Dynamic Pickup and Delivery with Transfers

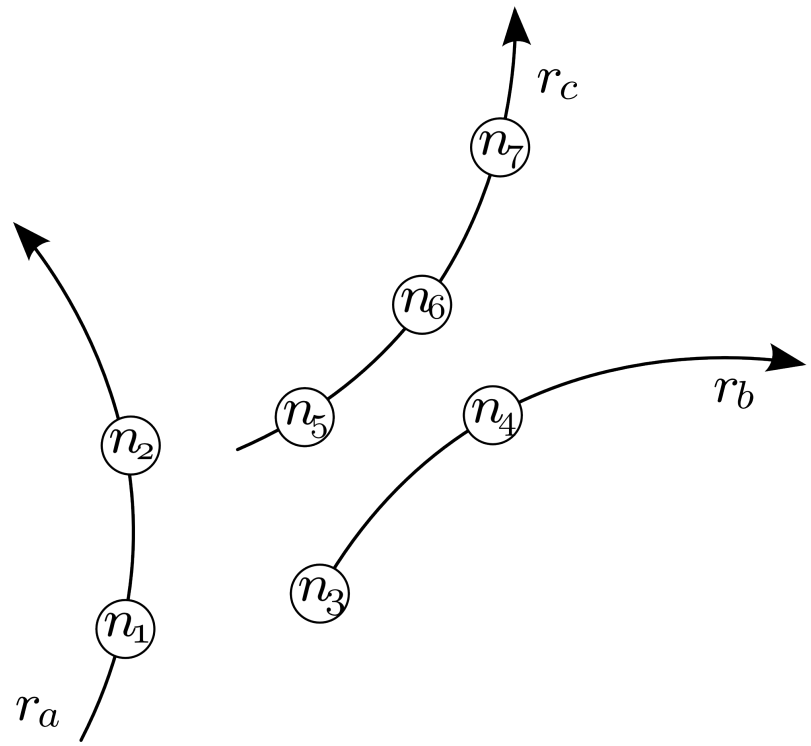
P. Bouros[1], D. Sacharidis[2], T. Dalamagas[2], T. Sellis[1,2]
[1]NTUA, [2]IMIS – RC "Athena"

Outline

- Introduction
- Related work
  - Pickup and delivery problems
  - Shortest path problems
- Solving dynamic Pickup and Delivery with Transfers
  - Actions
  - Dynamic plan graph
  - The SP algorithm
- Experimental evaluation
- Conclusions and Future work

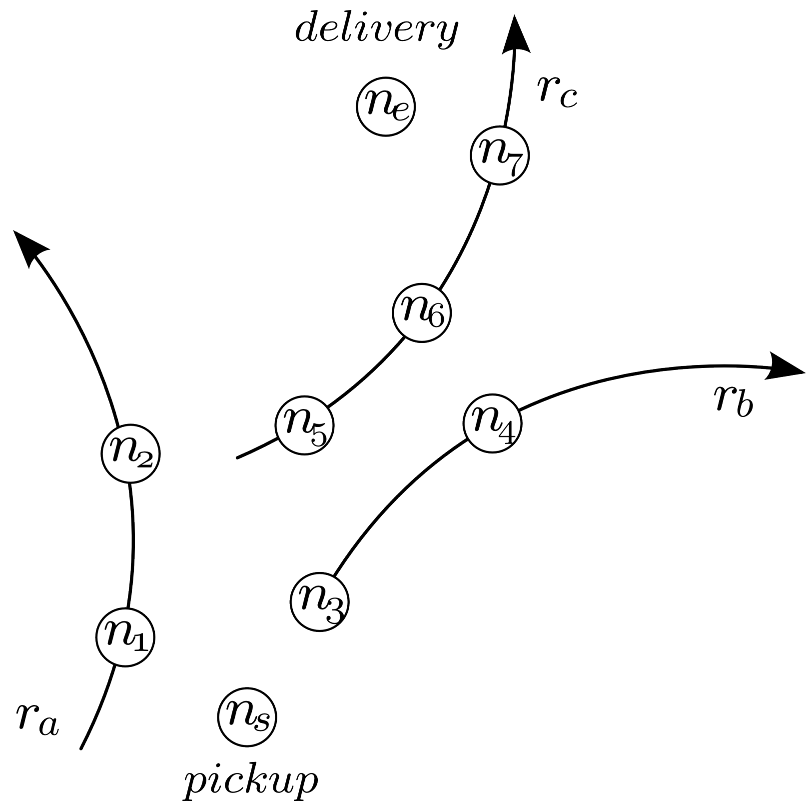
Motivation example

- A courier company offering pickup and delivery services
- Static plan
  - Set of requests
  - Transfers between vehicles
  - Collection of vehicles routes
- Pickup and Delivery with Transfers
  - Create static plan
- Ad-hoc requests
  - Pickup package from n_s, deliver it at n_e
- dynamic Pickup and Delivery with Transfers (dPDPT)
  - Modify static plan to satisfy new request



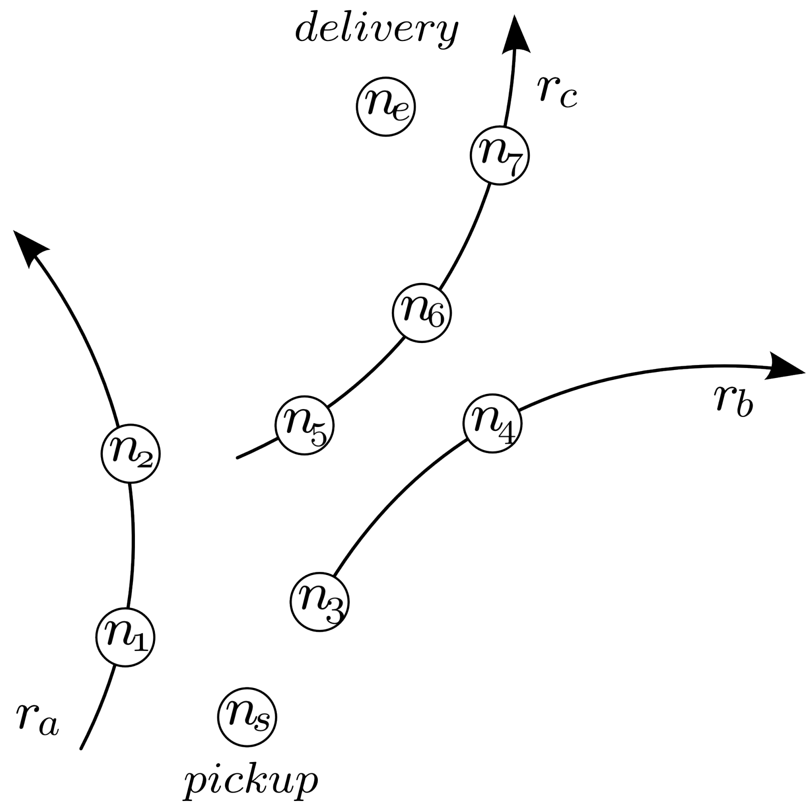
Motivation example

- A courier company offering pickup and delivery services
- Static plan
  - Set of requests
  - Transfers between vehicles
  - Collection of vehicles routes
- Pickup and Delivery with Transfers
  - Create static plan
- Ad-hoc requests
  - Pickup package from n_s, deliver it at n_e
- dynamic Pickup and Delivery with Transfers (dPDPT)
  - Modify static plan to satisfy new request



