# Exact and Approximate Algorithms for Finding k-Shortest Paths with Limited Overlap


Theodoros Chondrogiannis[1] Panagiotis Bouros[2] Johann Gamper[1] Ulf Leser[3]

[1]Free University of Bozen-Bolzano, Italy [2]Aarhus University, Denmark [3]Humboldt-Universität zu Berlin, Germany


## MOTIVATION

Finding multiple short yet different routes between two locations in a road network is a problem with various real-world applications:

- ✓ Commercial Route Planners
- ✓ Evacuation planning
- ✓ Humanitarian aid

## k-SPwLO PROBLEM

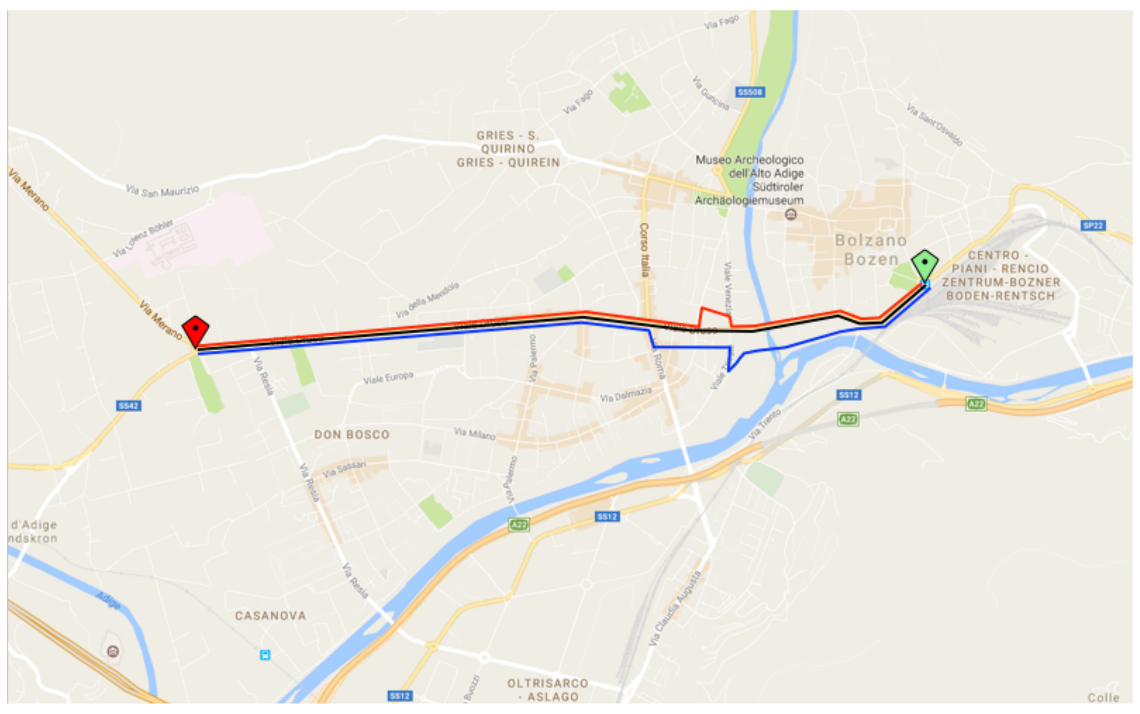
Given a source *s* and a target *t*, the **k-SPwLO** is a set of *k* paths from *s* to *t*, sorted by length in increasing order, such that:

(a) the set includes the shortest path $p_0(s \rightarrow t)$,

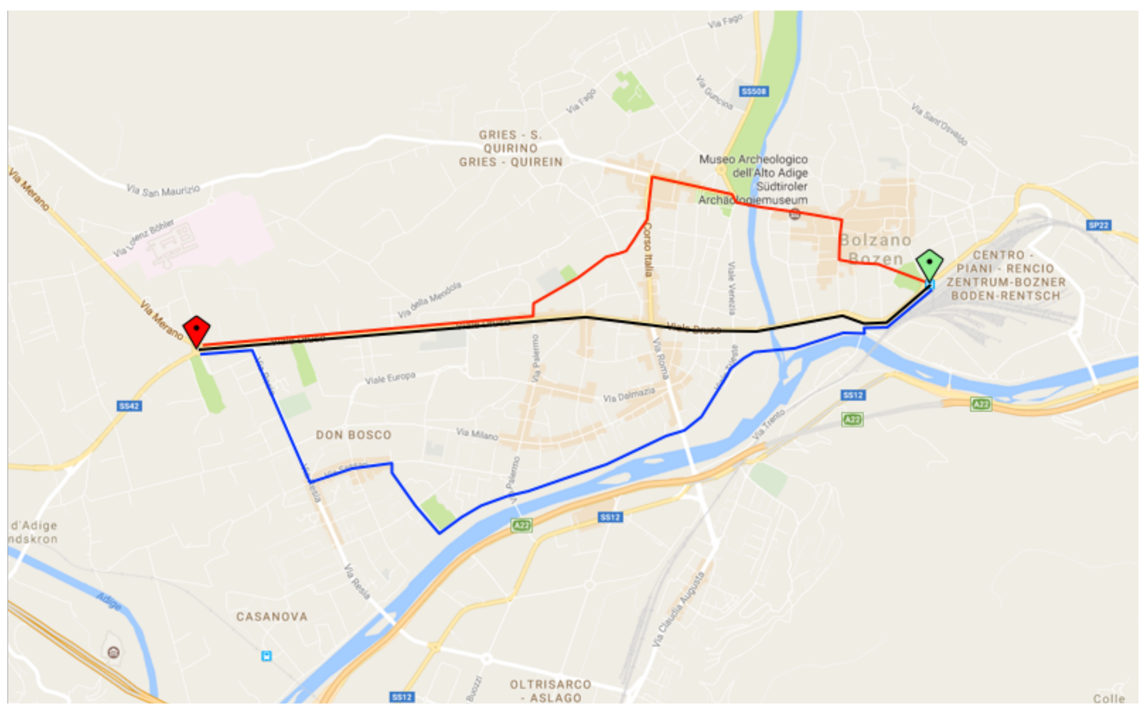
(b) every path is dissimilar to its predecessors w.r.t. a similarity threshold $\theta$,

(c) all *k* paths are as short as possible.

## EXAMPLE



K-SP: Short but very similar paths



k-SPwLO: Longer but dissimilar paths

**Result Set 1 - K-Shortest Paths**

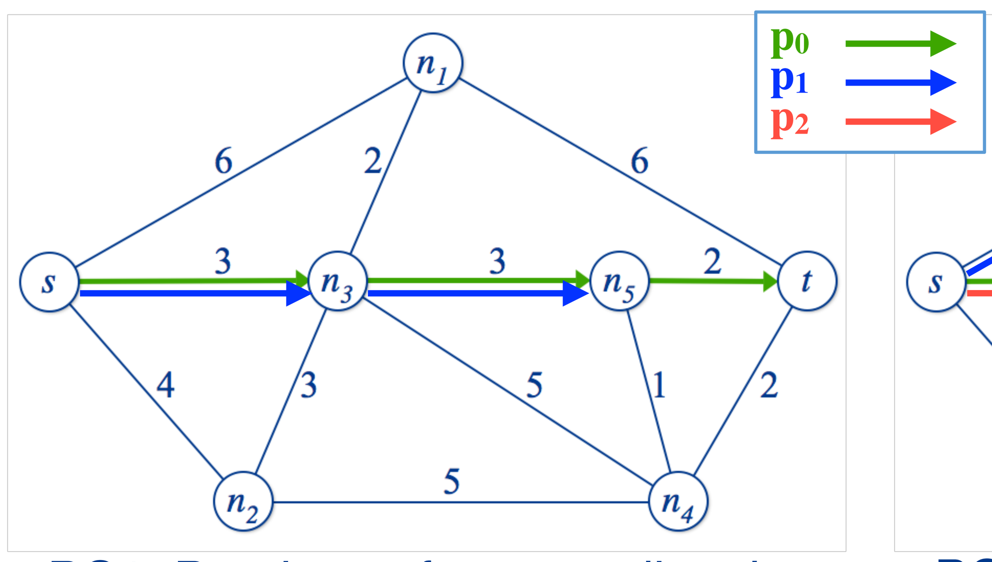
| *p* | *ℓ* | *Sim(p,p₀)* | *Sim(p,p₁)* |
|---|---|---|---|
| **Shortest path** | 3.4 km | - | - |
| **1st alternative** | 3.6 km | 95% | - |
| **2nd alternative** | 3.7 km | 80% | 76%% |

