Vehicle Routing & Scheduling

- Multiple Routes
- Construction Heuristics
  - Sweep
  - Nearest Neighbor, Nearest Insertion, Savings
  - Cluster Methods
- Improvement Heuristics
- Time Windows

Multiple Routes

- Capacitated VRP: vehicles have capacities.
  - Weight, Cubic feet, Floor space, Value.

- Deadlines force short routes.
  - Pickup at end of day.
  - Deliver in early morning.

- Time windows
  - Pickup.
  - Delivery.
  - Hard or Soft.

Multiple Route Solution Strategies

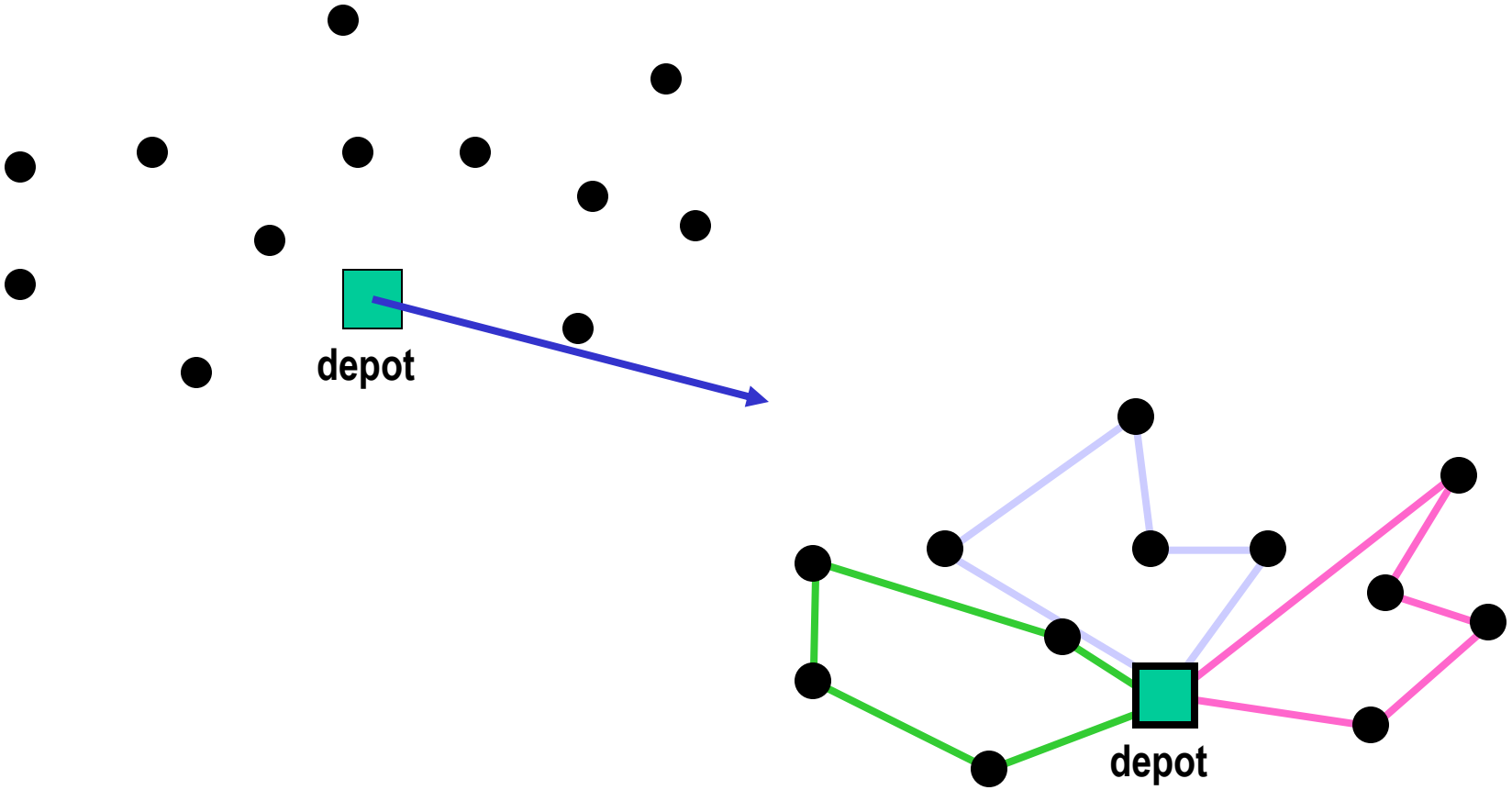
- Find feasible routes.
  - Cluster first, route second.
    - Cluster orders to satisfy capacities.
    - Create one route per cluster. (TSP for each cluster)
  - Route first, cluster second.
    - Create one route (TSP).
    - Break route into pieces satisfying capacities.
  - Build multiple routes simultaneously.
- Improve routes iteratively.

Sweep Algorithm

- Draw a ray starting from the depot.
- Sweep clockwise (or counter-clockwise) and add customers to the route as encountered.
- Start a new route when vehicle is full.
- Re-optimize each route (solve a TSP for customers in each route).

Sweep Algorithm

Suppose each vehicle capacity = 4 customers

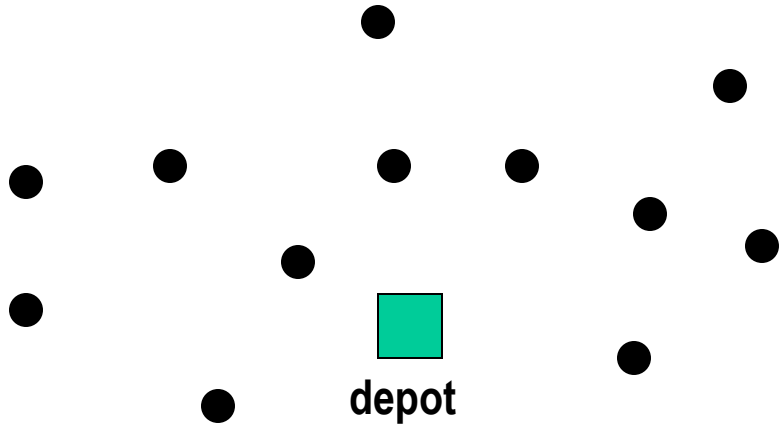


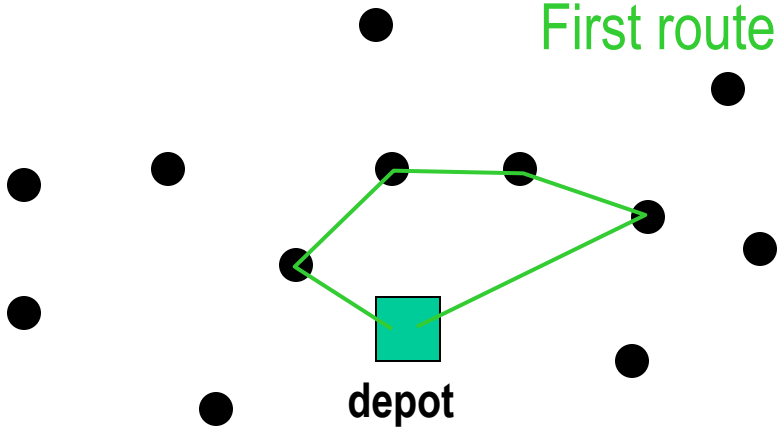
Nearest Neighbor, Nearest Insertion & Savings Algorithms

- Similar to for TSP, but keep track of demand on route.
- Start new route when vehicle is full.
- Re-optimize each route (solve a TSP for customers in each route).