Motivation example

- A courier company offering pickup and delivery services
- Static plan
  - Set of requests
  - Transfers between vehicles
  - Collection of vehicles routes
- Pickup and Delivery with Transfers
  - Create static plan
- Ad-hoc requests
  - Pickup package from n_s, deliver it at n_e
- dynamic Pickup and Delivery with Transfers (dPDPT)
  - Modify static plan to satisfy new request



Contributions

- First work targeting dPDPT
  - Works for dynamic Pickup and Delivery can be adapted to work with transfers
- dPDPT as a graph problem
  - Works for dynamic Pickup and Delivery involve two-phase local search method
- Cost metrics
  - Company's viewpoint, extra traveling or waiting time
  - Customer's viewpoint, delivery time
- Solution
  - Dynamic two-criterion shortest path

Related work

- Pickup and delivery problems
  - Precedence and pairing constraints
  - Variations
    - Time windows
    - Capacity constraint
    - Transfers
  - Static
    - Generalization of TSP
    - Exact solutions
      - Column generation, branch-and-cut
    - Approximation
      - Local search
  - Dynamic
    - Two phases, insertion heuristic and local search

Related work

- Pickup and delivery problems
  - Precedence and pairing constraints
  - Variations
    - Time windows
    - Capacity constraint
    - Transfers
  - Static
    - Generalization of TSP
    - Exact solutions
      - Column generation, branch-and-cut
    - Approximation
      - Local search
  - Dynamic
    - Two phases, insertion heuristic and local search

Related work (cont'd)

- Shortest path problems
  - Classic
    - Dijkstra, Bellman-Ford
    - ALT: bidirectional A*, graph embedding
    - Materialization and labeling techniques
  - Multi-criteria SP
    - Reduction to single-criterion: user-defined preference function
    - Interaction with decision maker
    - Label-setting or correcting algorithms: a label for each path reaching a node
  - Time-dependent SP
    - Cost from n_i to n_j depends on departure time from n_i
    - Dijkstra: consider earliest possible arrival time
    - FIFO, non-overtaking property

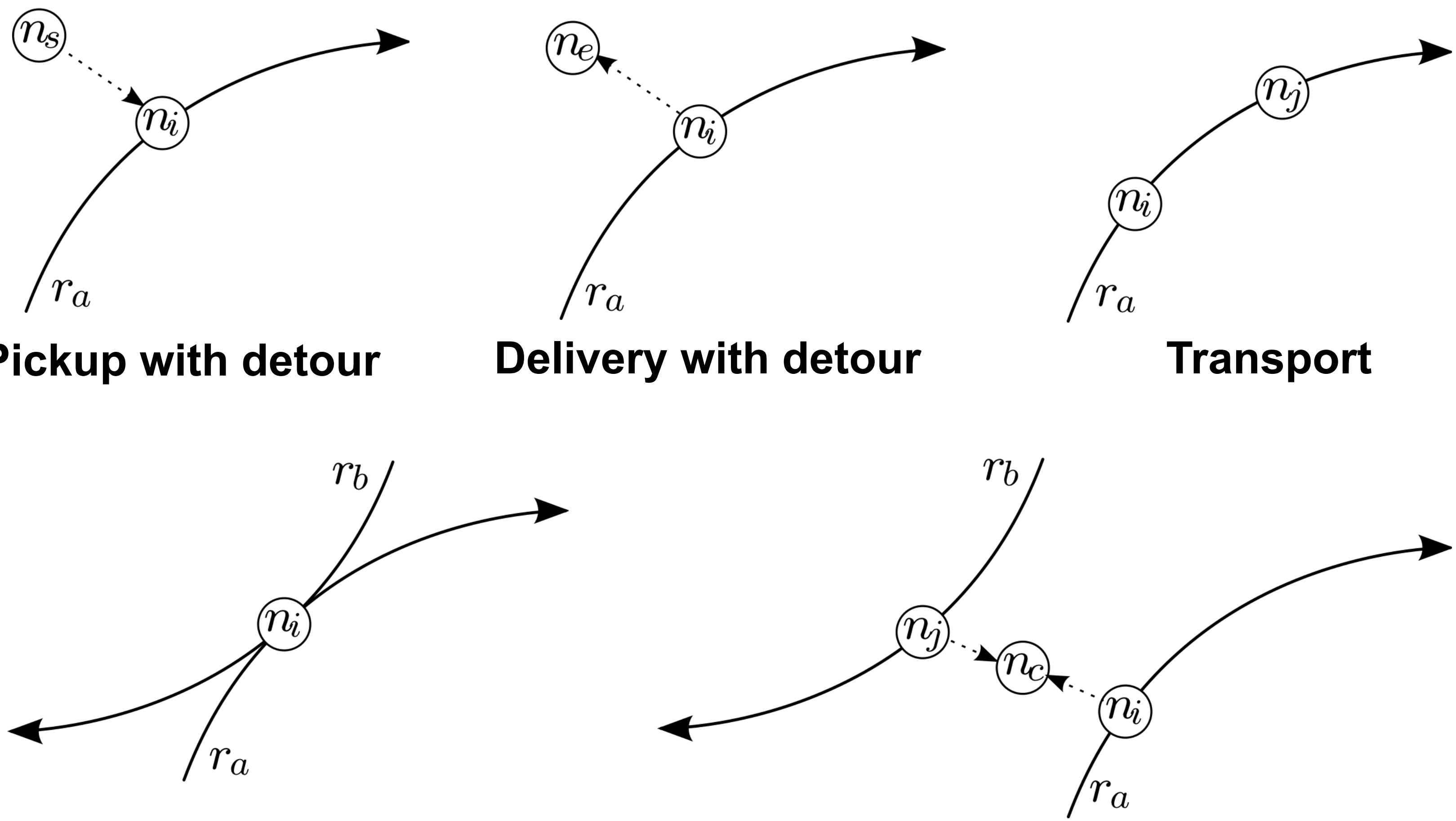
Related work (cont'd)

