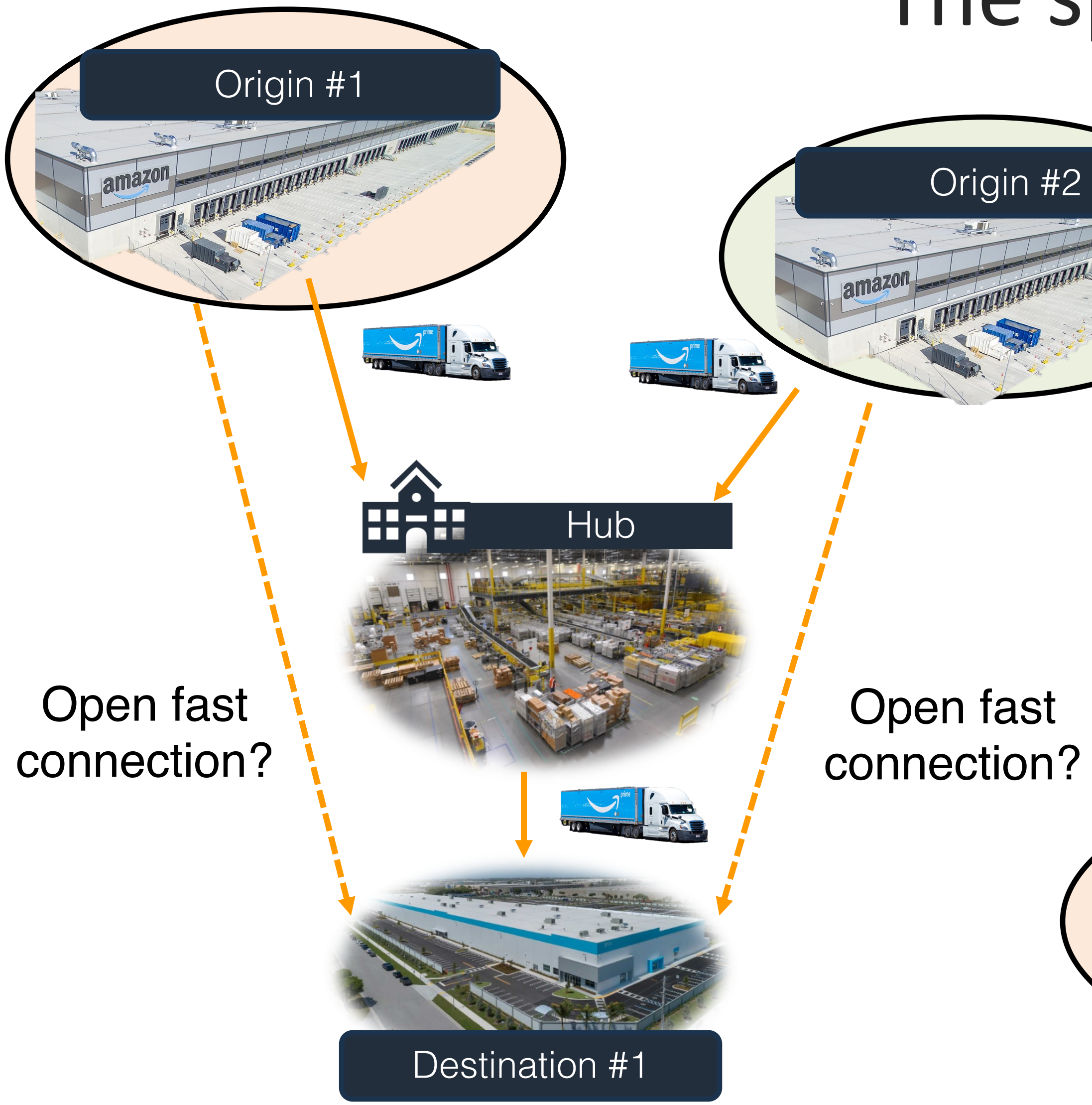


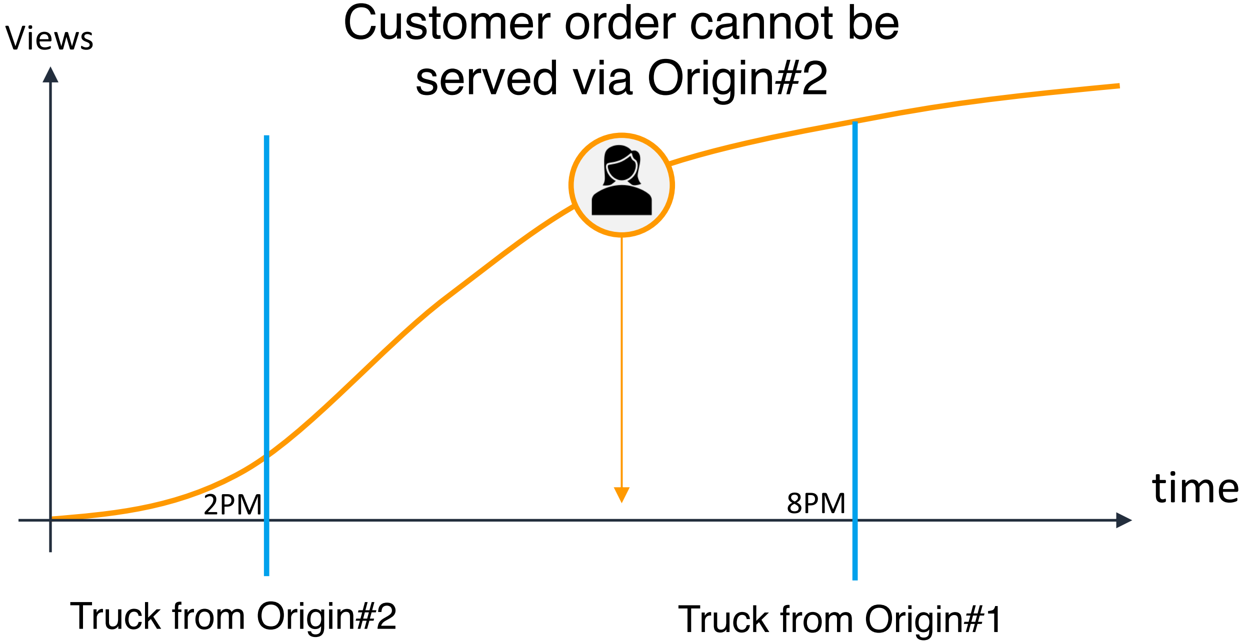


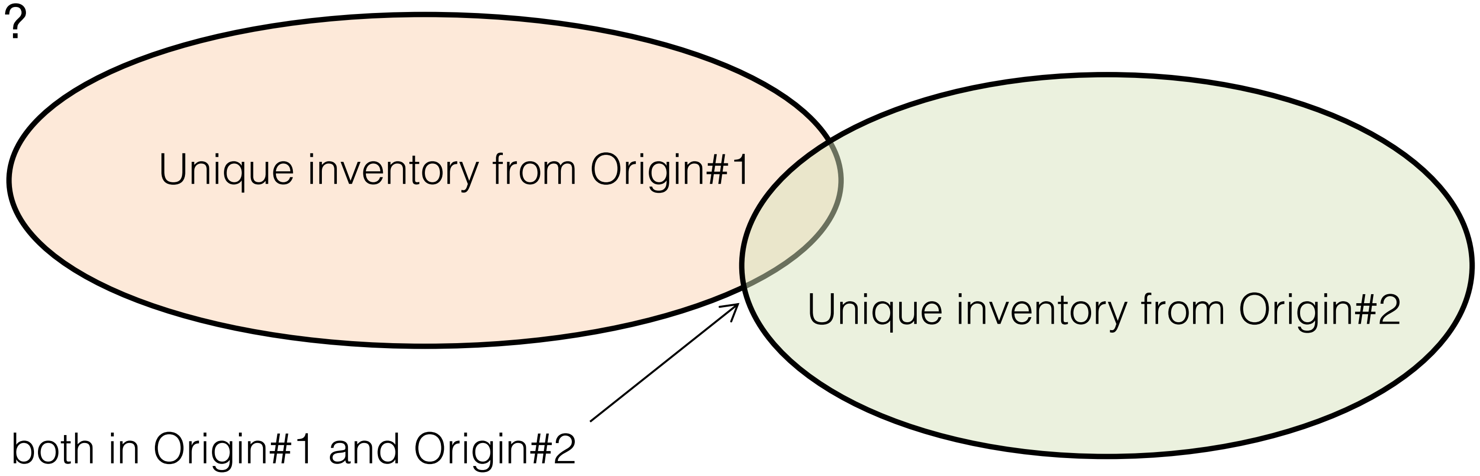


amazon | science

The speed objective



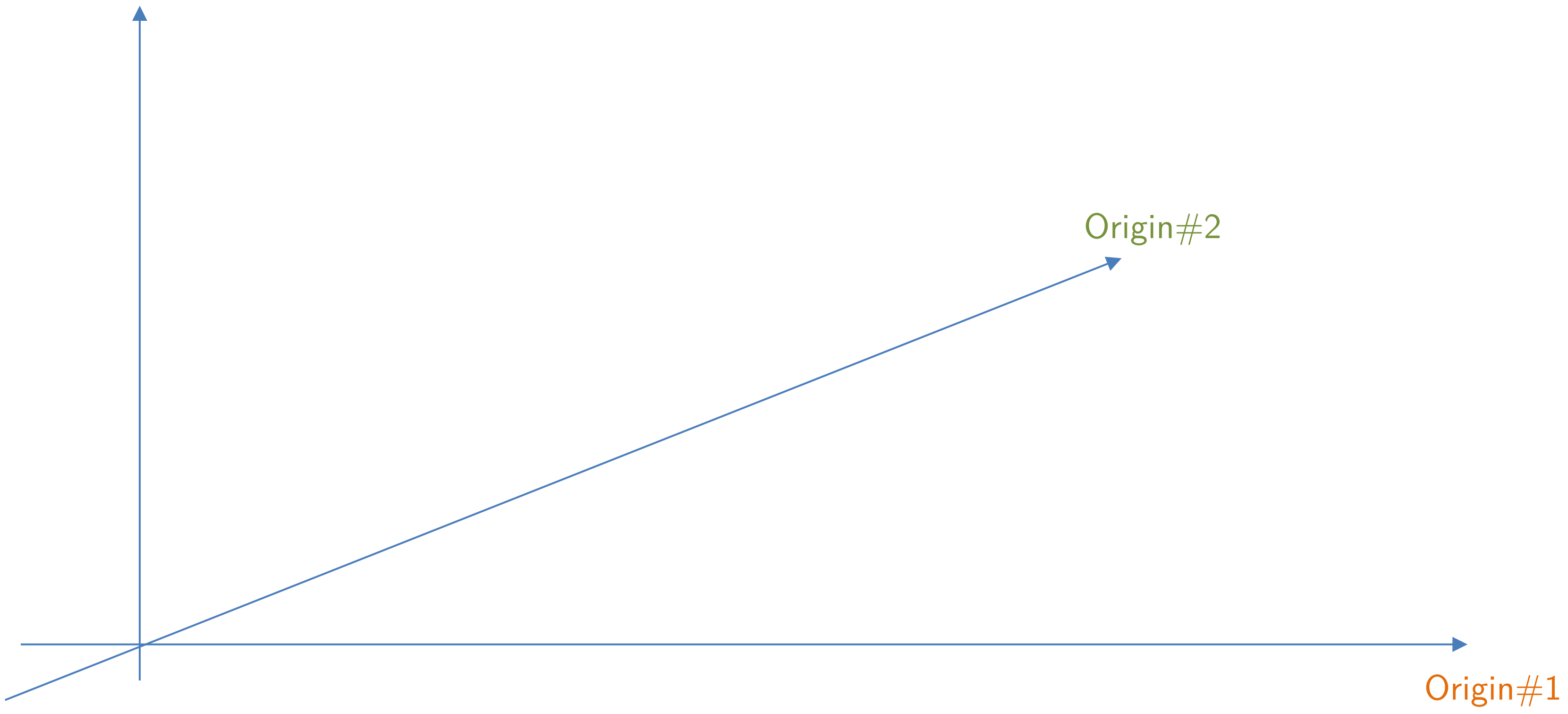




amazon | science

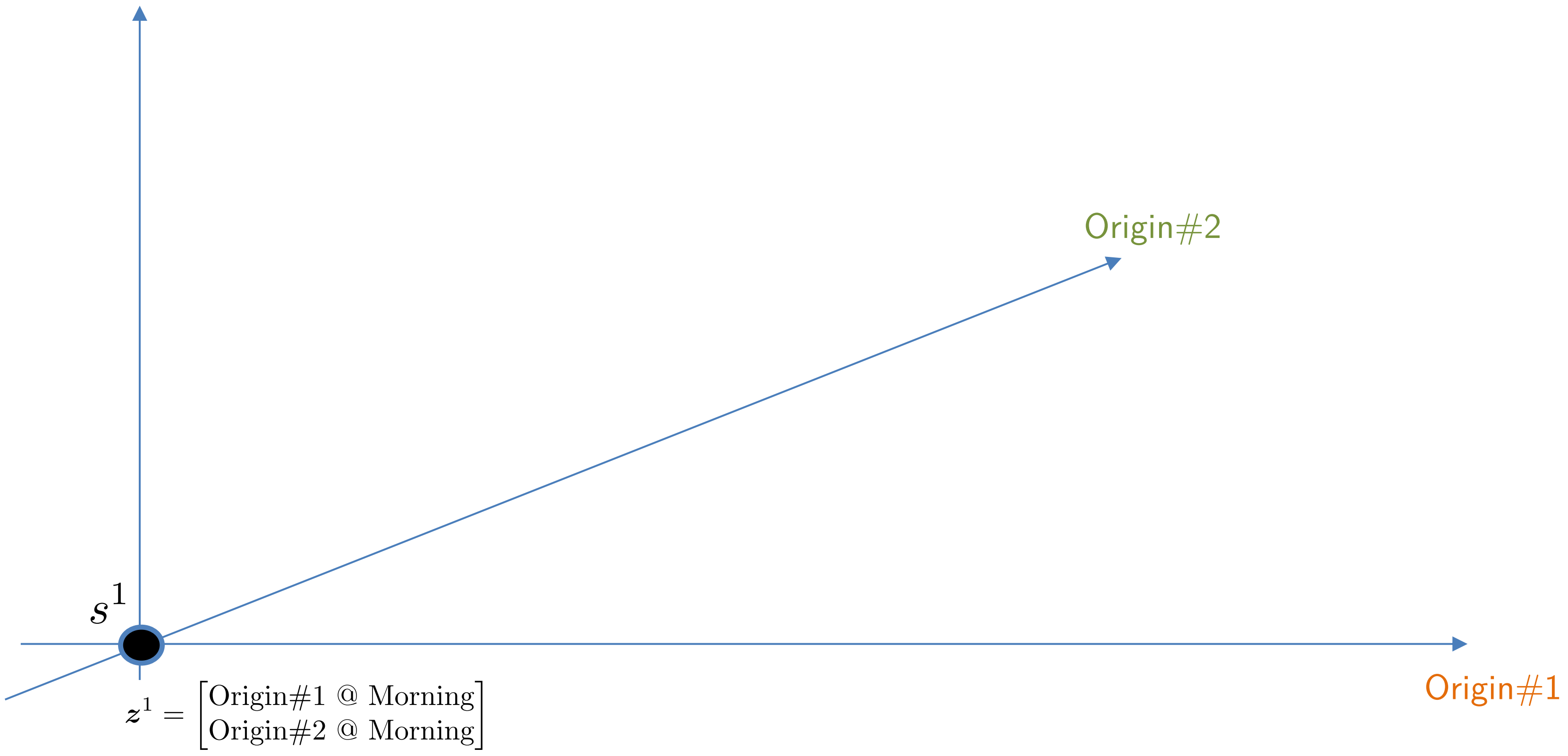
The speed objective