Nearest Neighbor Algorithm

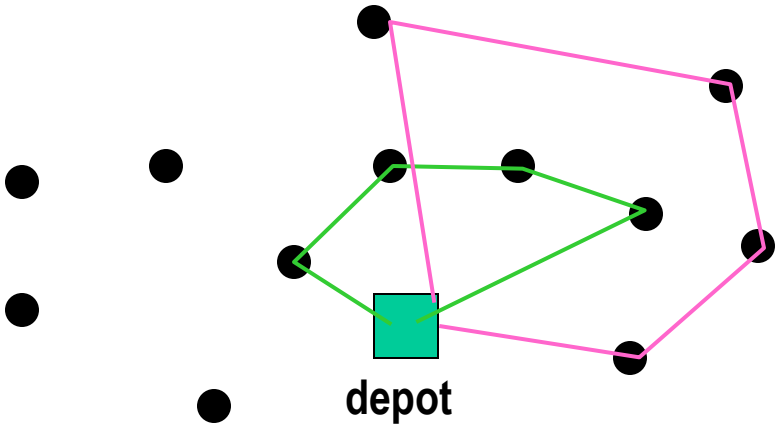
Suppose each vehicle capacity = 4 customers



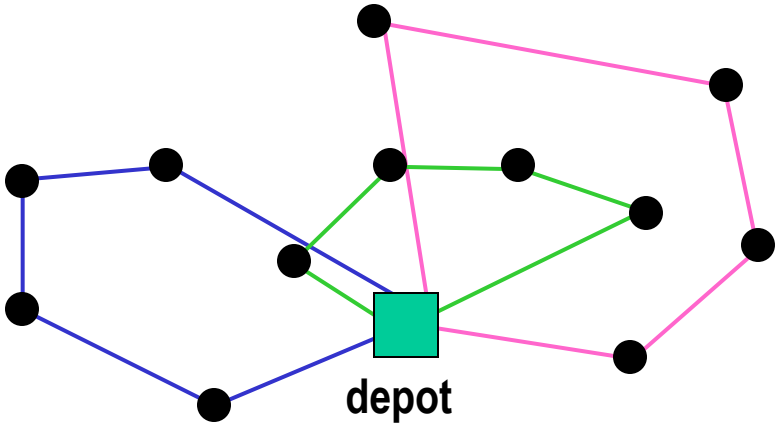


Nearest Neighbor Algorithm

Suppose each vehicle capacity = 4 customers



Second route



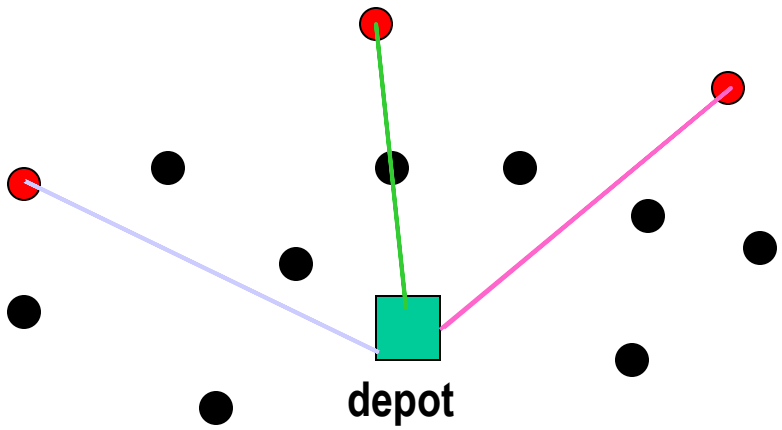
Third route

Cluster Algorithms

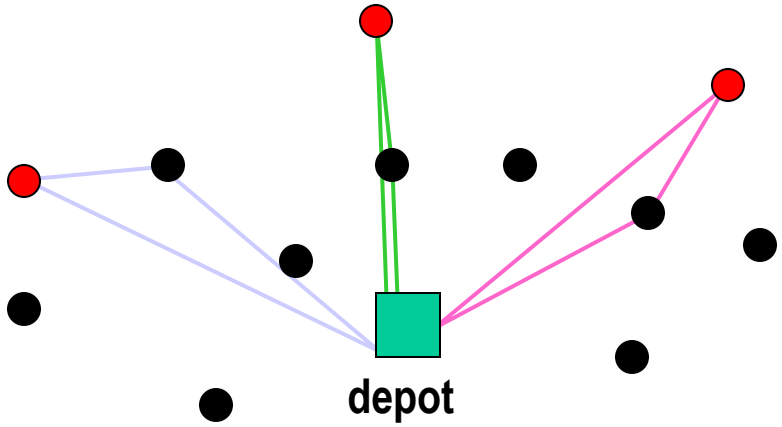
- Select certain customers as seed points for routes.
  - Farthest from depot.
  - Highest priority.
  - Equally spaced.
- Grow routes starting at seeds. Add customers:
  - Based on nearest neighbor or nearest insertion
  - Based on savings.
  - Based on minimum angle.
- Re-optimize each route (solve a TSP for customers in each route).

Cluster with Minimum Angle

Suppose each vehicle capacity = 4 customers



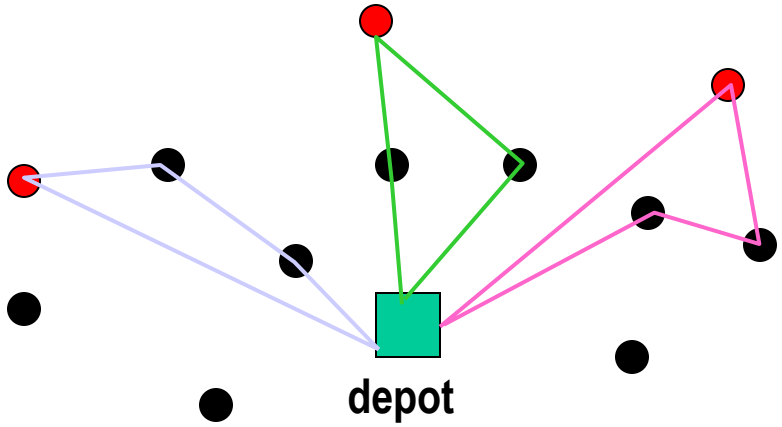
Select 3 seeds



Add customers with minimum angle

Cluster with Minimum Angle

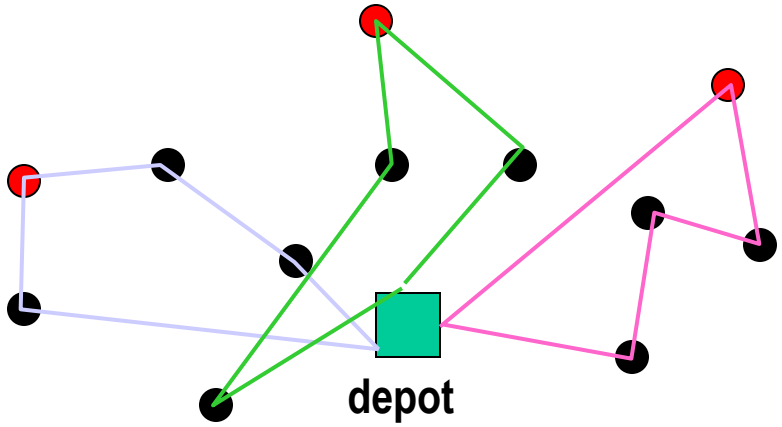
Suppose each vehicle capacity = 4 customers