- Shortest path problems
  - Classic
    - Dijkstra, Bellman-Ford
    - ALT: bidirectional A*, graph embedding
    - Materialization and labeling techniques
  - Multi-criteria SP
    - Reduction to single-criterion: user-defined preference function
    - Interaction with decision maker
    - Label-setting or correcting algorithms: a label for each path reaching a node
  - Time-dependent SP
    - Cost from n_i to n_j depends on departure time from n_i
    - Dijkstra: consider earliest possible arrival time
    - FIFO, non-overtaking property

Solving dPDPT

- Modify static plan
  - 4 modifications, called actions, allowed with/without detours
    - Pickup, delivery
    - Transport
    - Transfer
- A sequence of actions, path p
  - Operational cost Op
  - Customer cost Cp
- Dynamic plan graph
  - All possible actions
- Solution to a dPDPT request
  - Path p with that primarily minimizes Op, secondarily Cp

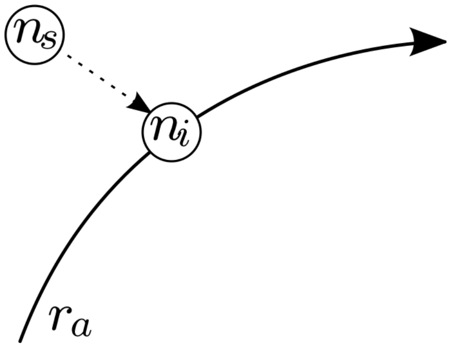
Actions



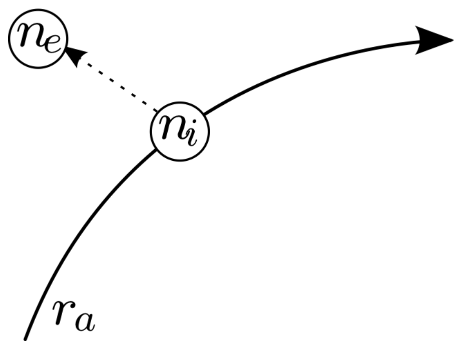
Pickup with detour **Delivery with detour** **Transport**

