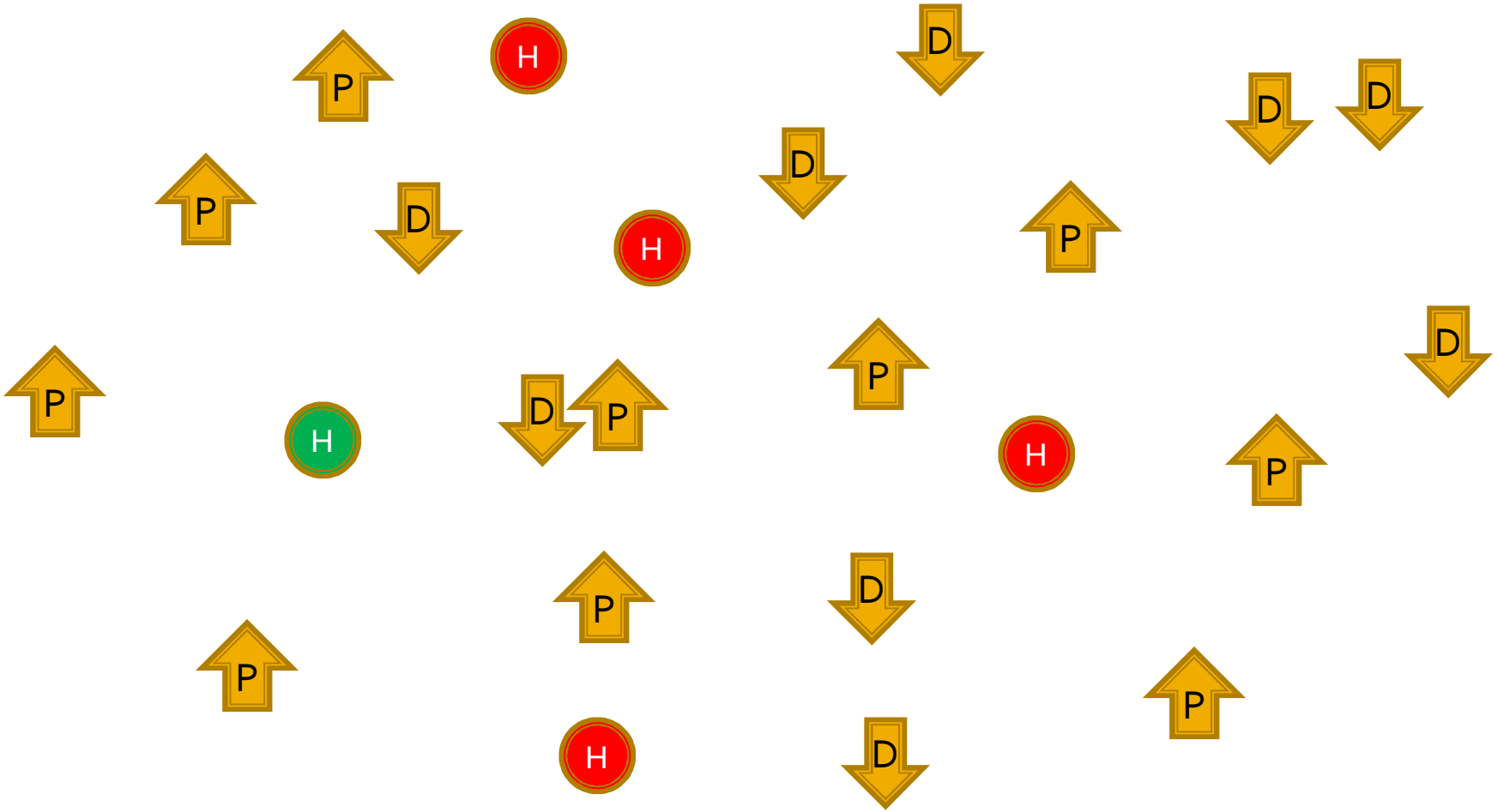
Douglas Popken, PhD
Systems View

Got Trash? Extending Vehicle Routing to Optimal Container Management

Problem Motivation

A client in the business of providing temporary bulk refuse containers needed an automated software system to help route their pickup and delivery trucks.