Add customers with minimum angle

Cluster with Minimum Angle

Suppose each vehicle capacity = 4 customers



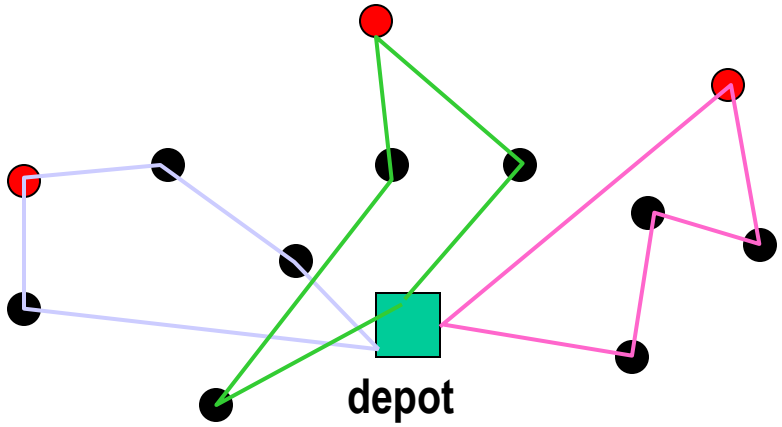
Add customers with minimum angle

VRP Improvement Heuristics

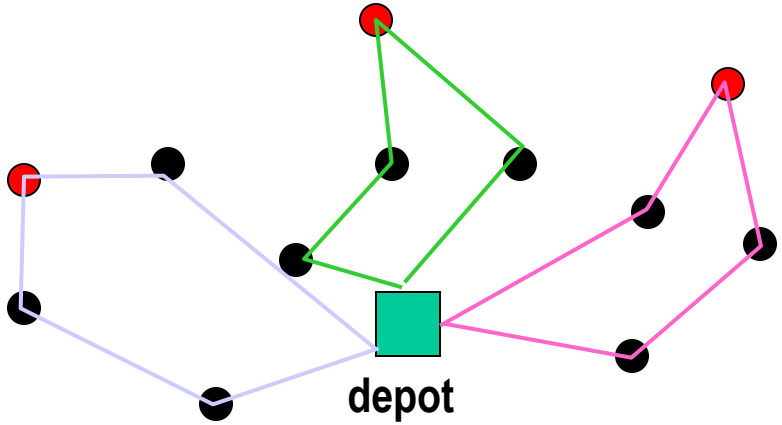
- Start with a feasible route.
- Exchange heuristics within a route.
  - Switch position of one customer in the route.
  - Switch 2 arcs in a route.
  - Switch 3 arcs in a route.
- Exchange heuristics between routes.
  - Move a customer from one route to another.
  - Switch two customers between routes.