Transfer without detour **Transfer with detour**

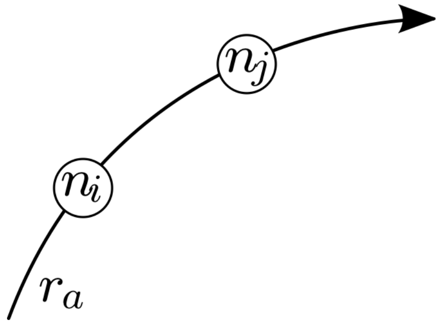
Actions



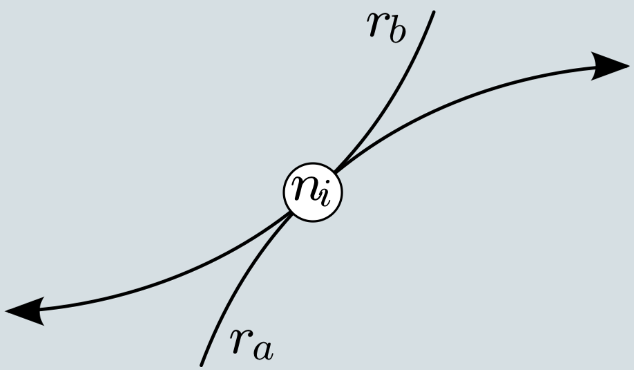
Pickup with detour



Delivery with detour



Transport

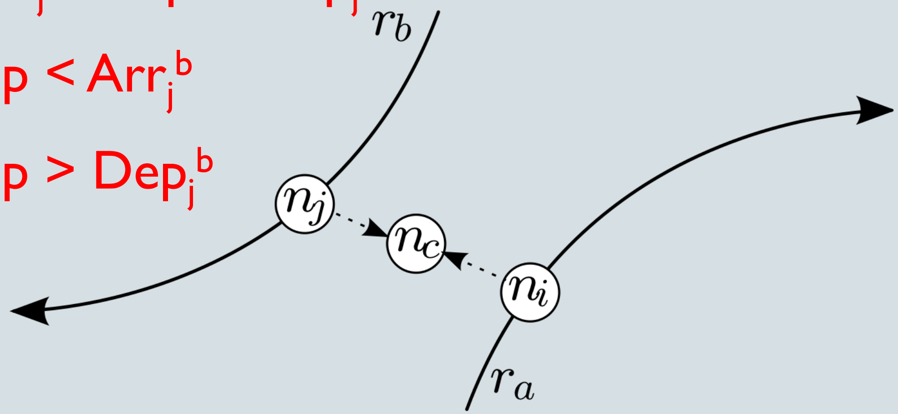


Transfer without detour

If $Arr_j^b < Cp < Dep_j^b$

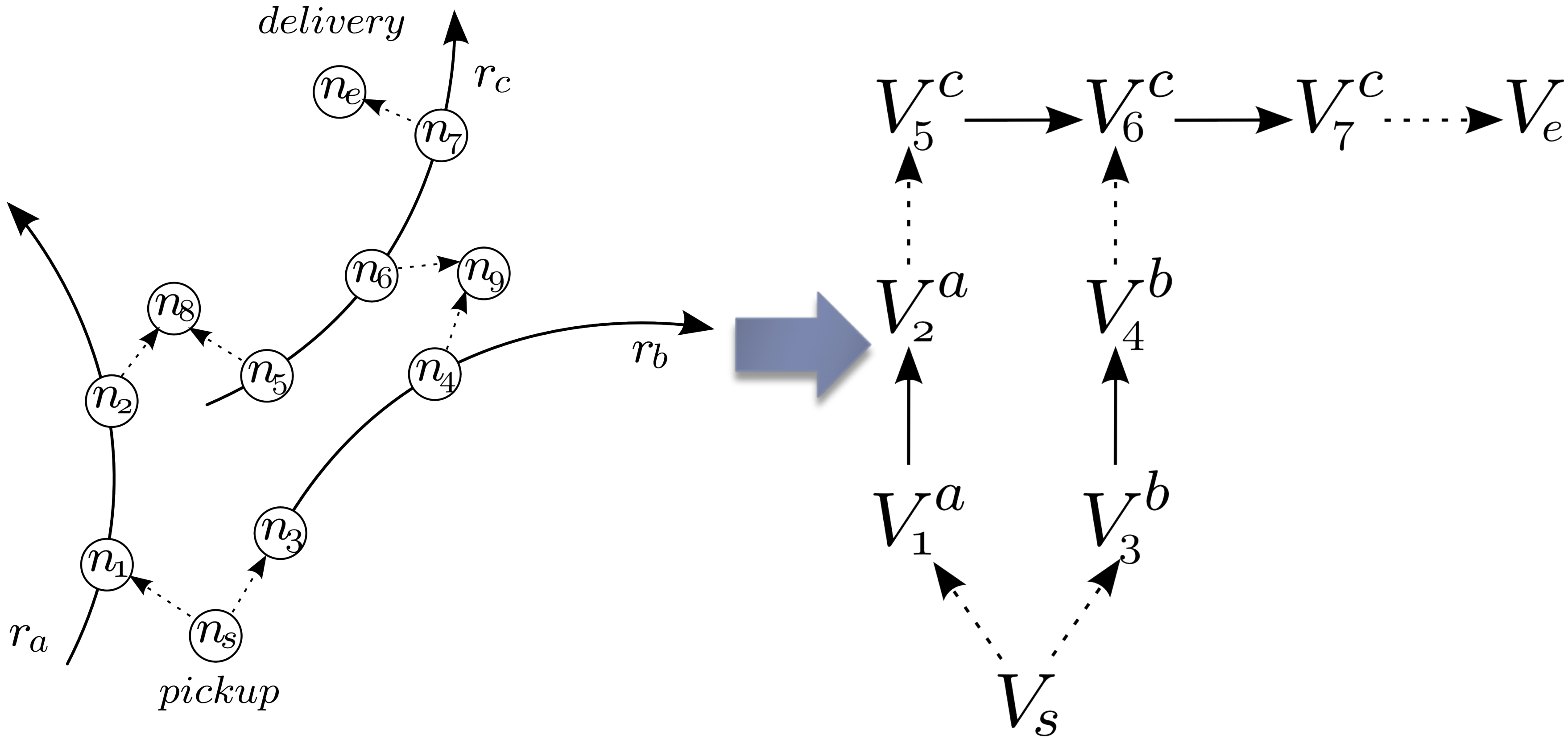
If $Cp < Arr_j^b$

If $Cp > Dep_j^b$



Transfer with detour

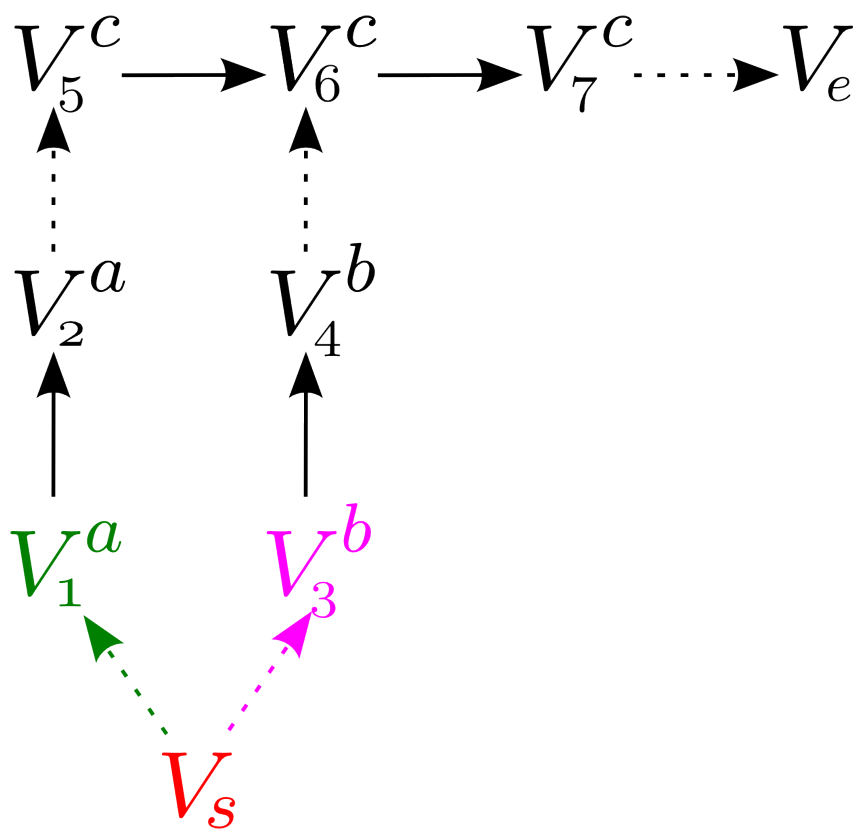
Dynamic plan graph



The SP algorithm

- Shortest path on dynamic plan graph
- BUT:
  - Dynamic plan graph violates subpath optimality
    - Answer path $(V_s,\ldots,V_i,\ldots,V_e)$ to dPDPT(n_s,n_e) does not contain answer path $(V_s,\ldots,V_i)$ to dPDPT(n_s,n_i)
  - Cannot adopt Dijkstra or Bellman-Ford
- The SP algorithm
  - Label-setting for two-criteria, Op and Cp
    - A label $<V_i^a,p,Op,Cp>$ for each path to V_i^a
  - At each iteration select label with lowest combined cost
  - Compute candidate answer – upper bound
    - When a delivery edge is found
    - Prune search space
    - Terminate search

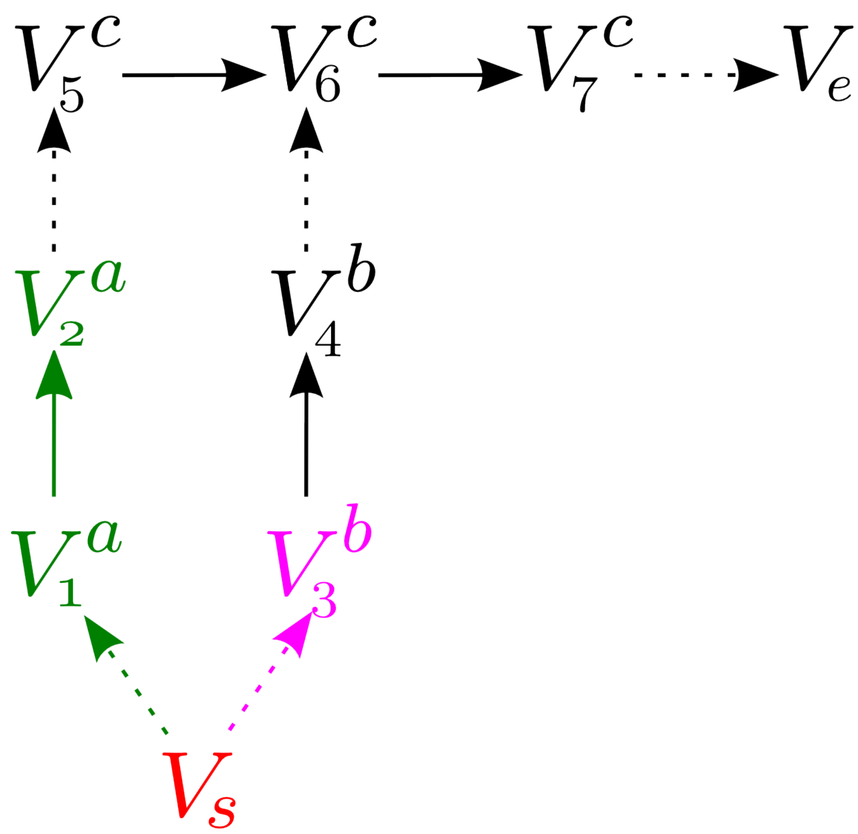
The SP algorithm (cont'd)