$\texttt{Speed}(z)$ = unique items that can be delivered in 1-day



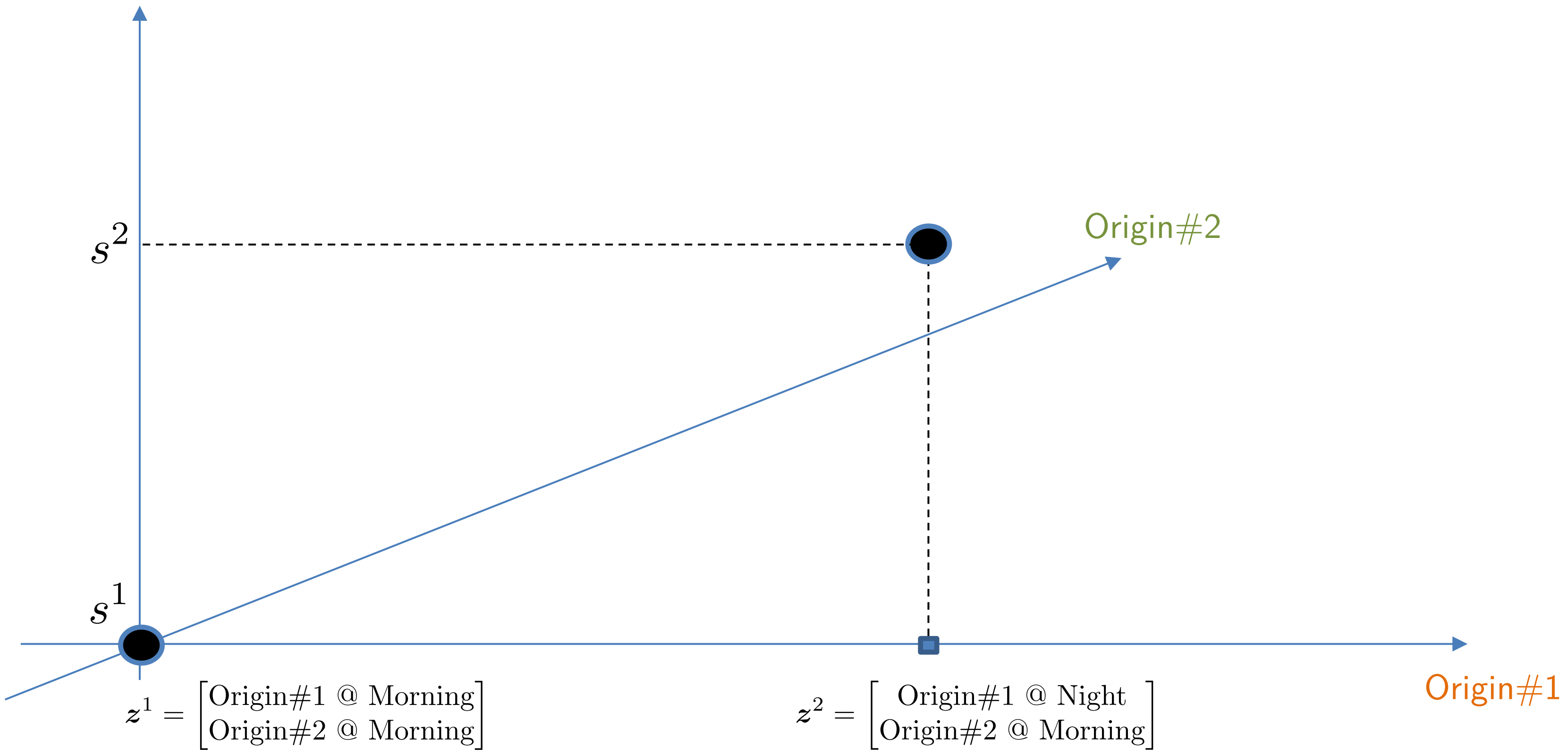
The speed objective

$\texttt{Speed}(z)$ = unique items that can be delivered in 1-day



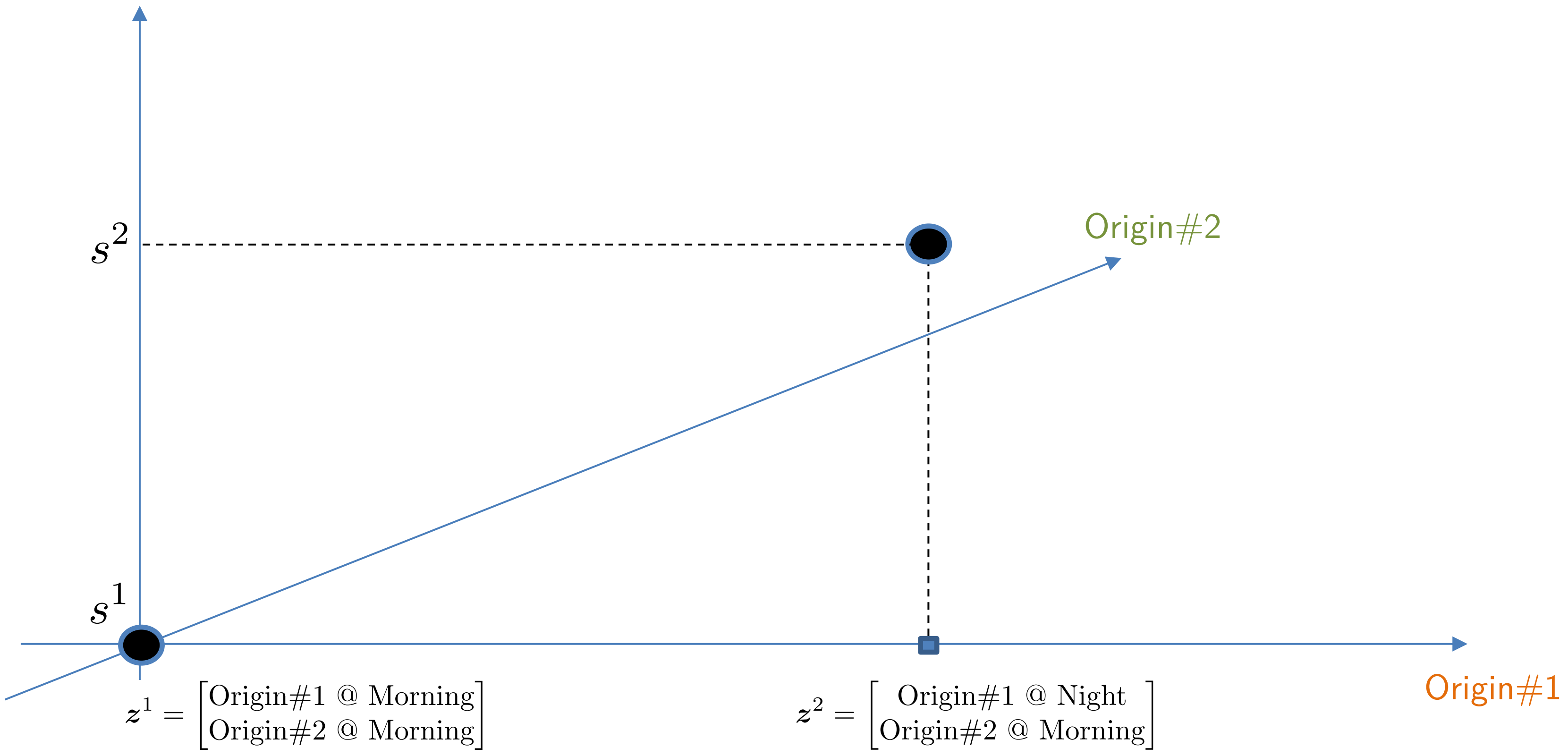
The speed objective