Improvement Heuristics

Cluster with Minimum Angle



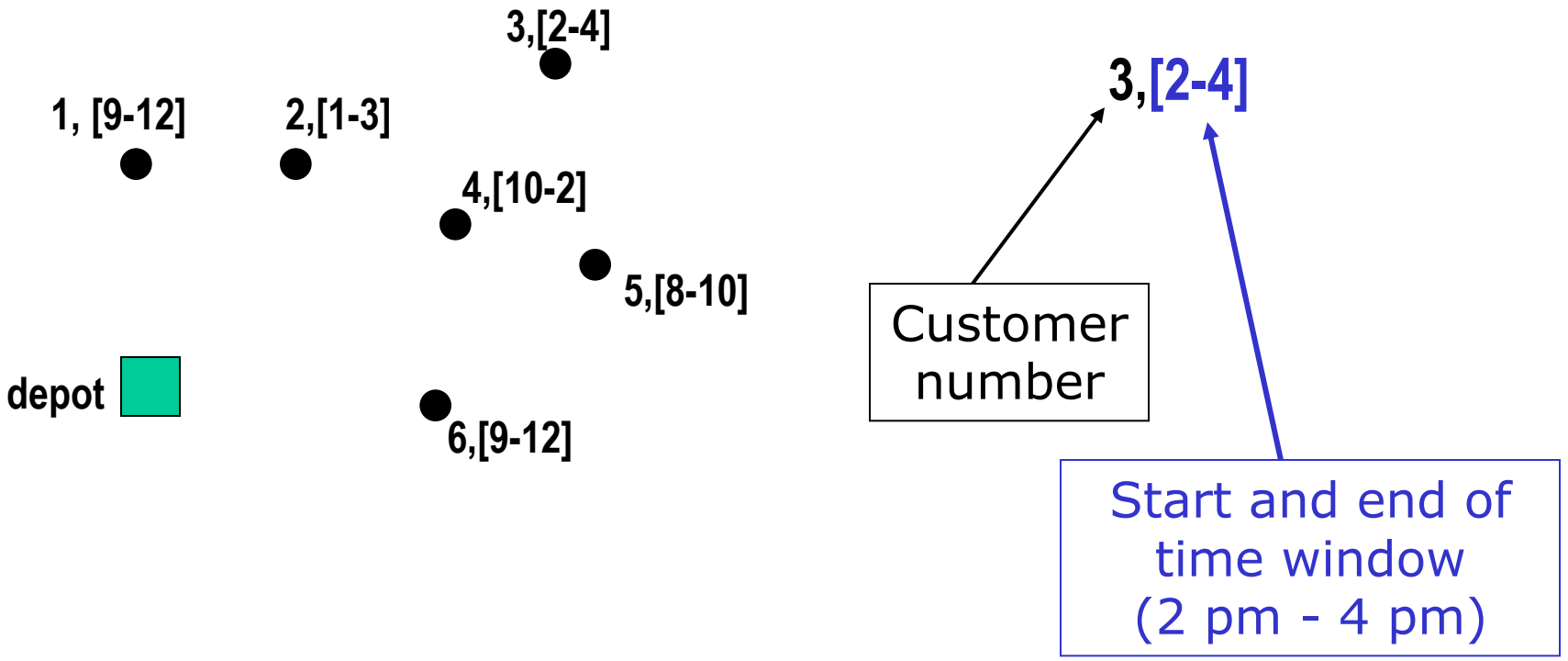
Starting routes



“Optimized” routes

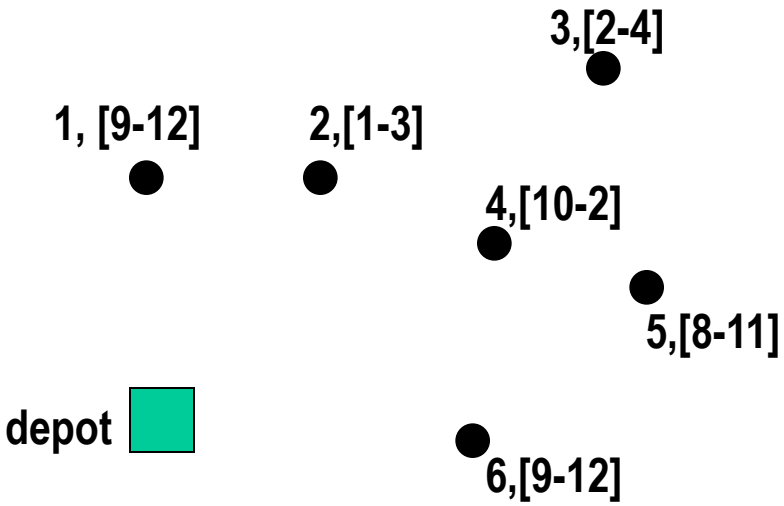
Time Windows

- Problems with time windows involve routing and scheduling.



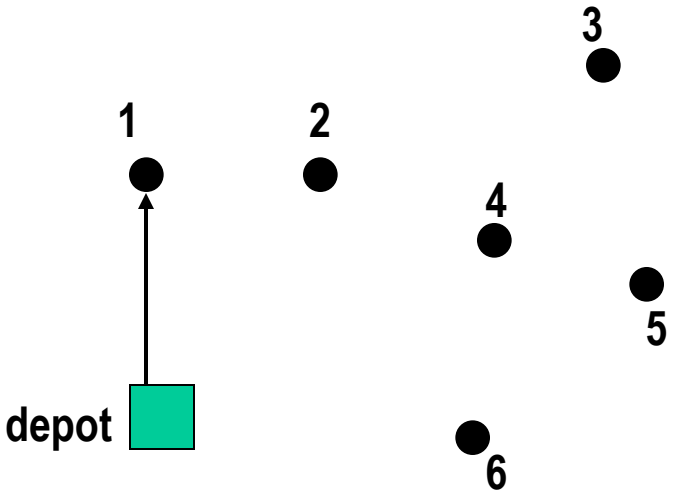
Routing Example with Time Windows

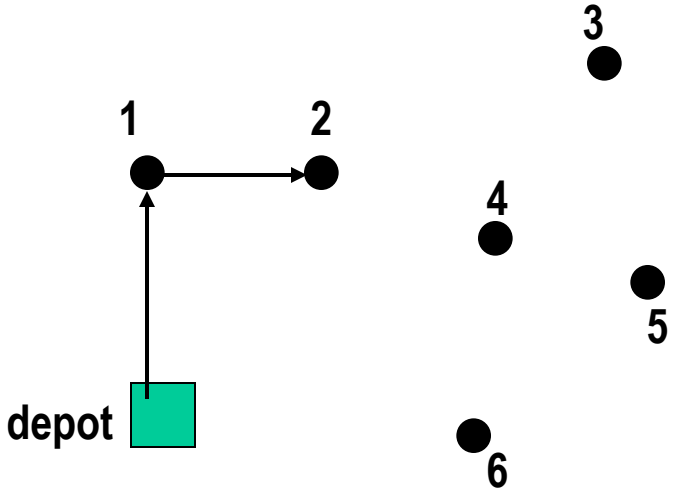
One hour travel time between any two customers.
Half hour delivery time at each customer.



Nearest Insertion - No Time Windows

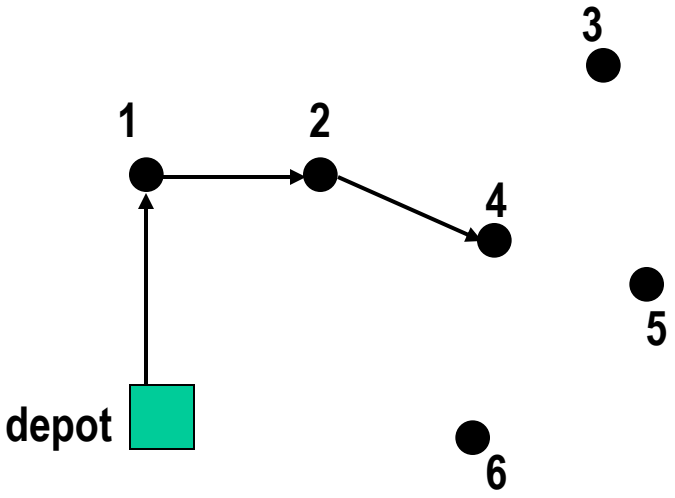
Insert nearest customer to route in best location.

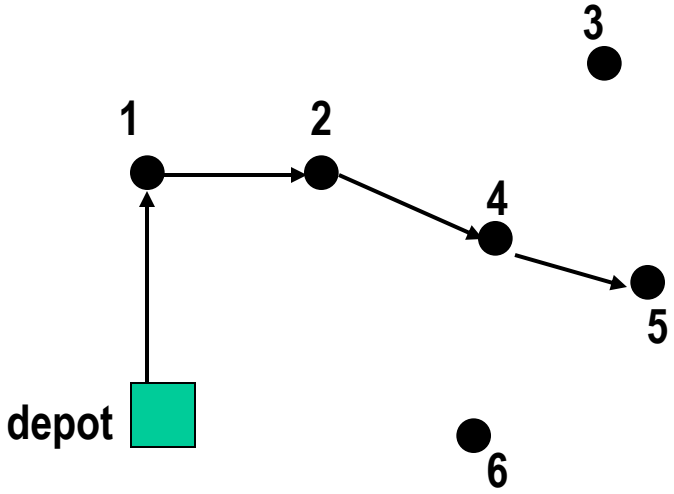




Nearest Insertion - No Time Windows

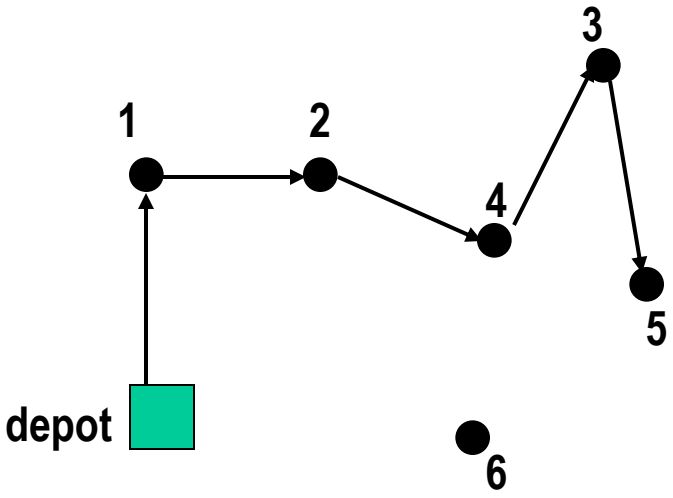
Insert nearest customer to route in best location.