Detour cost T = 6

- INITIALIZATION
- CONSIDER pickup E_{s1}^a and E_{s3}^b
- Q = {<V_1^a, (V_s,V_1^a),6,16>, <V_3^b,(V_s,V_3^b),6,36>}
- P_{cand} = null

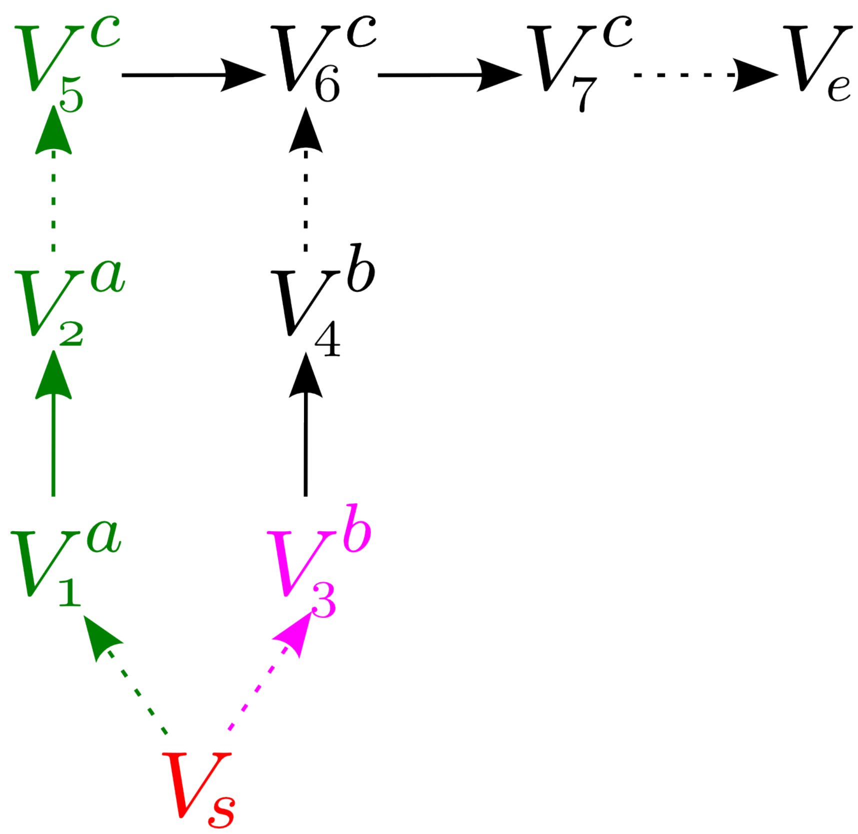
The SP algorithm (cont'd)



Detour cost T = 6

- POP $<V_1^a, (V_s, V_1^a), 6, 16>$
- CONSIDER transport E_{12}^a
- $Q = \{<V_2^a, (V_s, V_1^a, V_2^a), 6,26>, <V_3^b, (V_s, V_3^b), 6, 36>\}$
- P_{cand} = null

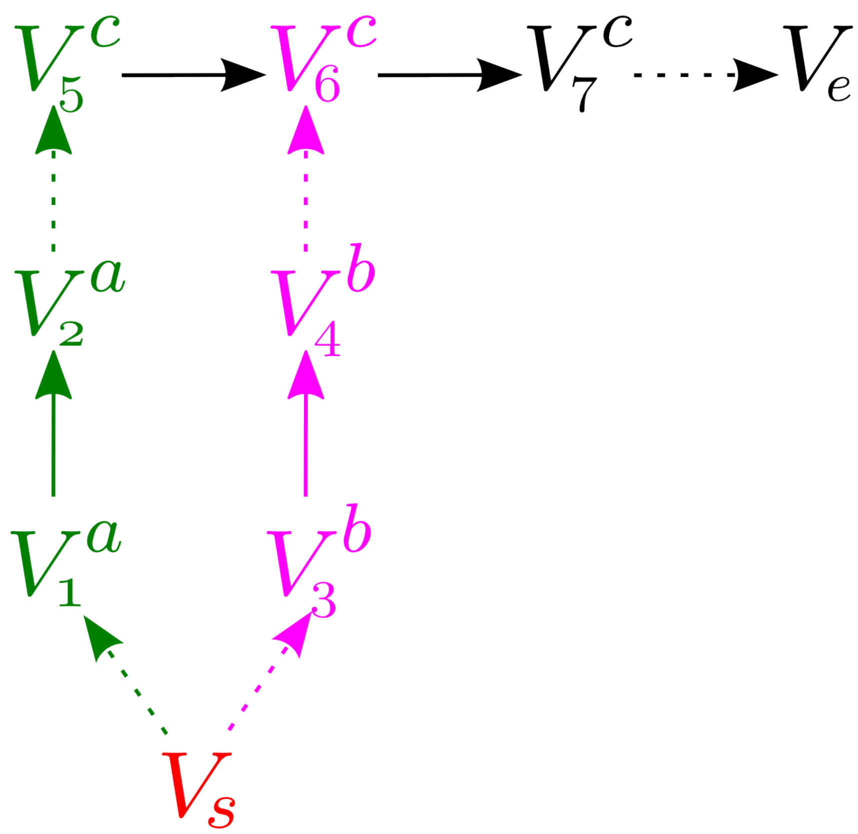
The SP algorithm (cont'd)