$\texttt{Speed}(z)$ = unique items that can be delivered in 1-day



The speed objective

$\texttt{Speed}(z)$ = unique items that can be delivered in 1-day

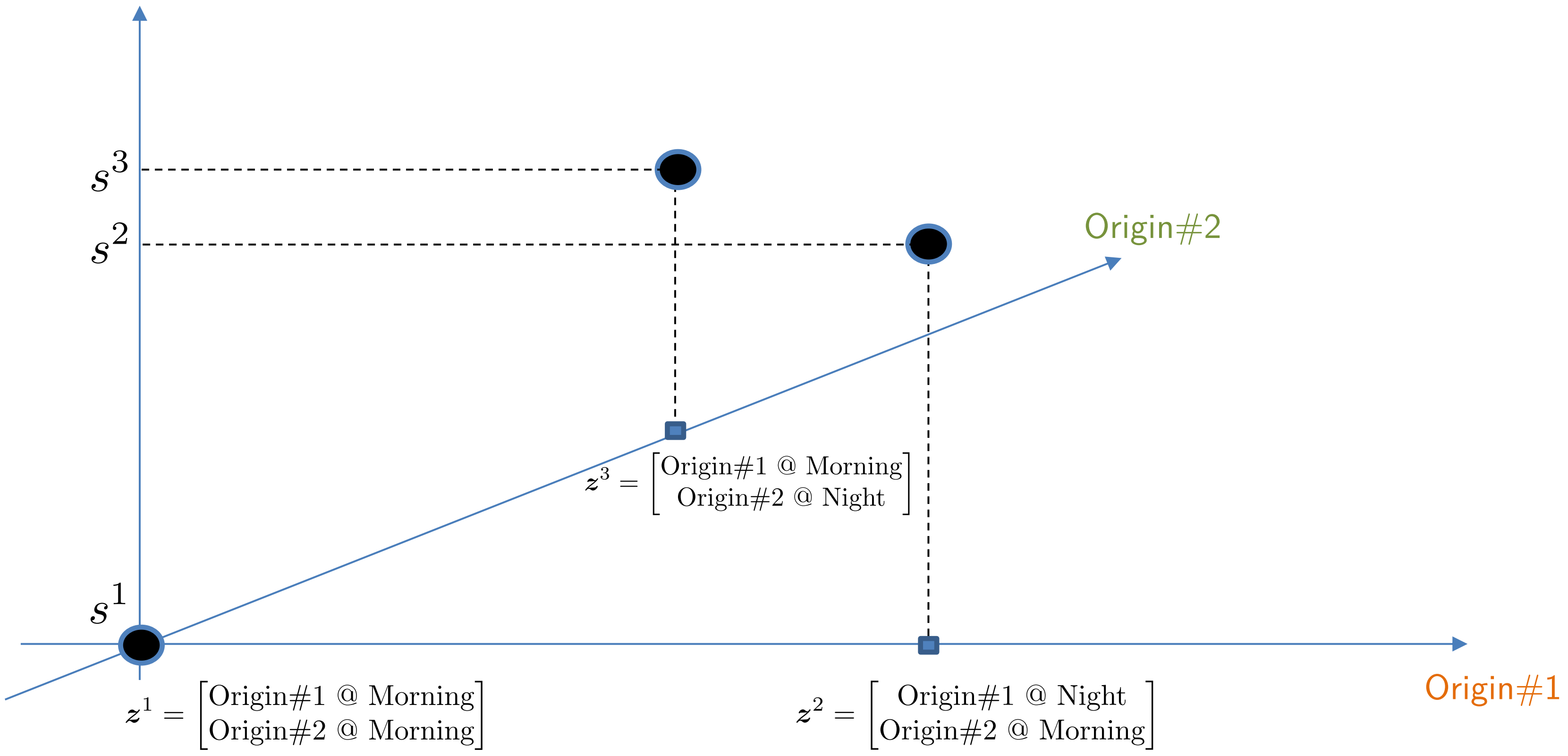


$\boxed{\texttt{Speed}(z) \text{ can be computed via a look-up table}}$

amazon | science

The speed objective