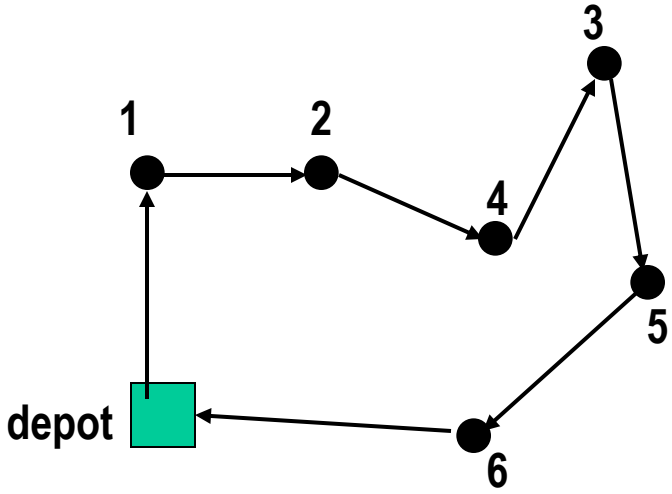




Nearest Insertion - No Time Windows

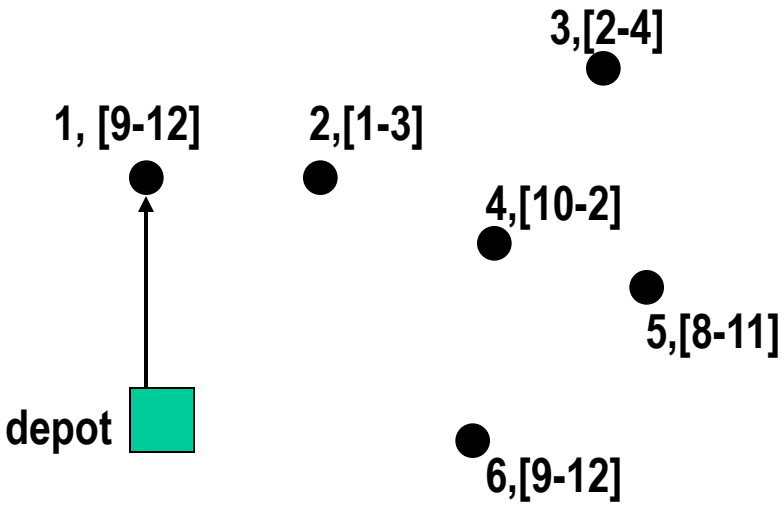
Insert nearest customer to route in best location.





Nearest Insertion with Time Windows

One hour travel time between any two customers.
Half hour delivery time at each customer.

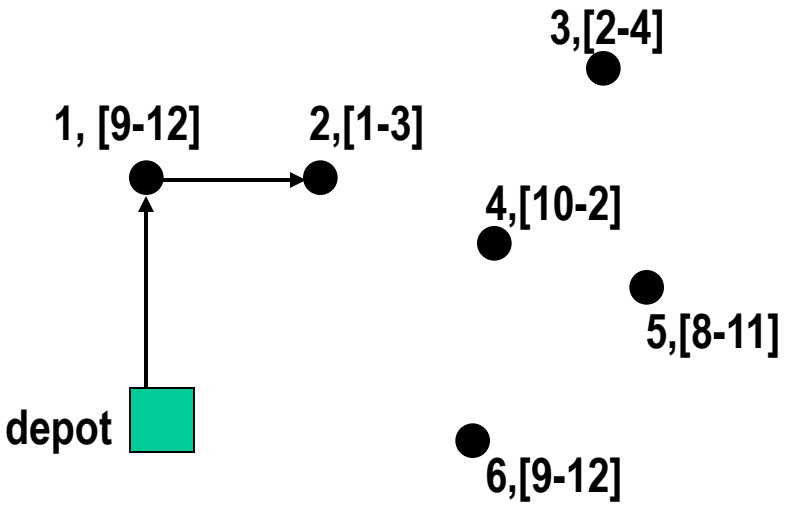


Leave depot: 8:00 - 11:00
Arrive at 1: 9:00 - 12:00
Leave 1: 9:30 - 12:30

depot-1

Nearest Insertion with Time Windows

One hour travel time between any two customers.
Half hour delivery time at each customer.

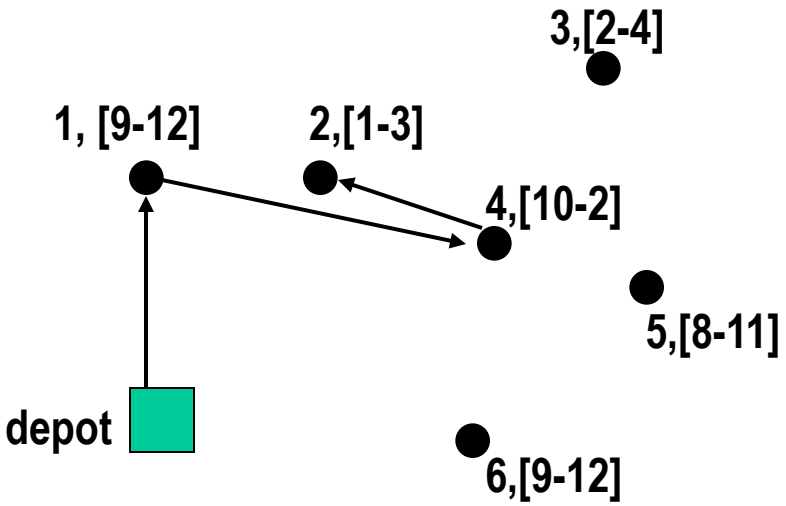


Leave depot:	10:30
Arrive at 1:	11:30
Leave 1:	12:00
Arrive at 2:	1:00
Leave 2:	1:30

depot-1-2

Nearest Insertion with Time Windows

One hour travel time between any two customers.
Half hour delivery time at each customer.

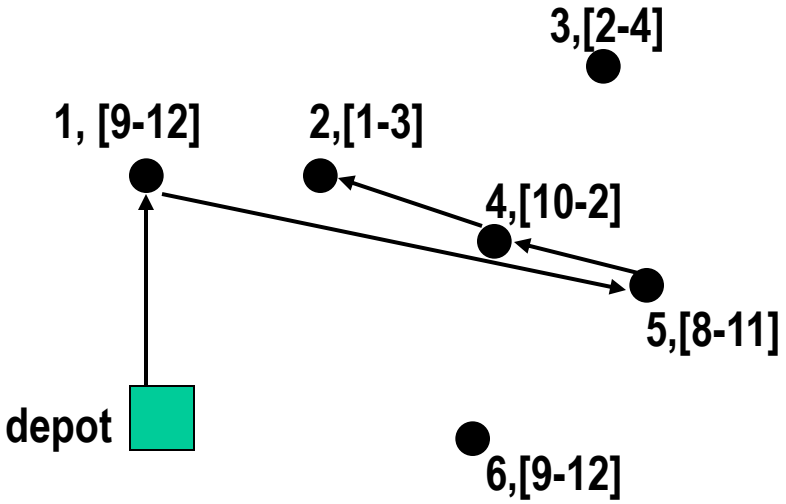


Leave depot:	9:00
Arrive at 1:	10:00
Leave 1:	10:30
Arrive at 4:	11:30
Leave 4:	12:00
Arrive at 2:	1:00
Leave 2:	1:30

depot-1-4-2

Nearest Insertion with Time Windows - one option

One hour travel time between any two customers.
Half hour delivery time at each customer.