Detour cost T = 6

- POP <V_2^a, (V_s,V_1^a,V_2^a), 6,26>
- CONSIDER transfer E_{25}^{ac}
- $Arr_5^c = 10 < 26 < Dep_5^c = 40$
- Q = {<V_3^b,(V_s,V_3^b),6,36>, <V_5^c, (V_s,V_1^a,V_2^a,V_5^c), 18,36>}
- P_{cand} = null

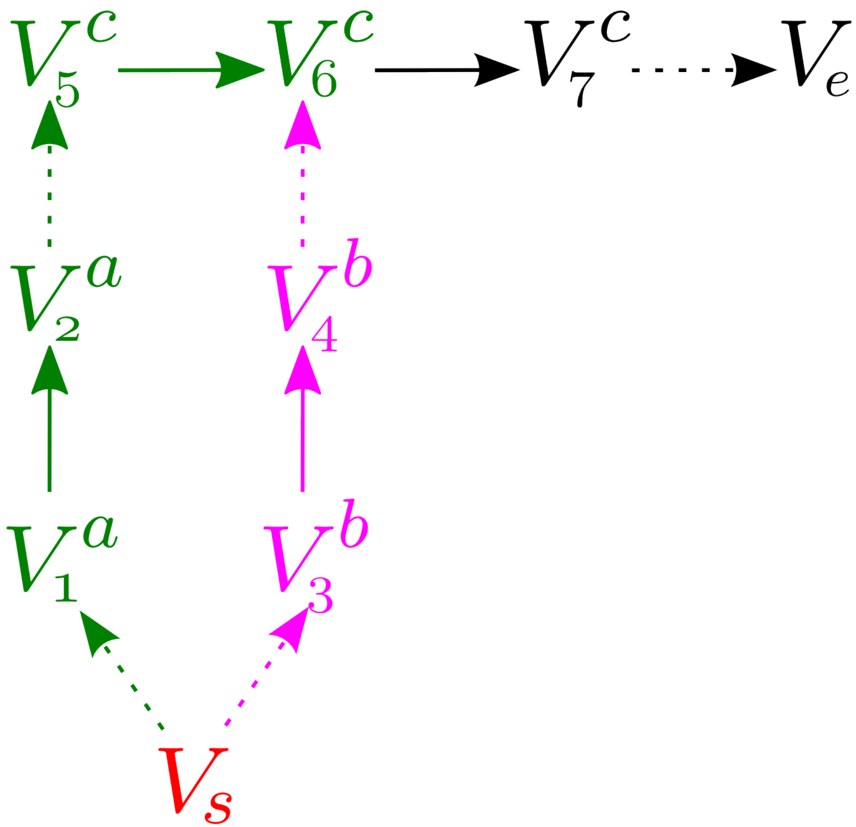
The SP algorithm (cont'd)



Detour cost T = 6

- POP $<V_3^b, (V_s,V_3^b),6,36>$ and $<V_4^b, (V_s,V_3^b,V_4^b), 6,46>$
- CONSIDER transport E_{34}^b and transfer E_{46}^{bc}
- $46 > Dep_6^c = 40$
- $Q = \{<V_5^c,(V_s,V_1^a,V_2^a,V_5^c), 18,36>, <V_6^c, (V_s,V_3^b,V_4^b,V_6^c),24,52>\}$
- P_{cand} = null

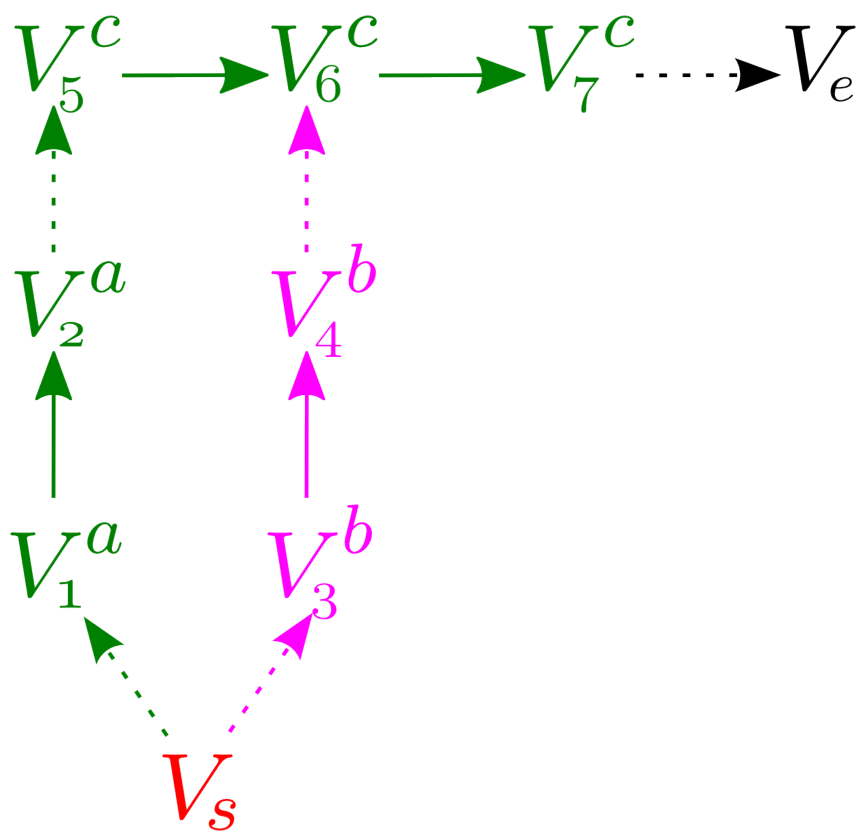
The SP algorithm (cont’d)



Detour cost T = 6

- POP $<V_5^c,(V_s,V_1^a,V_2^a,V_5^c), 18,36>$
- CONSIDER transport E_{56}^c
- $Q = \{<V_6^c, (V_s,V_1^a,V_2^a,V_5^c,V_6^c), 18, 46>, <V_6^c,(V_s,V_3^b,V_4^b,V_6^c), 24,52>\}$
- P_{cand} = null