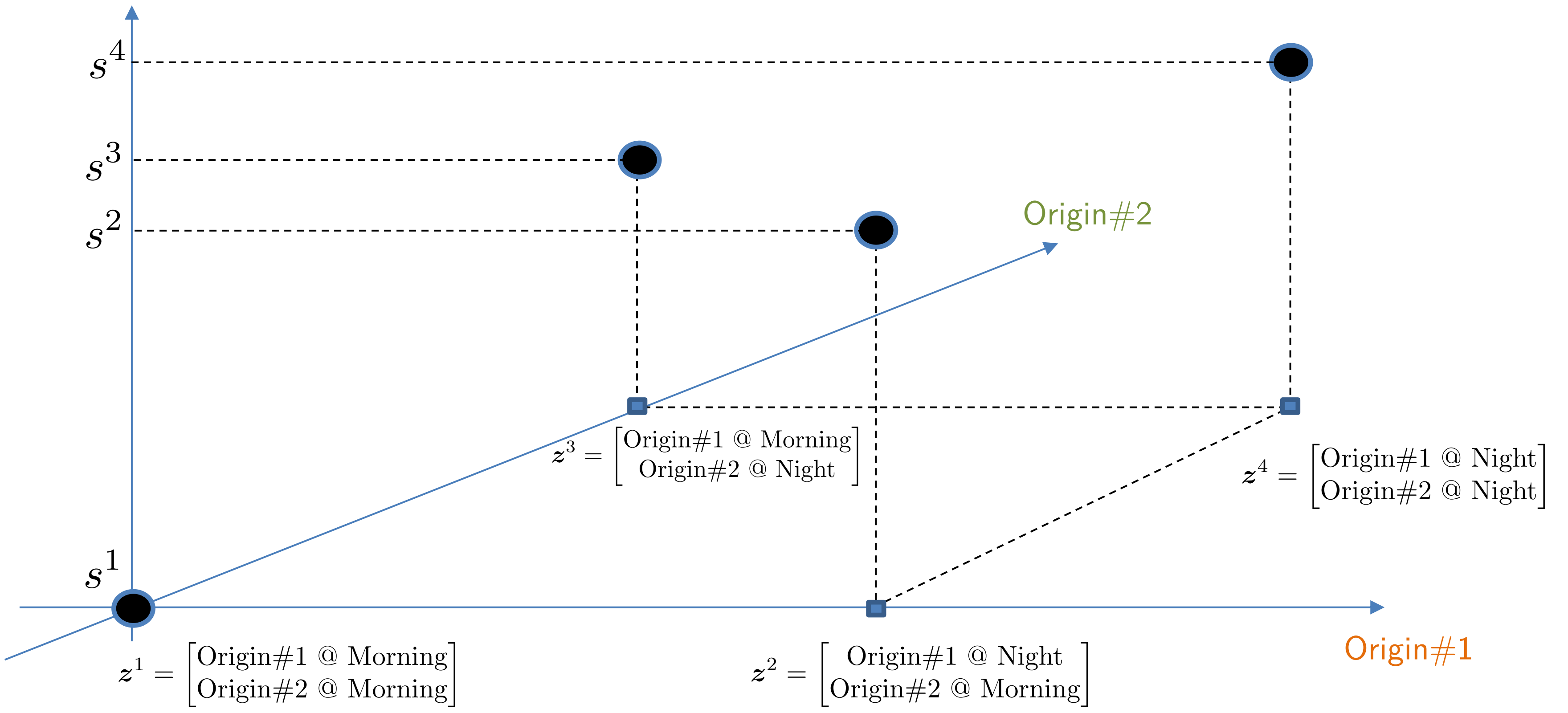
$\texttt{Speed}(z)$ = unique items that can be delivered in 1-day



$\texttt{Speed}(z)$ can be computed via a look-up table

The speed objective

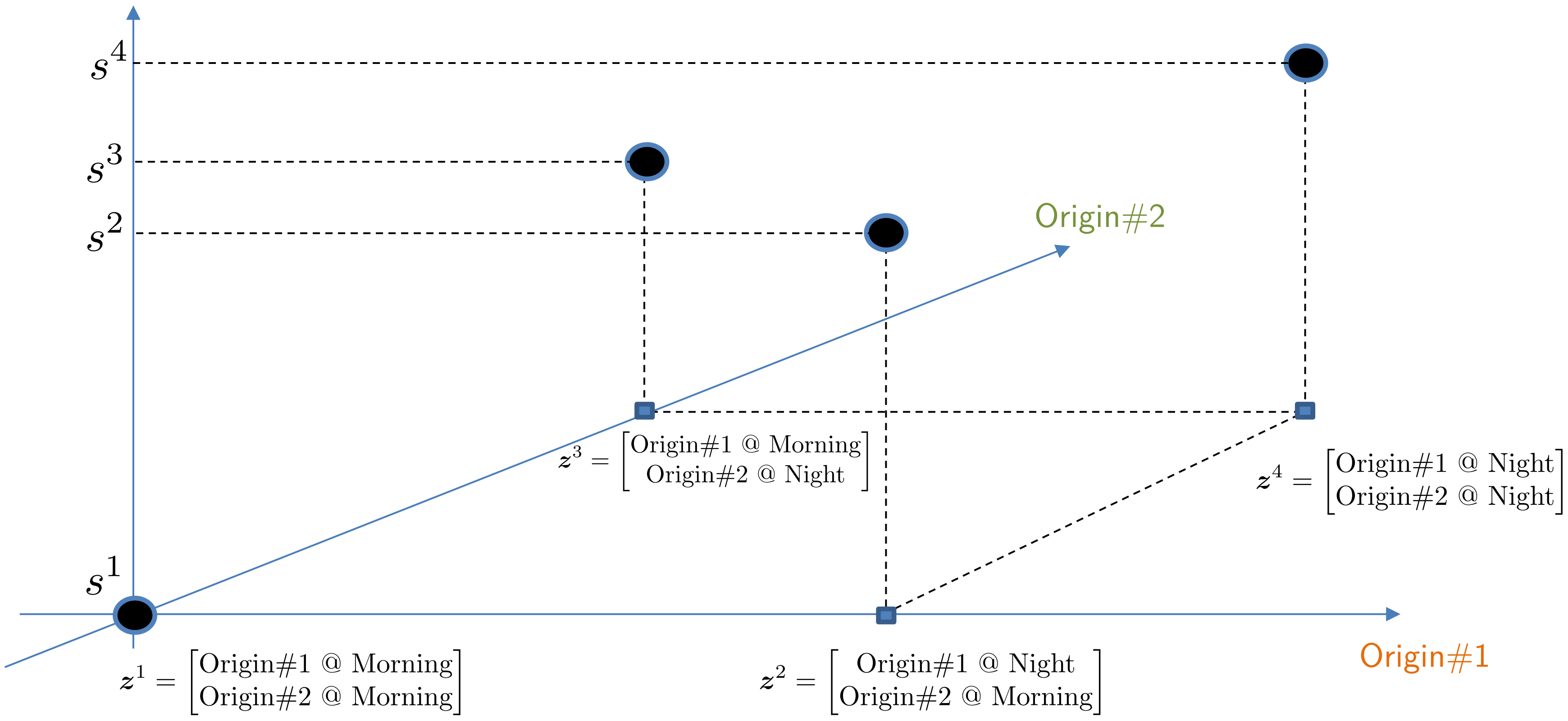
$\texttt{Speed}(z)$ = unique items that can be delivered in 1-day