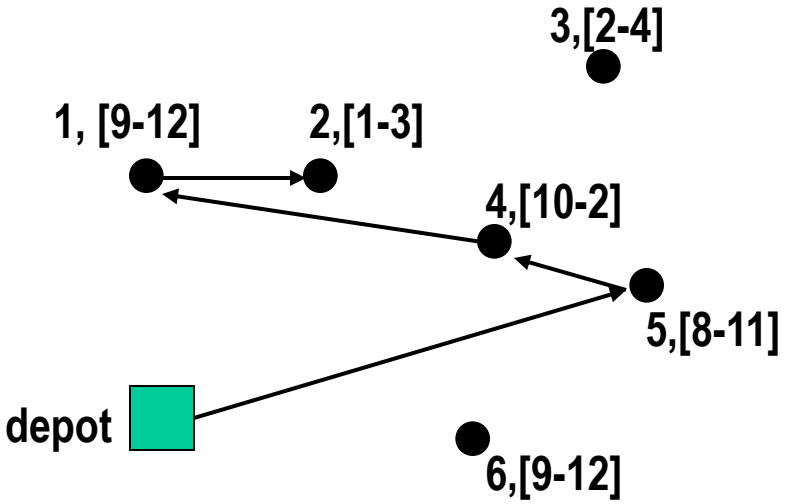


Leave depot:	8:00
Arrive at 1:	9:00
Leave 1:	9:30
Arrive at 5:	10:30
Leave 5:	11:00
Arrive at 4:	12:00
Leave 4:	12:30
Arrive at 2:	1:30
Leave 2:	2:00

depot-1-5-4-2

Nearest Insertion with Time Windows - a better option

One hour travel time between any two customers.
Half hour delivery time at each customer.

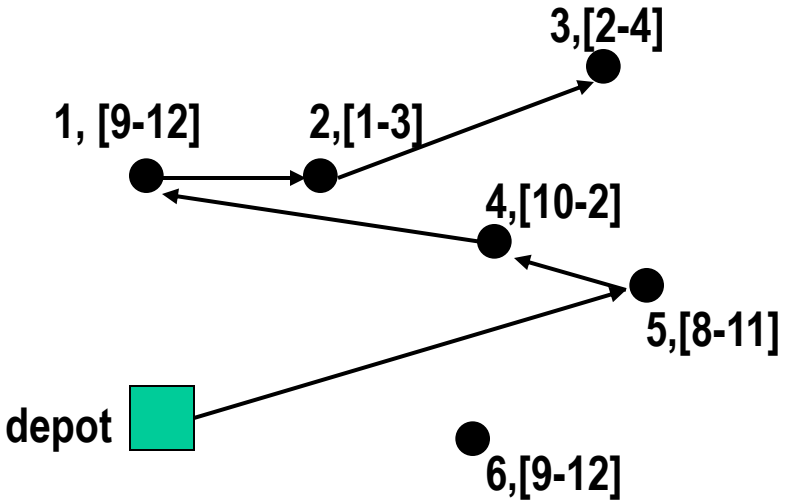


Leave depot:	7:30
Arrive at 5:	8:30
Leave 5:	9:00
Arrive at 4:	10:00
Leave 4:	10:30
Arrive at 1:	11:30
Leave 1:	12:00
Arrive at 2:	1:00
Leave 2:	1:30

depot-5-4-1-2

Nearest Insertion with Time Windows

One hour travel time between any two customers.
Half hour delivery time at each customer.

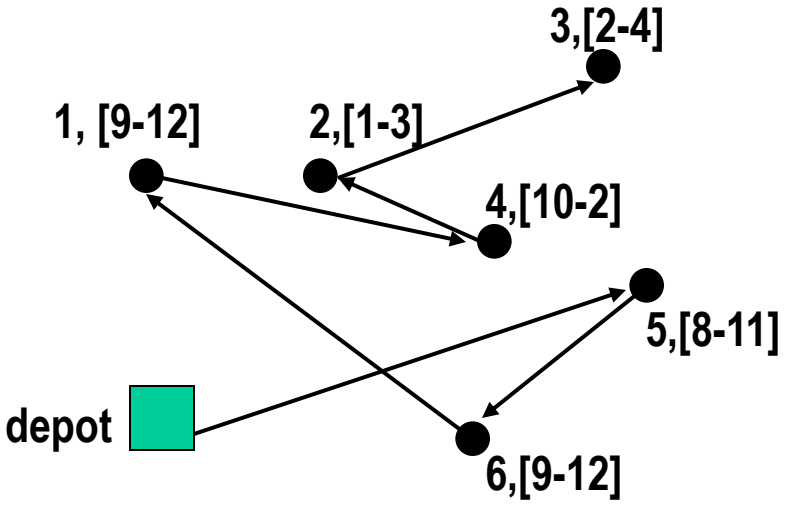


Leave depot:	7:30
Arrive at 5:	8:30
Leave 5:	9:00
Arrive at 4:	10:00
Leave 4:	10:30
Arrive at 1:	11:30
Leave 1:	12:00
Arrive at 2:	1:00
Leave 2:	1:30
Arrive at 3:	2:30
Leave 3:	3:30

depot-5-4-1-2-3

Nearest Insertion with Time Windows

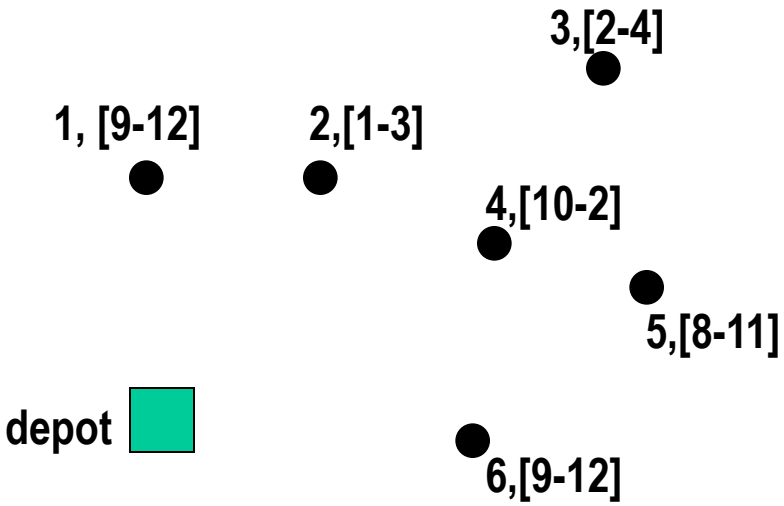
One hour travel time between any two customers.
Half hour delivery time at each customer.