Problem Characteristics

- Customers receive empty containers that are picked up at a later date.
- A customer may also request a "switch": new container is delivered, while a full container is picked up.
- Five container sizes (cy).
- All transports start and end at one hub (in green)
- Transports carry one container at a time.
- Orders may require certain transports or exclude certain transports (accessibility limitations).
- Some orders have time of day restrictions.
- Three hubs store emptys and receive full containers
- Two additional hubs (landfill) can receive full containers, but must have no containers left overnight.

Route Generation Process

- Daily route generation with several dozen orders and up to 7 transports.
- A route defines all of the pickups and deliveries a transport will make during the day
- Previous routing approach was fully manual, using grid-sheets to track container inventory. Owner likened it to solving several simultaneous Sudoku puzzles. Took several hours by experienced planners, and generally assumed hub-and-spoke setup to simplify planning.

Why Not Use Route Factory Spreadsheet?

General purpose route optimization software that handles any mixture of pickup and delivery [see spreadsheet]. Written in C# using VSTO/.NET.

Operates within Microsoft Excel (2007,2010). Uses Microsoft MapPoint (2010,2011) for geocoding , road travel time/distance, and route display.

Orders flow over a set of defined ***Locations***. Each Order has a specified starting Location and ending Location. It may have a time window associated with pickup and/or delivery.

Transports start and end their routes at any defined Location, and stop at any Locations in between in any sequence. They have an earliest start and latest finish date/time.

Transports may have up to 3 capacity constraints; similarly, Orders may have up to 3 capacity types. In addition, Orders can have up to two attributes, for which each Transport *Can Carry*, *Must Carry*, or *Cannot Carry* (up to 3 attributes per Transport).

What's Missing?

No assurance that an empty container will be at a hub when you need it!

Not just an issue of starting inventory, containers become available for use after they are brought back to the hub and dumped.

→Two major requirements

1. Modify Route Factory Spreadsheet to dynamically monitor inventory levels at each location as routes progress so that a pickup never leaves unless and until an empty container is available
2. A preprocessing capability to assign orders to hubs so that containers are available for order fulfillment.

Route Factory Spreadsheet Algorithm Modifications

•Each Location is given an Inventory Manager (IM) object. IM's have a time ordered list of inventory events (increase/decrease) for each Inventory Type (e.g. container size) as well as starting inventories.

•The inventory events correspond to route stops with associated pickup/delivery of containers.

•Every time a route changes, IM's on the route must be notified and modified.

•Before a pickup is deemed feasible, the inventory of the given type, at that specific time, must be positive. You also must ensure you are not stealing another routes inventory (all future levels must be nonnegative). This must be checked at each stop.

•This created linkages between routes and huge problems since previously, all trial routes could be handled by cloning a route, modifying the clone, and evaluating it's feasibility independently. If you don't like the trial route, toss.

•Eventually used a somewhat convoluted scheme involving cloning of IMs.

Route Factory Interface Addition

See spreadsheet

Preprocesor: Model Objective

- Objective: Assign customers to hubs so that total hub to customer distance is minimized and container inventories end within minimum and maximum values.
- This provides a network design for the routing algorithms, that is, each order has a defined start and end point.
- Note that this is an approximation since actual routes are unknown at this point. Transports may make several stops and can travel between hubs. (A further approximation we made is the use of great-circle distance rather than road miles.)

Model Formulation

- Components ["Sets"]:
  - Customers i
  - Hubs j
  - Container Sizes k
- Variables ["Decisions"]:
  - x_{ij} = 1 if customer i is assigned to hub j, 0 otherwise
- Input Parameters ["Parameters"]:
  - d_{ij} = distance from customer i to hub j
  - p_i = 1 if customer is a pickup, -1 if a dropoff
  - $start_{jk}$ = hub j inventory of container size k at beginning of day
  - $emin_{jk}$ = minimum ending inventory of container size k at hub j at end of day
  - $emax_{jk}$ = maximum ending inventory of container size k at hub j at end of day
  - m_{ik} = 1 if customer i uses container size k, 0 otherwise

