Approximation

- Simple/trivial ($r+\varepsilon$)-approximation (r=speed ratio = v_{drone}/v_{truck}):
  - Just have truck+drone travel to each target together
  - Use PTAS ($1+\varepsilon$)-approx for TSP of targets

Thus, goal is to get *below* factor r, ideally a factor O(1) indep of r .

- For approximation purposes, to get O(1)-approx in Euclidean problem, it suffices to solve/approximate the L_1 metric version (give up factor sqrt(2))

Approximation Algorithms

- A first provable approximation (polylog(n) factor) algorithm for general instances, indep of speed ratio: uses one-of-set-TSP
- Improvement: O(log n)-approximation

In the works: O(1)-approx (complex, m-guillotine)

Possible PTAS?

Natural *simple* candidate heuristic: cheapest insertion

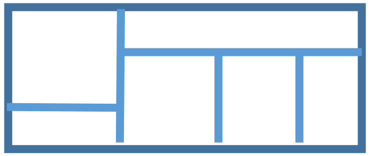
For a solution tour (truck, drone) on *i* targets, examine each of the remaining n-i targets and insert (cheapest local insertion) the one that lengthens the makespan the least.

Q: Is this a provable approximation? (for usual TSP, it is 2-approx)

O(log n)-approx (sketch)

- Dynamic Program: Subproblem is a rectangle R, around which the truck travels the full perimeter.

  Optimize: Find cheapest "BSP" truck network, with "spokes" to all target sites, weighting the length of the truck network by r, the speed ratio (and spokes with weight 1)



Proof of approx factor:

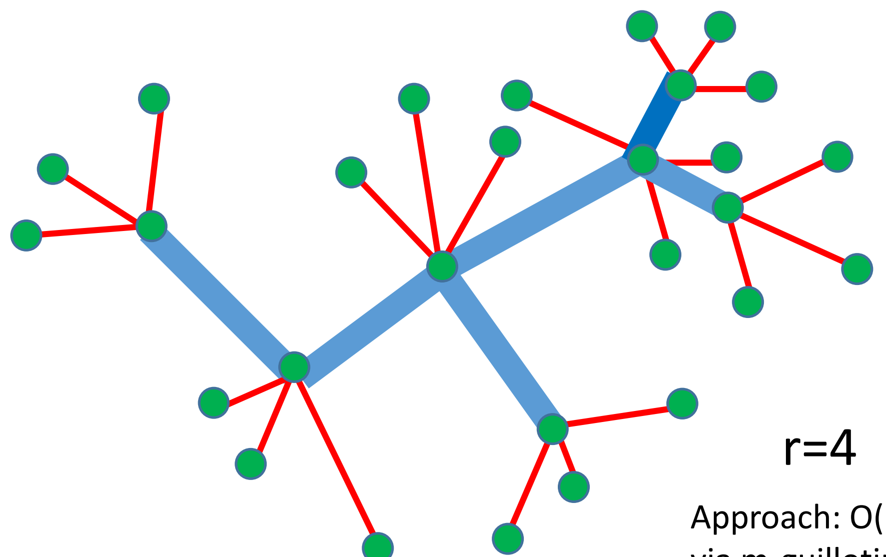
- Convert drone route to be a set of doubled "spokes" attached to truck route at pause points (truck waits) Pay factor 2
- Convert Opt routes to be rectilinear (L_1), on grid Pay factor sqrt(2)
- Augment truck route to be a BSP network; Pay factor log(n)

note that resulting solution is among those searched by DP

- Solution recovery: From DP solution, at O(1) factor can retrieve a valid solution to original problem

Towards an O(1) Approximation

- Lemma: An α-approximation for the Weighted-Backbone-and-Spoke Spanning Tree (WBSST) problem implies an O(α)-approximation for Horsefly.



Cover sites with r-stars (weight 1 edges), linking the centers of r-stars via a spanning tree (blue) of edges weighted r.

WBSST: Compute a min-weight such spanning structure

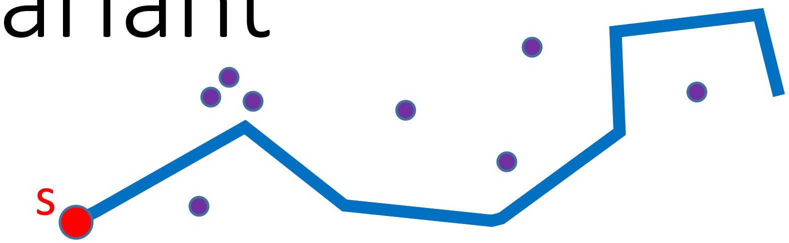
r=4

Approach: O(1)-approx for WBSST via m-guillotine methods

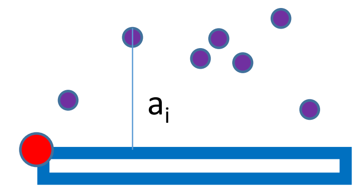
Other Variants

- Fixed truck route (“truck” might be a train)
- *Reverse Horsefly*: Each customer owns their own drone and dispatched it to rendezvous with the truck to retrieve package. Find optimal truck route (after which drone route is trivial – shortest segment to truck route)
- Truck travel on road network (vs L_2 metric)
- Obstacles also for drone flights (buildings, no-fly zones)
- Limited range drone flights
- Drone has package capacity C: can carry up to C packages and deliver them before return to truck
- Loading/unloading/chargin times for drones
- Multiple drones per truck
- Multiple trucks, possibnly with multiple depots
- Uncertain travel times (truck in traffic)
- Truck can deliver packages too (without drone)
- Customers are not points but are *regions* (TSPN)
- Online, dynamic variants

Fixed Truck Route Variant



Truck route is *given* and the goal is to schedule the speed of the truck (which may stop/start) and the drone routes to min makespan.



- Weakly NP-hard (from PARTITION), in general, for route=cycle
- Special case: Single segment path *st* (not cycle) for truck.

  Two subcases:

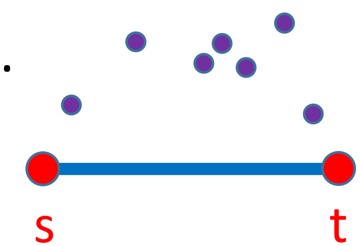
  (a) Completion when drone and truck both reach *t*

  - *(a) In L_1 metric, exactly solved: Fact (exchange argument): Targets visited in x-order*
  - *(b) In L_2 metric, complexity is open*

  (b) Completion when drone returns to truck after last delivery

  Issue: Even for L_1 metric, Opt may visit sites in order other than x-order

  Open: Complexity of problem, for either L_1 or L_2 metric



“Reverse Horsefly” Variant

Plan a route for the truck, but now there is a drone at each customer, which is sent to rendezvous with the truck to pick up the package. Goal is still to min makespan.

- In general, NP-hard (from TSP)
- Goal: Good approximations