Leave depot:	7:00
Arrive at 5:	8:00
Leave 5:	8:30
Arrive at 6:	9:30
Leave 6:	10:00
Arrive at 1:	11:00
Leave 1:	11:30
Arrive at 4:	12:30
Leave 4:	1:00
Arrive at 2:	2:00
Leave 2:	2:30
Arrive at 3:	3:30

depot-5-6-1-4-2-3

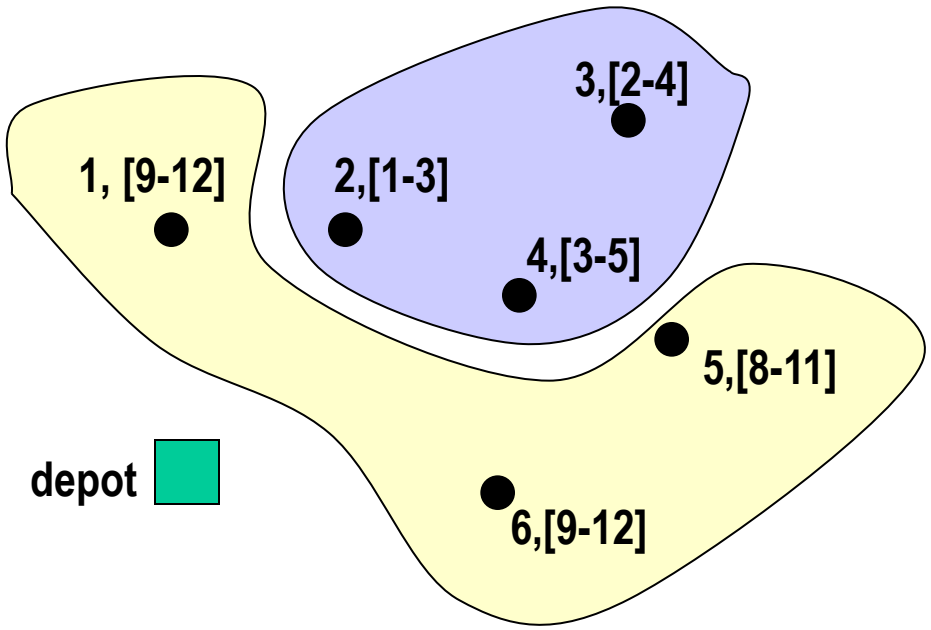
Routing with Time Windows



Route must be built with both time windows & geography in mind!

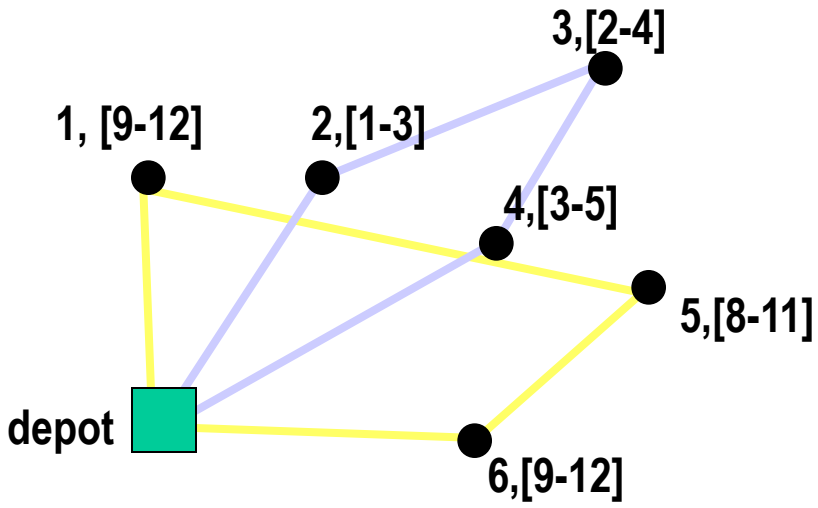
Clustering and Time Windows

- Cluster customers based on location and time window.
- Design routes for each cluster.



Clustering and Time Windows

- Cluster customers based on location and time window.
- Design routes for each cluster.



ArcLogistics Route Solution Technique

- Construct travel time (distance) matrix.
  - Shortest paths on actual (GIS) road network.
- Build initial routes by inserting orders on routes.
  - Can not exceed vehicle capacities, precedence rules, skill set constraints (specialties).
  - Can violate time windows and route duration.
- Improve routes.
  - Within route improvements.
  - Between route improvements.

ArcLogistics Route Insertions

- Objective: Minimize weighted sum of:
  - travel times *(t),*
  - time window violations *(v),* and
  - free (waiting) time *(w).*

$$\sum_r \alpha_1 t_r + \alpha_2 v_r + \alpha_3 w_r$$

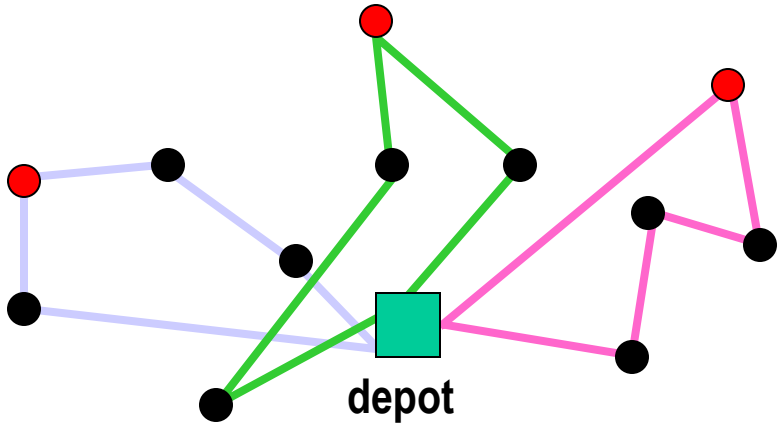
- User sets weights α_1, α_2, α_3 via slider bar in "Rate the importance of meeting time windows".
  - Lowest value = 0 is for lowest cost
  - Highest value = 10 is for most importance on meeting time windows.