$\texttt{Speed}(z)$ can be computed via a look-up table

The speed objective

$\texttt{Speed}(z)$ = unique items that can be delivered in 1-day



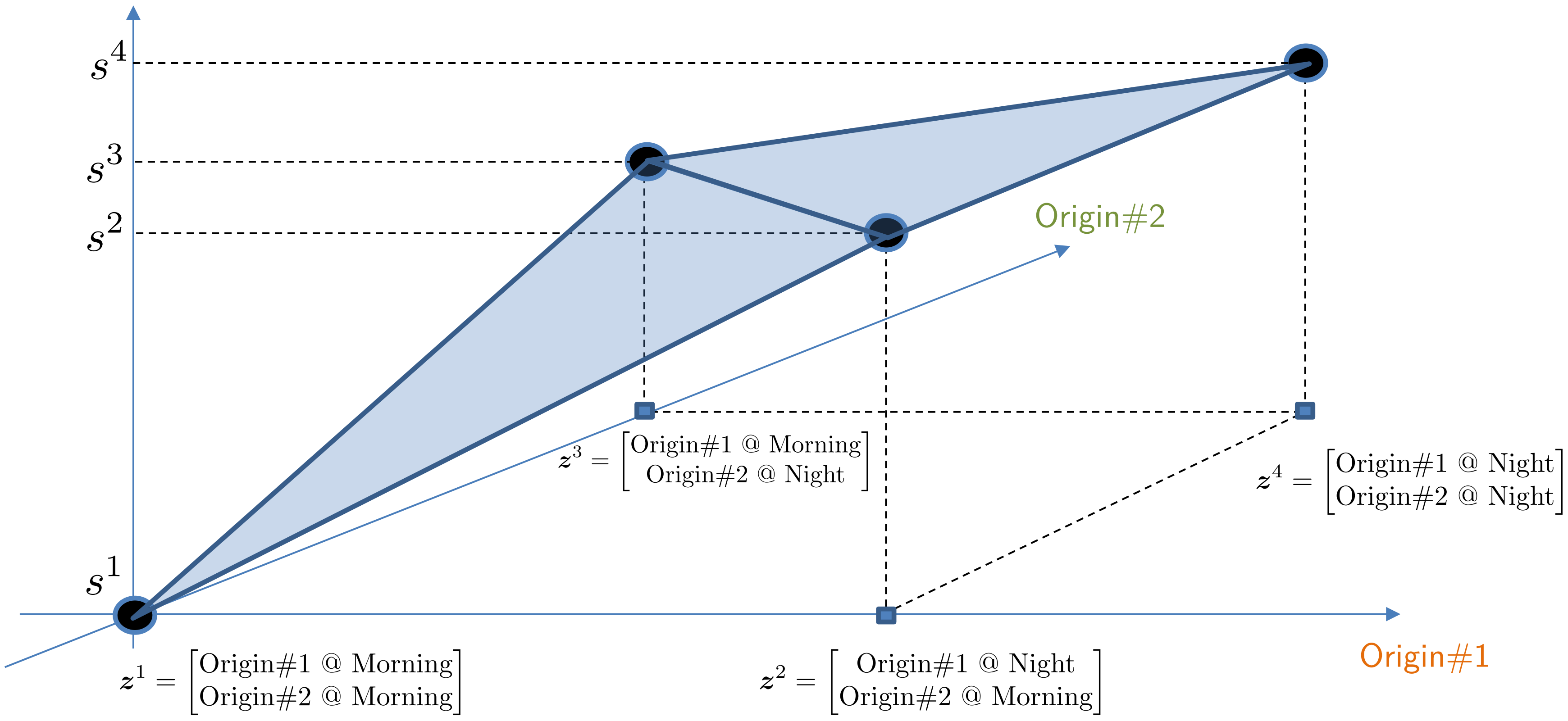
$\texttt{Speed}(z)$ can be computed via a look-up table

amazon | science

$$\texttt{Unique}(\text{Origin\#1}) + \texttt{Unique}(\text{Origin\#2}) \geq \texttt{Unique}(\text{Origin\#1} \cup \text{Origin\#2})$$

The speed objective

$\texttt{Speed}(z)$ = unique items that can be delivered in 1-day



$\texttt{Speed}(z)$ can be computed via a look-up table

$$\texttt{Unique}(\text{Origin\#1}) + \texttt{Unique}(\text{Origin\#2}) \geq \texttt{Unique}(\text{Origin\#1} \cup \text{Origin\#2})$$

Approximating the speed objective

Approximating the speed objective

Step 1: Consider truck departure time combinations:

$$\boldsymbol{z}^1 = \begin{bmatrix} \text{Origin\#1 @ Morning} \\ \text{Origin\#2 @ Morning} \end{bmatrix}, \boldsymbol{z}^2 = \begin{bmatrix} \text{Origin\#1 @ Night} \\ \text{Origin\#2 @ Morning} \end{bmatrix}, \boldsymbol{z}^3 = \begin{bmatrix} \text{Origin\#1 @ Morning} \\ \text{Origin\#2 @ Night} \end{bmatrix}, \boldsymbol{z}^4 = \begin{bmatrix} \text{Origin\#1 @ Night} \\ \text{Origin\#2 @ Night} \end{bmatrix}$$

amazon | science

Approximating the speed objective

Step 1: Consider truck departure time combinations:

$$\boldsymbol{z}^1 = \begin{bmatrix} \text{Origin\#1 @ Morning} \\ \text{Origin\#2 @ Morning} \end{bmatrix}, \boldsymbol{z}^2 = \begin{bmatrix} \text{Origin\#1 @ Night} \\ \text{Origin\#2 @ Morning} \end{bmatrix}, \boldsymbol{z}^3 = \begin{bmatrix} \text{Origin\#1 @ Morning} \\ \text{Origin\#2 @ Night} \end{bmatrix}, \boldsymbol{z}^4 = \begin{bmatrix} \text{Origin\#1 @ Night} \\ \text{Origin\#2 @ Night} \end{bmatrix}$$