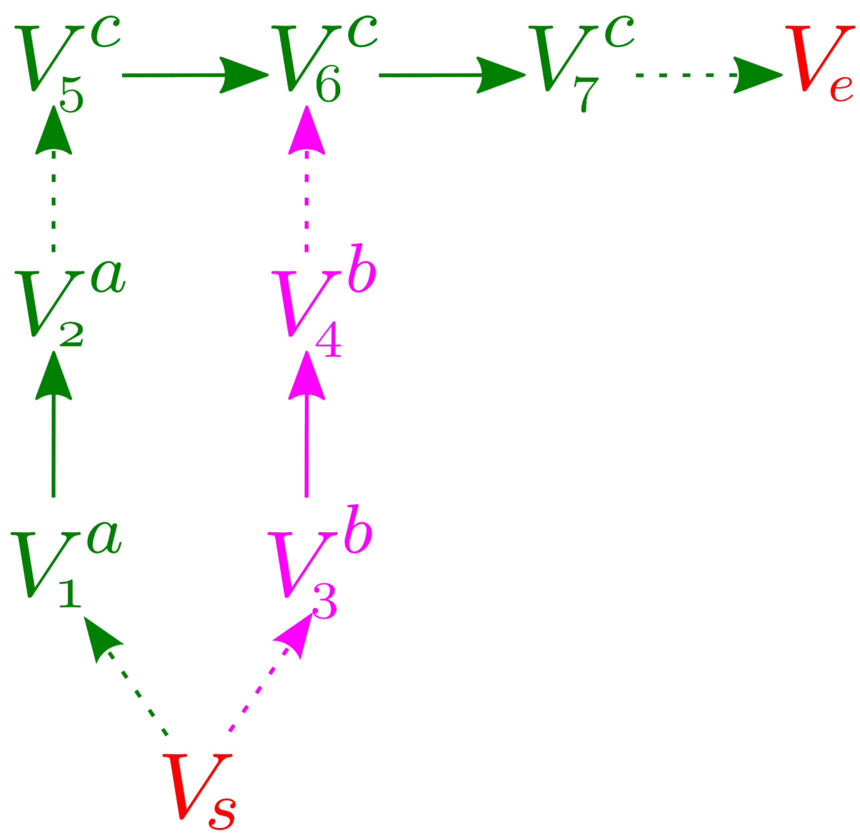
The SP algorithm (cont'd)



Detour cost T = 6

- POP $<V_6^c, (V_s, V_1^a, V_2^a, V_5^c, V_6^c), 18, 46>$
- CONSIDER transport E_{67}^c
- Q = $\{<V_7^c, (V_s, V_1^a, V_2^a, V_5^c, V_6^c, V_7^c), 18, 56>, <V_6^c, (V_s, V_3^b, V_4^b, V_6^c), 24, 52>\}$
- P_{cand} = null

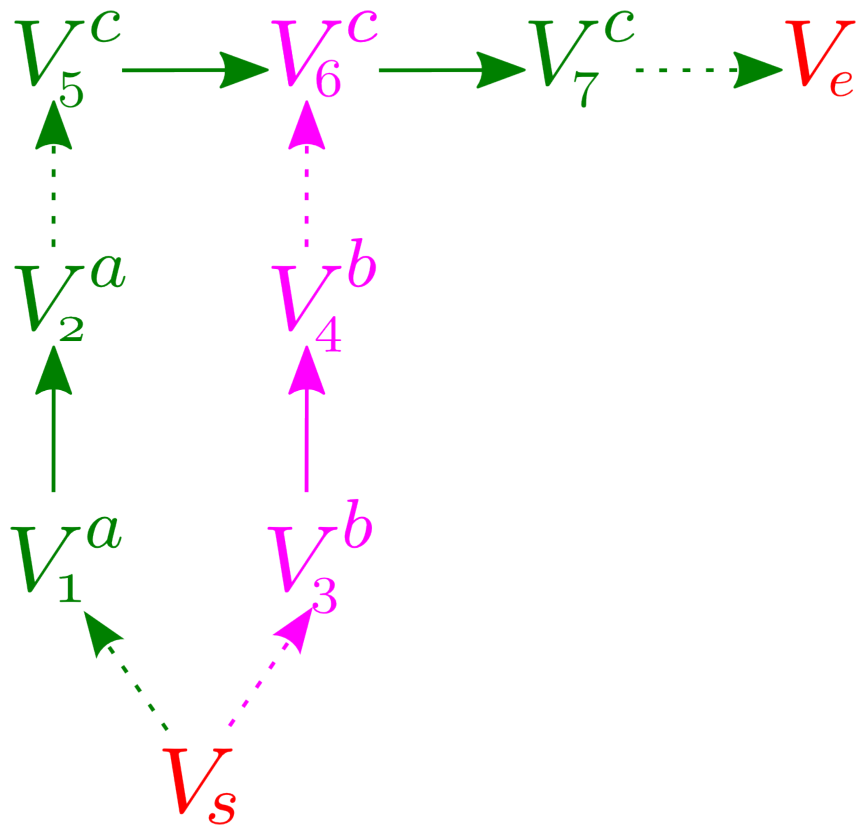
The SP algorithm (cont'd)



Detour cost T = 6

- POP <V_7^c, (V_s,V_1^a,V_2^a,V_5^c,V_6^c,V_7^c), 18,56>
- CONSIDER delivery $E_{7_e}^c$
- FOUND P_{cand}
- Q = {<V_6^c,(V_s,V_3^b,V_4^b,V_6^c), 24,52>}
- P_{cand} = (V_s,V_1^a,V_2^a,V_5^c,V_6^c,V_7^c,V_e)
- Op_{cand} = 24
- Cp_{cand} = 59

The SP algorithm (cont'd)