Mathematical Formulation

Minimize total distance = $\sum_i \sum_j d_{ij} x_{ij}$

Subject to:

$$Start_{jk} + \sum_i p_i x_{ij} m_{ik} \geq e\min_{jk} \quad \text{for all j,k}$$

$$Start_{jk} + \sum_i p_i x_{ij} m_{ik} \leq e\max_{jk} \quad \text{for all j,k}$$

$$\sum_j x_{ij} = 1 \quad \text{for all i}$$

$x_{ij} = 0,1$

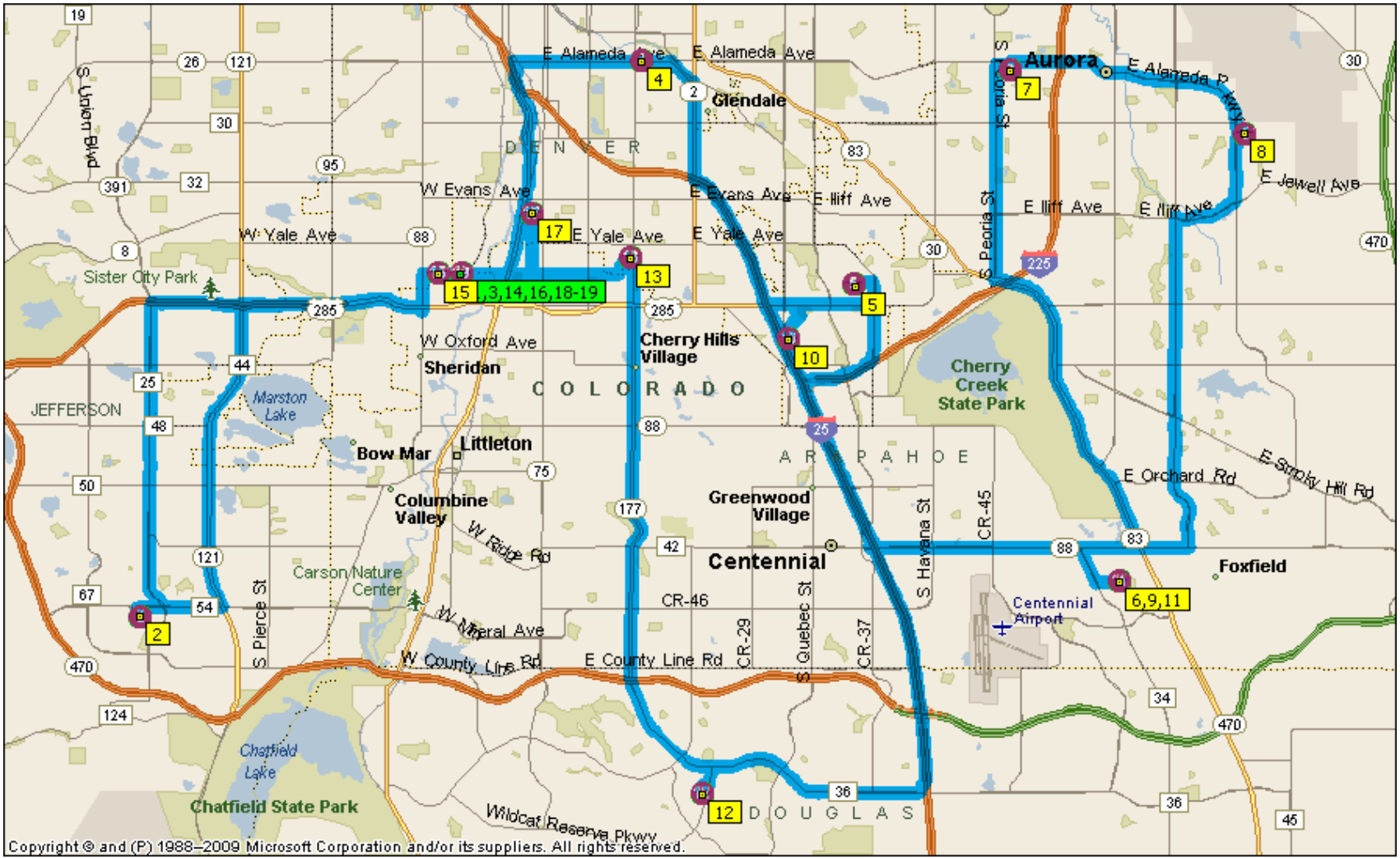
Solver Selection Considerations

- Need to integrate with Excel for ease of data import/export (cut and paste).
- Free or nearly so.
- Standard Excel solver is limited to 200 decision variables (with 5 hubs we would be limited to 40 customers).
- Numerous other Excel compatible solvers available, but free versions are usually time-limited, for academic use only, or would require significant intermediate programming for model/data transfer.
- Discovered Microsoft Solver Foundation Express Edition.
  - Free
  - Available as Excel Add-in
  - Easy to learn UI for model building
  - Size limitation more generous/flexible (50,000 non-zeros → 95 customers).