Step 2: Compute speed for each combination:

$$s^i = \texttt{Speed}(\boldsymbol{z}^i)$$

amazon | science

Approximating the speed objective

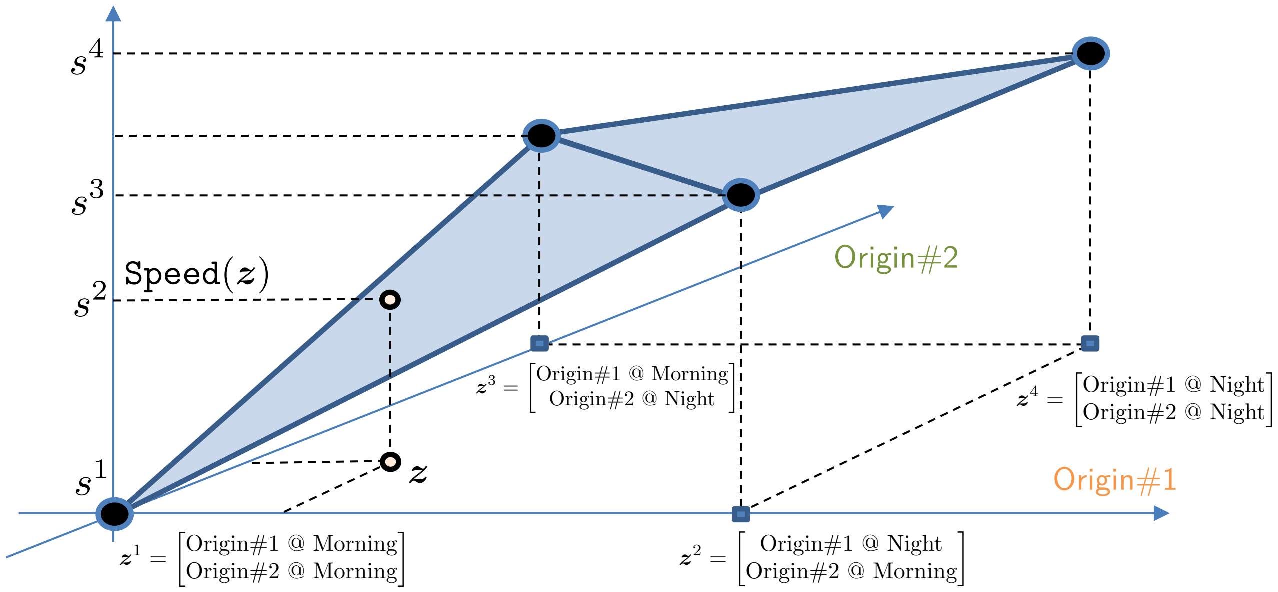
Step 1: Consider truck departure time combinations:

$$\boldsymbol{z}^1 = \begin{bmatrix}\text{Origin\#1 @ Morning}\\ \text{Origin\#2 @ Morning}\end{bmatrix}, \boldsymbol{z}^2 = \begin{bmatrix}\text{Origin\#1 @ Night}\\ \text{Origin\#2 @ Morning}\end{bmatrix}, \boldsymbol{z}^3 = \begin{bmatrix}\text{Origin\#1 @ Morning}\\ \text{Origin\#2 @ Night}\end{bmatrix}, \boldsymbol{z}^4 = \begin{bmatrix}\text{Origin\#1 @ Night}\\ \text{Origin\#2 @ Night}\end{bmatrix}$$

Step 2: Compute speed for each combination:

$$s^i = \texttt{Speed}(\boldsymbol{z}^i)$$

Step 3: Interpolation via parametric optimization:



amazon | science

Approximating the speed objective

Step 1: Consider truck departure time combinations:

$$\boldsymbol{z}^1 = \begin{bmatrix}\text{Origin\#1 @ Morning}\\ \text{Origin\#2 @ Morning}\end{bmatrix}, \boldsymbol{z}^2 = \begin{bmatrix}\text{Origin\#1 @ Night}\\ \text{Origin\#2 @ Morning}\end{bmatrix}, \boldsymbol{z}^3 = \begin{bmatrix}\text{Origin\#1 @ Morning}\\ \text{Origin\#2 @ Night}\end{bmatrix}, \boldsymbol{z}^4 = \begin{bmatrix}\text{Origin\#1 @ Night}\\ \text{Origin\#2 @ Night}\end{bmatrix}$$

Step 2: Compute speed for each combination:

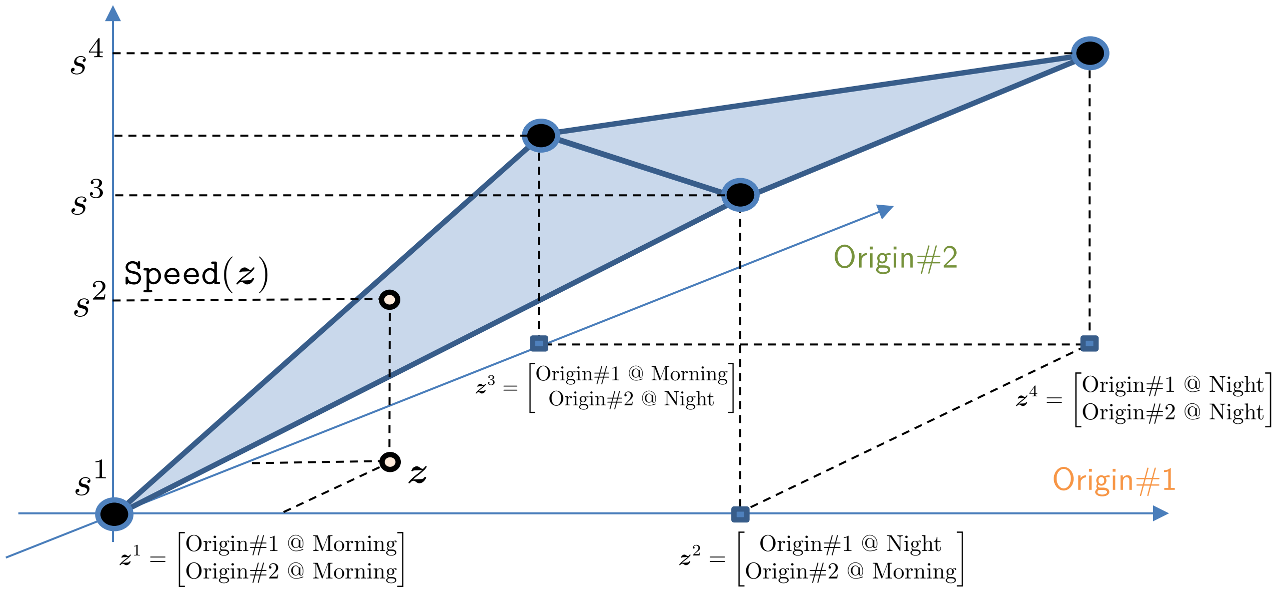
$$s^i = \texttt{Speed}(\boldsymbol{z}^i)$$

Step 3: Interpolation via parametric optimization:

$$\texttt{Speed}(\boldsymbol{z}) = \max_{\boldsymbol{\alpha} \geq 0} \quad \sum_i \alpha^i s^i$$

$$\text{subject to} \quad \sum_i \alpha^i \boldsymbol{z}^i = \boldsymbol{z}$$

$$\sum_i \alpha_d^i = 1$$



amazon | science

Approximating the speed objective

Step 1: Consider truck departure time combinations:

$$\boldsymbol{z}^1 = \begin{bmatrix} \text{Origin\#1 @ Morning} \\ \text{Origin\#2 @ Morning} \end{bmatrix}, \boldsymbol{z}^2 = \begin{bmatrix} \text{Origin\#1 @ Night} \\ \text{Origin\#2 @ Morning} \end{bmatrix}, \boldsymbol{z}^3 = \begin{bmatrix} \text{Origin\#1 @ Morning} \\ \text{Origin\#2 @ Night} \end{bmatrix}, \boldsymbol{z}^4 = \begin{bmatrix} \text{Origin\#1 @ Night} \\ \text{Origin\#2 @ Night} \end{bmatrix}$$

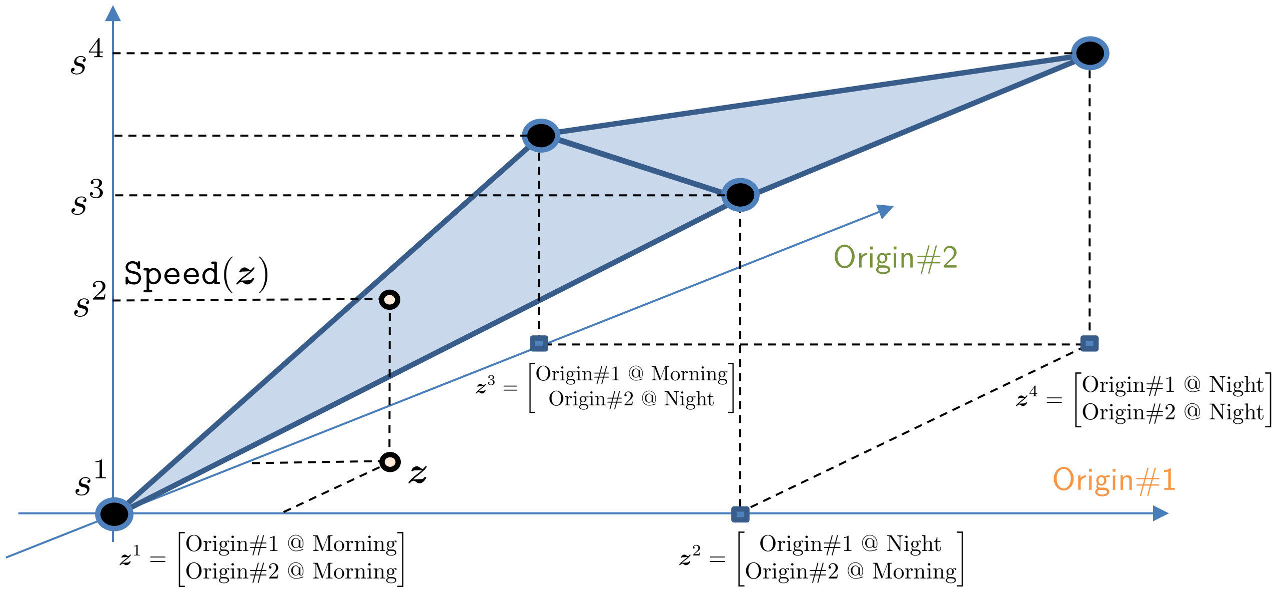
Step 2: Compute speed for each combination:

$$s^i = \texttt{Speed}(\boldsymbol{z}^i)$$

Step 3: Interpolation via parametric optimization:

$$\begin{aligned} \texttt{Speed}(\boldsymbol{z}) = \max_{\boldsymbol{\alpha} \geq 0} \quad & \sum_i \alpha^i s^i \\ \text{subject to} \quad & \sum_i \alpha^i \boldsymbol{z}^i = \boldsymbol{z} \\ & \sum_i \alpha_d^i = 1 \end{aligned}$$

Parameter defining the optimal value of the LP



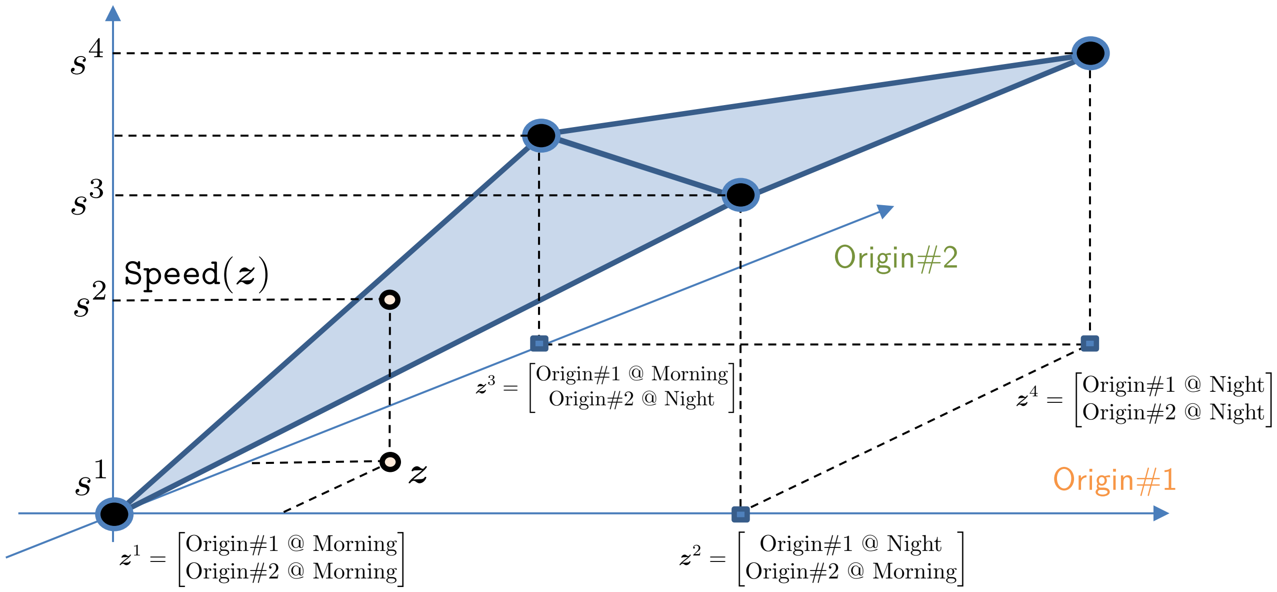
amazon | science

Approximating the speed objective

Step 3: Interpolation via parametric optimization:

$$\begin{aligned}\texttt{Speed}(z) = \max_{\boldsymbol{\alpha}\geq 0} \quad & \sum_i \alpha^i s^i \\ \text{subject to} \quad & \sum_i \alpha^i \boldsymbol{z}^i = z \\ & \sum_i \alpha^i_d = 1\end{aligned}$$

Parameter defining the optimal value of the LP



amazon | science

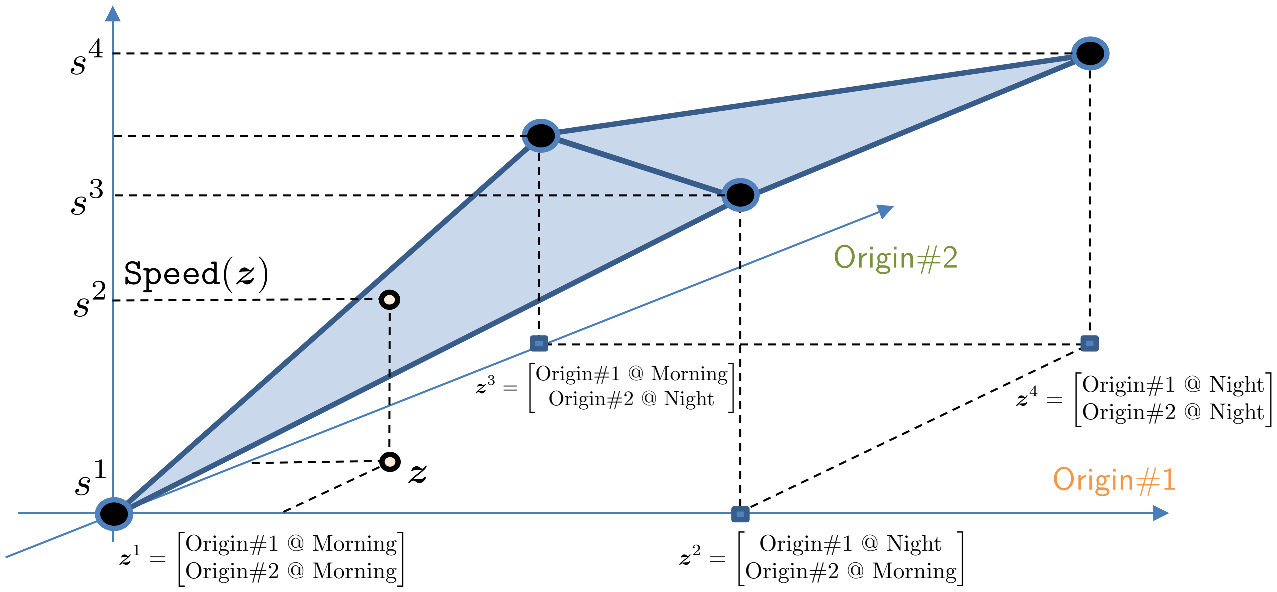
Approximating the speed objective

$$
\begin{aligned}
\min_{\boldsymbol{p},\boldsymbol{y},\boldsymbol{z}} \quad & \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y}) - \texttt{Speed}(\boldsymbol{z}) \\
\text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork} \\
& \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p})
\end{aligned}
$$

Step 3: Interpolation via parametric optimization:

$$
\begin{aligned}
\texttt{Speed}(\boldsymbol{z}) = \max_{\boldsymbol{\alpha}\geq 0} \quad & \sum_i \alpha^i s^i \\
\text{subject to} \quad & \sum_i \alpha^i \boldsymbol{z}^i = \boldsymbol{z} \\
& \sum_i \alpha^i_d = 1
\end{aligned}
$$