- POP <V_6^c,(V_s,V_3^b,V_4^b,V_6^c), 24,52>
- Op_{cand} = 24
- STOP

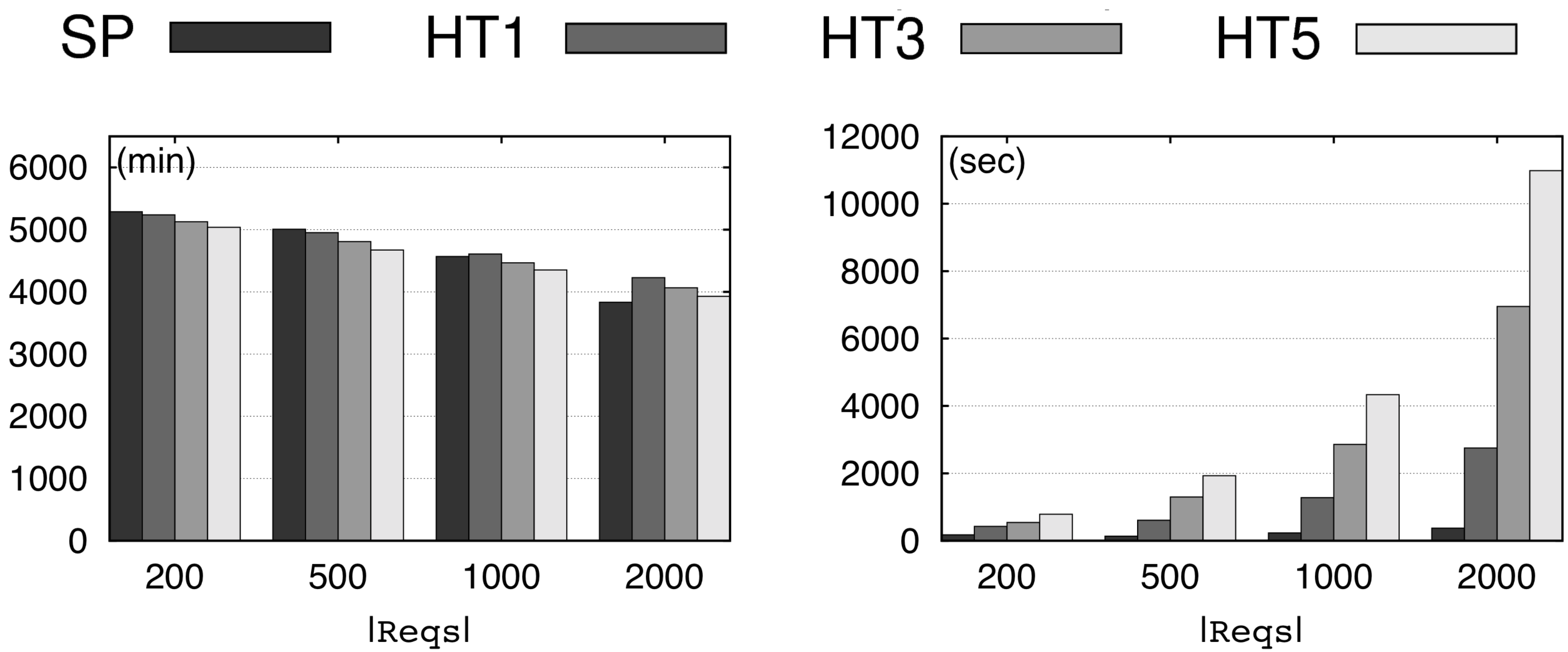
Detour cost T = 6

Experimental analysis

- Rival: two-phase method, HT
  - Cheapest insertion for pickup and delivery location, for every new request
  - After k requests perform tabu search
- Datasets
  - Road networks, OL with 6105 locations, ATH with 22601 locations
  - Static plan with HT method
    - Vary |Reqs| = 200, 500, **1000**, 2000
    - Vary |R| = 100, 250, **500**, 750, 1000
  - Stored on disk
- Experiments
  - 500 dPDPT requests
  - HT1, HT3, HT5
- Measure
  - Total operational cost increase
  - Total execution time
  - 10% cache

Varying |Reqs|

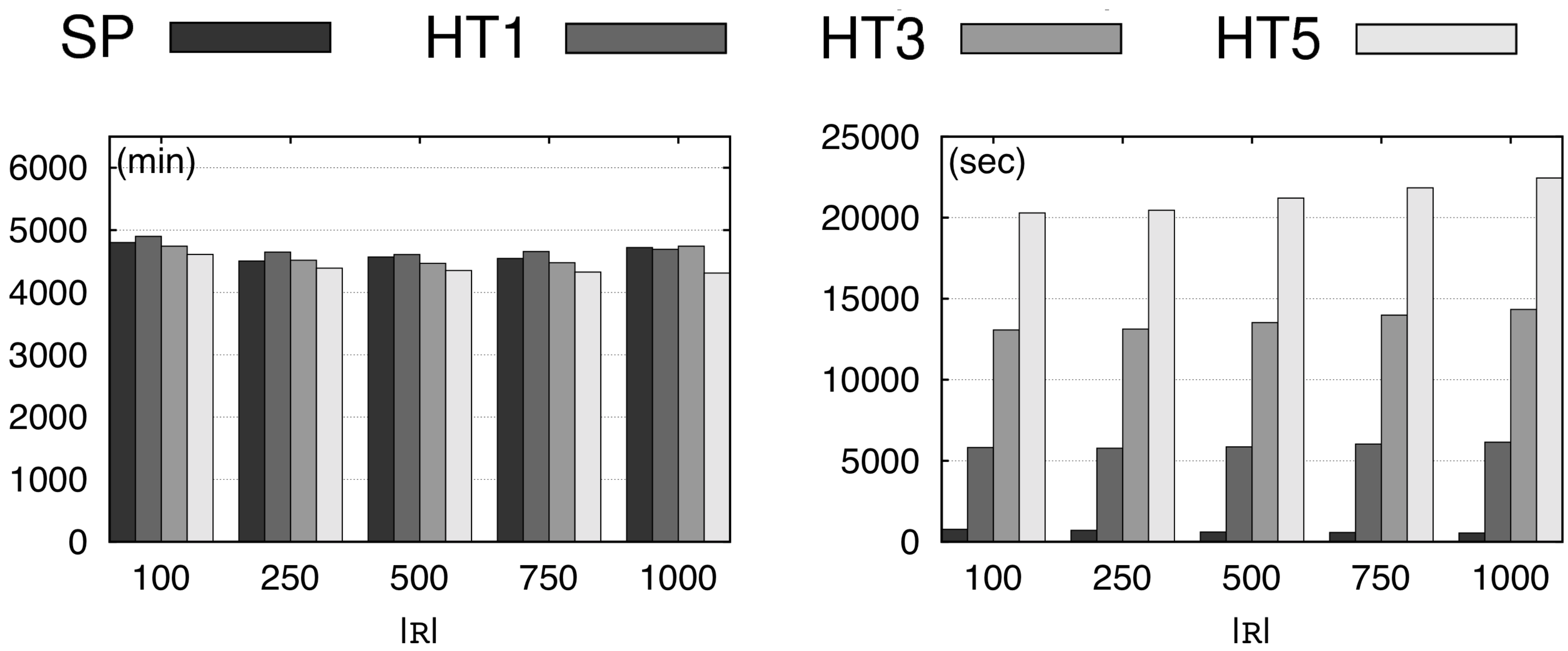
Operational cost increase | **Execution time**



OL road network

Varying |R|

Operational cost increase **Execution time**



OL road network

To sum up

- Conclusions
  - First work on dPDPT
  - Formulation as graph problem
  - Solution as dynamic two-criterion shortest path
  - Faster than a two-phase local search-based method, solutions of marginally lower quality
- Future work
  - Subpath optimality
  - Exploit reachability information within routes
  - Additional constraints, e.g., vehicle capacity

Questions ?

customer cost
pickup
delivery
operational cost
dynamic
transfer
two-criterion
detour
shortest path