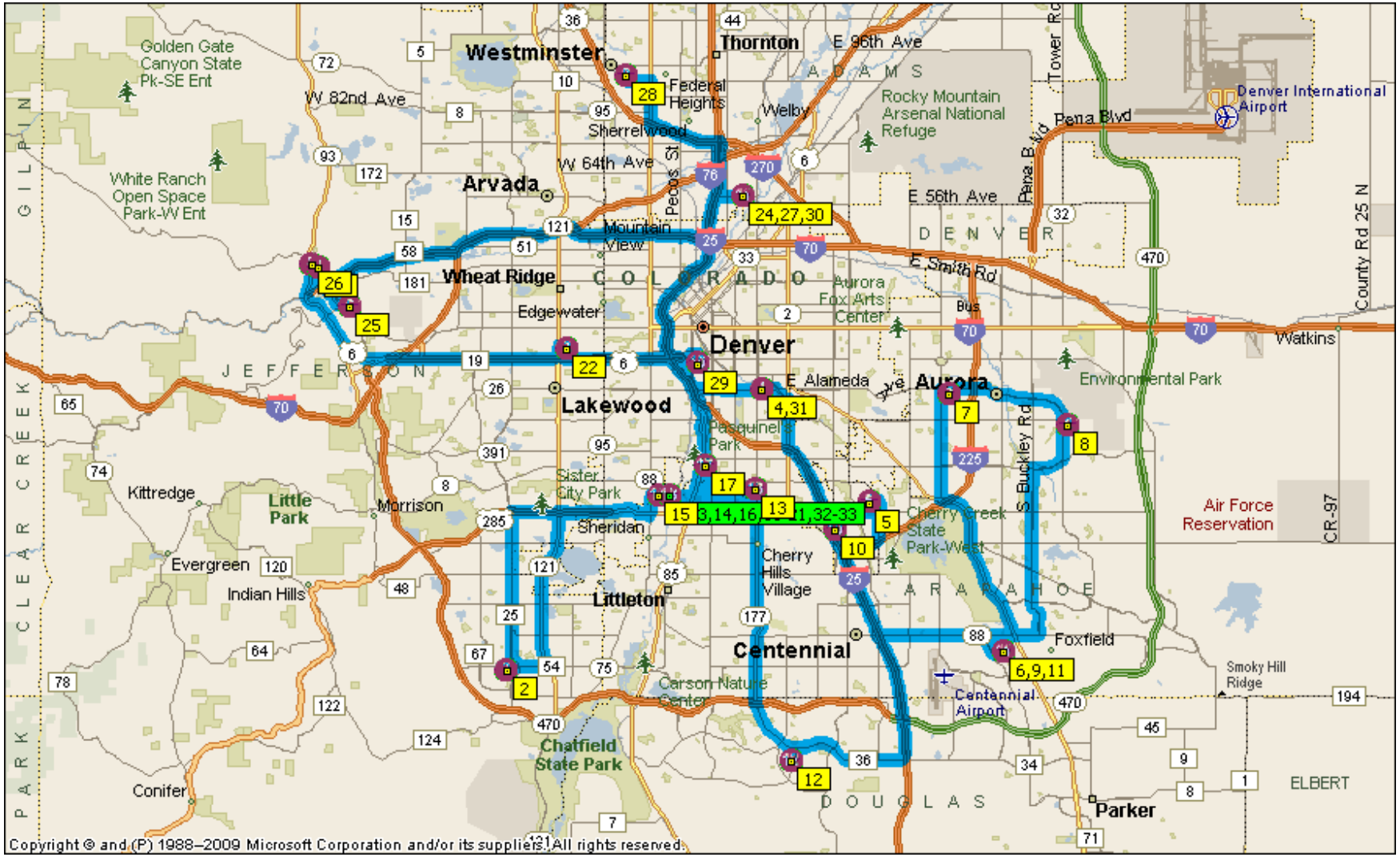
Assignment Model Implementation

See spreadsheet

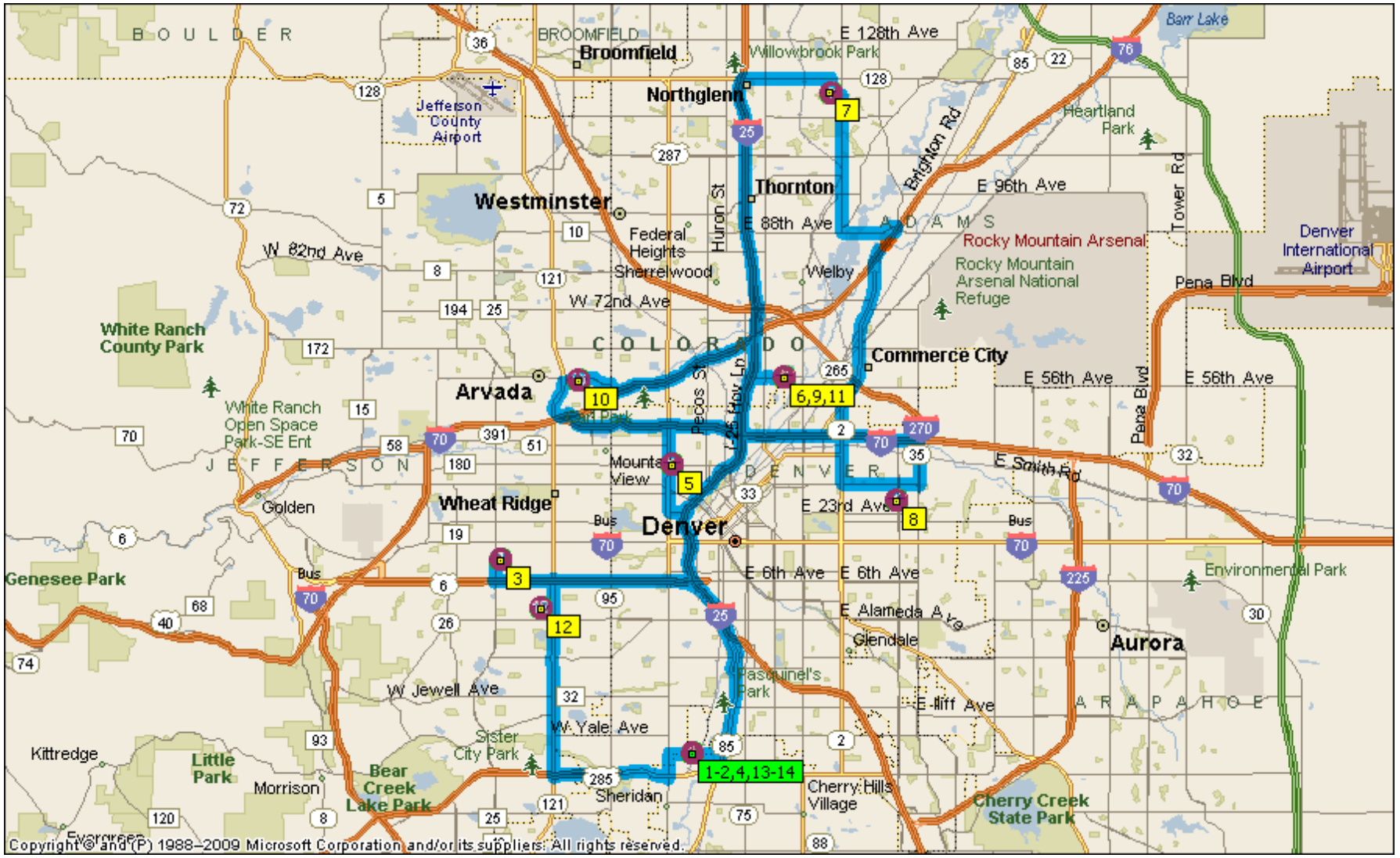
Sample Route Maps (1 of 3)



Sample Route Maps (2 of 3)



Sample Route Maps (3 of 3)



Lessons Learned

- VSTO (user friendly-yes; developer friendly-not so much)
- You will spend a lot of time on software issues unrelated to OR algorithms (e.g install issues).
- It always takes longer than you think.
- Each customer introduces new scenarios and dynamics you may not have considered.
- It's not over until the customer is happy (e.g. switch handling)
- It's very satisfying when the customer *is* happy and measurable savings are being achieved ("Doing good through good OR").