Parameter defining the optimal value of the LP



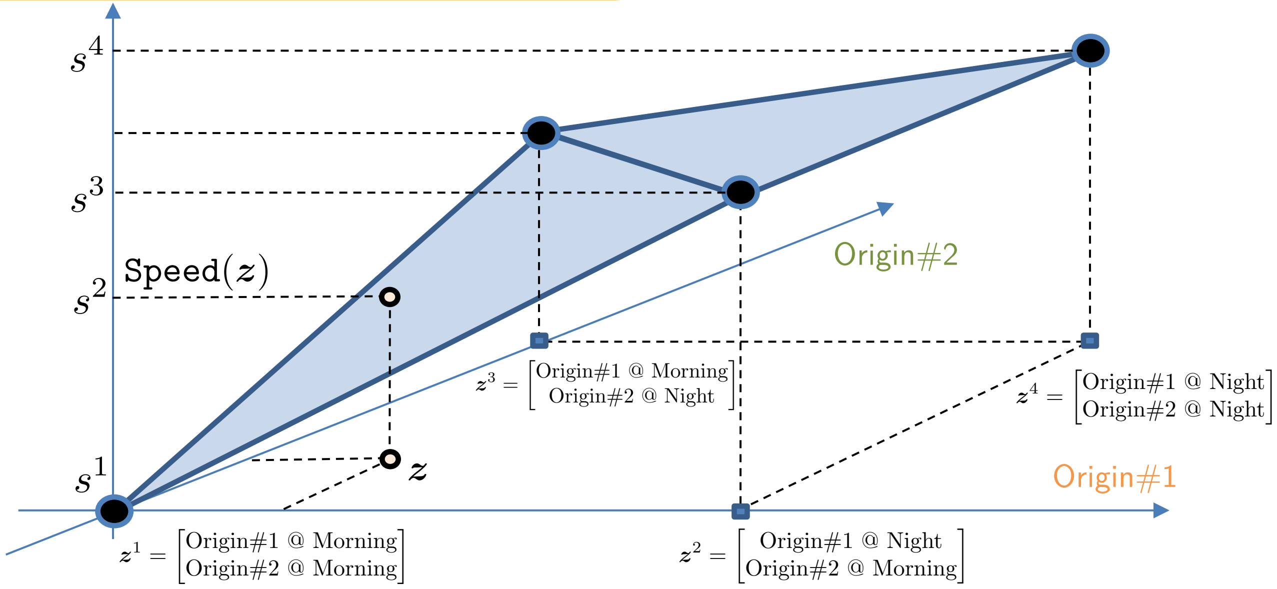
Approximating the speed objective

$$\begin{aligned} \min_{\boldsymbol{p},\boldsymbol{y},\boldsymbol{z},\boldsymbol{\alpha}>0} \quad & \texttt{NetworkCost}(\boldsymbol{p},\boldsymbol{y}) - \textstyle\sum_i \alpha^i s^i \\ \text{s.t.} \quad & (\boldsymbol{p},\boldsymbol{y}) \in \texttt{FeasibleNetwork}, \\ & \boldsymbol{z} \in \texttt{FeasibleSchedule}(\boldsymbol{p}) \\ & \textstyle\sum_i \alpha^i \boldsymbol{z}^i = \boldsymbol{z}, \textstyle\sum_i \alpha^i = 1 \end{aligned}$$

Step 3: Interpolation via parametric optimization:

$$\begin{aligned} \texttt{Speed}(\boxed{z}) = \max_{\boldsymbol{\alpha}\geq 0} \quad & \sum_i \alpha^i s^i \\ \text{subject to} \quad & \sum_i \alpha^i \boldsymbol{z}^i = \boxed{z} \\ & \sum_i \alpha_d^i = 1 \end{aligned}$$

Parameter defining the optimal value of the LP



Outline

- Motivation
- Problem formulation
- Solution strategy
- Results

Results on a randomly generated datasets

Results on a randomly generated datasets

Results on 4 randomly generated datasets

- Location of nodes generated at random (cost of operating a truck proportional to distance)
- Items stored in each warehouse randomly generated
- Assuming a 0.1 conversion factor from speed and cost

Results on a randomly generated datasets

Results on 4 randomly generated datasets

- Location of nodes generated at random (cost of operating a truck proportional to distance)
- Items stored in each warehouse randomly generated
- Assuming a 0.1 conversion factor from speed and cost

Speed parametric approximation compute by subsampling datapoint

Algorithm 1: Unique Items Approximation

```
Inputs: κ, D, F;
for d ∈ D do
    Initialize Λ_d = ∅;
    Sort by number of unique item the FCs that can offer 1-day delivery to DS d;
    Compute the set Z_d of all possible combinations of speed lane assignments for top κ FCs;
    For all z_d^i ∈ Z_d, add the tuple (z_d^i, U_d(z_d^i)) to the set Λ_d;
    Compute the speed lane assignments from (12) for remaining (n_f − κ) FCs;
    For all z̃_d^i ∈ Z̃_d, add the tuple (z̃_d^i, U_d(z̃_d^i)) to the set Λ_d;
Return: Set of vectors and cost coefficients Λ_d for d ∈ D.
```

Take all combination for top κ origins

Results on a randomly generated datasets

Baseline method only considering cost

n_F	n_D	Costs	App. Rev.	Rev.	Cost - (App. Rev.)	Cost - Rev.	κ	%Gap
10	10	689.1	-	342.8	689.1	346.3	-	0
10	10	698.5	383.7	383.7	314.8	**314.8**	1	0
10	10	698.5	383.7	383.7	314.8	**314.8**	5	0
10	10	698.5	383.7	383.7	314.8	**314.8**	10	0
10	20	1419.2	-	370.7	1419.2	677.7	-	1.8
10	20	1459.7	405.9	407	647.9	645.7	1	3.4
10	20	1434.1	397.8	400.7	638.5	632.7	5	2.8
10	20	1435.5	403.5	403.5	628.4	628.4	10	2.5
20	10	1392.2	-	770.4	1392.2	621.8	-	0.1
20	10	1422.9	807.3	815.7	615.6	607.2	1	0.1
20	10	1422.9	807.3	815.7	615.6	607.2	5	0.1
20	10	1413.6	806.5	811.2	607.1	**602.4**	10	0.1
50	10	3375.8	-	1845.2	3375.8	1530.6	-	1.8
50	10	3528.2	2010	2032.2	1518.2	1496.0	1	7.5
50	10	3499.0	2031.5	2039.1	1467.5	1459.9	5	3.8
50	10	3489.8	2025.3	2037.3	1464.4	**1452.4**	10	3.6
100	100	62204.2	-	3651.1	62204.2	25693.2	-	2.8
100	100	65911.4	4159.3	4175.7	24317.9	24154.2	1	11.2
100	100	65531.7	4156.8	4173.1	23963.2	23800.5	5	9.5
100	100	65258.6	4154.3	4172.1	23714.9	**23537.2**	10	8.4

8.36% improvement over baseline

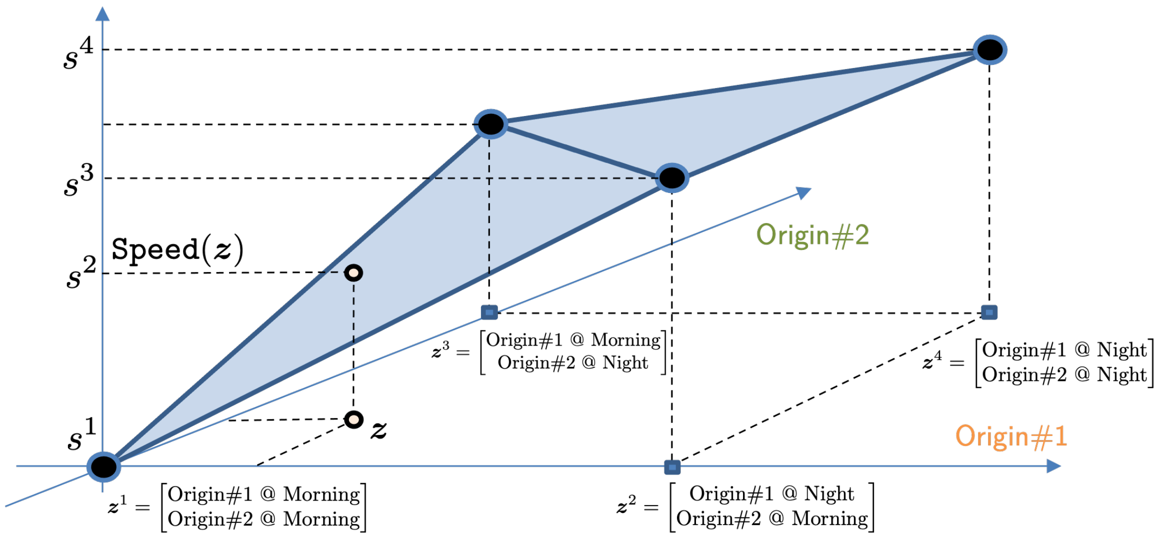
Summary

Problem

- Speed and cost should be jointly optimised
- Speed objective is submodular and can be evaluated with a data query

Solution

- Leverage parametric optimization
- Subsampling strategy to reduce complexity



Results

- 8.36% speed and cost benefits compared to baseline
- No additional computational cost compared to baseline