ArcLogistics Route Improvements

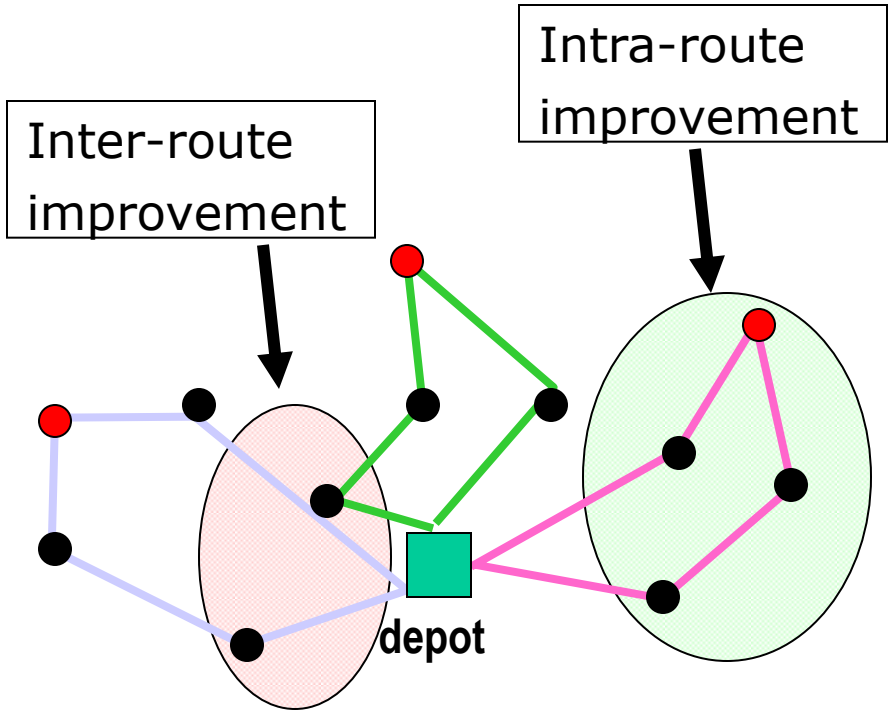
- Within route (intra-route) improvements.
  - Maintain stops on each route.
  - Consider each route separately.
  - Find best position for each stop on each route.

- Between route (inter-route) improvements.
  - Transfer or switch: Move one stop from one route to another.
  - Exchange: Exchange two stops between two routes.

Improvement Heuristics



Starting routes



"Improved" routes

Modeling Details

- Road network.
  - How detailed is the road network?
  - Affects distance and cost calculations.
- Geocoding.
  - Where are addresses located?
- Linking locations to road network.
  - How do vehicles get from locations to road nework?

Road Network Detail

- Models should measure travel distance, time and cost on actual road network.
- More detailed road networks mean:
  - More accurate cost, distance and travel time.
  - Slower solutions.
  - Longer time to set up data.
  - Larger data files.
- What is purpose?
  - Operational: To provide detailed driving directions?
  - Strategic: To estimate costs, times, and routes?

Road Network Detail

- How are speeds specified?
  - Different types of roads have different speeds.
  - Hard to include time varying speeds.
- Showing every road may be unrealistic.
  - Large trucks may not use small local roads.
- Road network changes over time.
  - Example: new residential areas.
  - Are customers in older, established areas or newer suburbs?

Geocoding

- Finding geographic locations from addresses.
- Depending on scale of problem, may geocode at various levels.
- Zip codes.
  - Geocode to a point at center of zip code.
  - Zip codes are defined with boundaries and centroids.
- Cities.
  - Geocode to a point representing city location.

Geocoding - Street Addresses

- Street addresses
  - Each road segment has associated address ranges on each side.
  - Address is interpolated between start and end of address range.
  - Assumes even spacing of addresses.

- Example:
  - Main Street, left side: 300-498, right side 301-499
  - 420 E. Main Street is about 60% of the way down the road segment on the left side.

Address Matching

- Can be very complicated.
- Only part of address may match.
  - Street name, city and zip code match; address range doesn't.
  - Street name and city match; zip code doesn't.
- Part of address may be incorrect.
  - Missing prefix (East).
  - Incorrect suffix (Street vs. Boulevard).
  - Name may be abbreviated in database (Ave for Avenue or Avenida).
  - Zip code may be wrong.

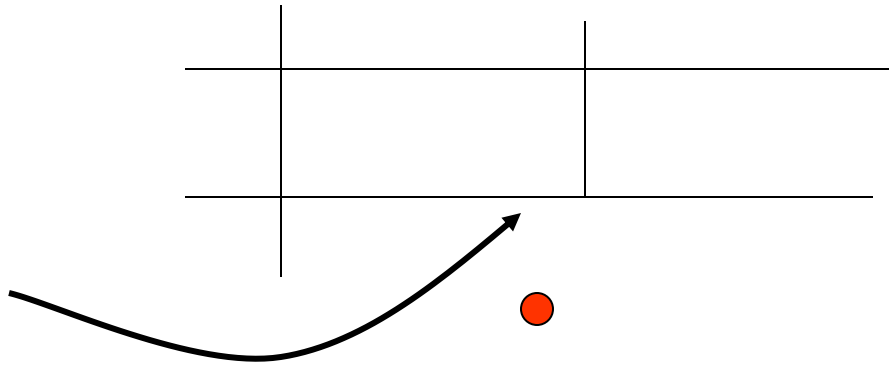
Address Matching

- Streets may have several names.
  - Highway 67 is also Lindbergh Blvd.
- Streets change names, especially at city boundaries.
  - Lindbergh, Highway 67, Kirkwood Road.
  - Olive Boulevard, Olive Street, Olive Street Road, Old Olive Street Road.
- New streets are added and old streets are renamed.

Linking Locations to Road Network

- User may specify locations not on streets.
  - User Pick in ArcLogistics Route.
- How should this be linked to street network?

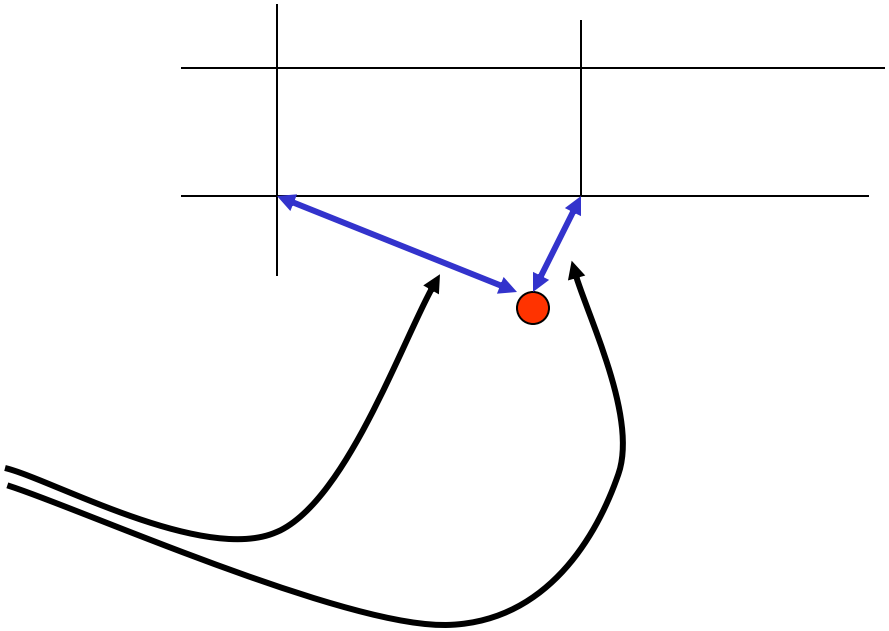
- ArcLogistics Route forces locations not on streets to be on nearest street.



ArcLogistics Route assumes locations is here

Linking Locations to Road Network

- Other packages assume straight line travel from nearest street intersection.



Some packages compares these distances.