**Result Set 2 - kSPwLO ($\theta$ = 50%)**

| *p* | *ℓ* | *Sim(p,p₀)* | *Sim(p,p₁)* |
|---|---|---|---|
| **Shortest path** | 3.4 km | - | - |
| **1st alternative** | 4.1 km | 47% | - |
| **2nd alternative** | 4.5 km | 20% | 6% |

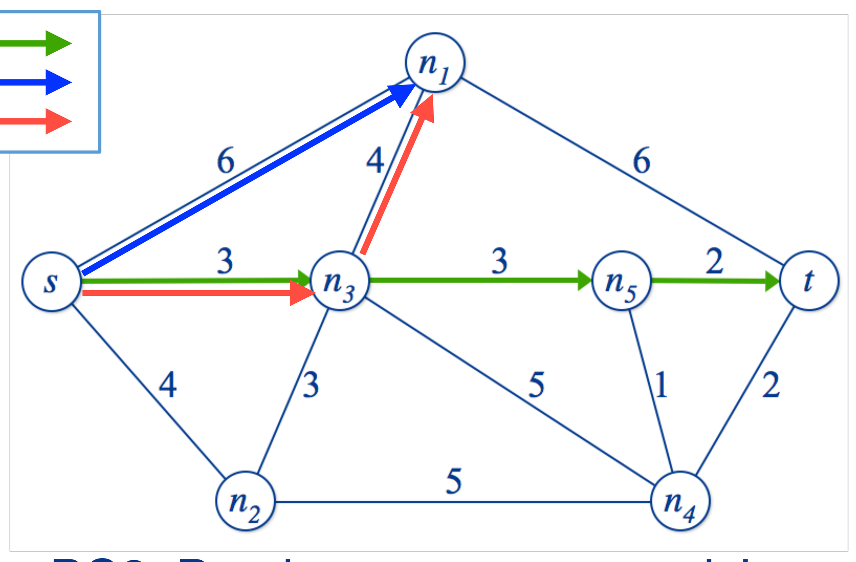
## EXACT ALGORITHM

- **MultiPass**
  - ✓ Traverses the road network k-1 times
  - ✓ Restarts the expansion after each alternative path is found
  - ✓ Employs two powerful pruning criteria





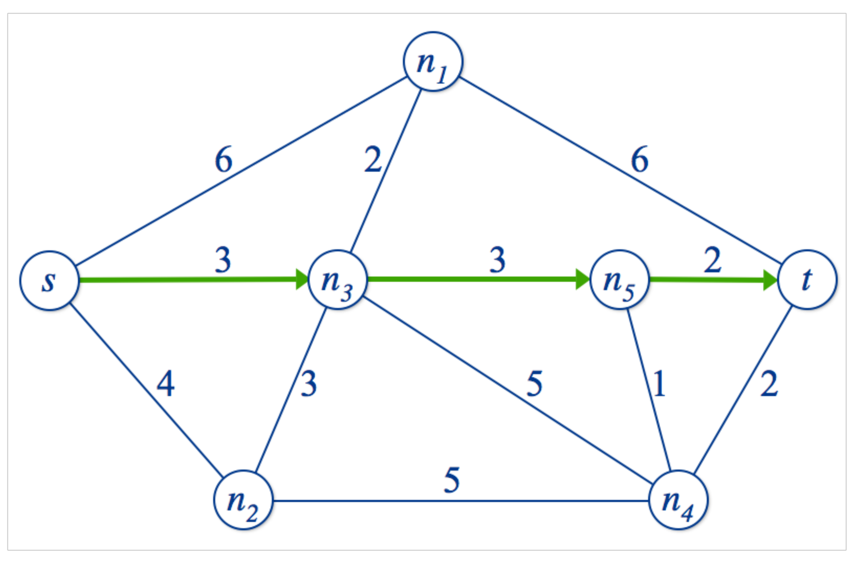
PC1: Pruning $p_1$ for exceeding the similarity threshold $\theta$=50%



PC2: Pruning $p_2$ as non-promising

## APPROXIMATE ALGORITHMS

- **OnePass+**
  - ✓ Traverses the road network once
  - ✓ Prunes paths with both PC1 and PC2
  - ✓ Does not guarantee that the exact solution will be found
- **ESX (Edge Subset Exclusion)**
  - ✓ Iteratively removes edges from the road network that lie on some already computed alternative path
  - ✓ Computes the shortest path on the updated graph
  - ✓ Continues until a sufficiently dissimilar path is found



1. **Remove edge ($n_5$,*t*)**

   $p_1 = \langle (s,n_3)(n_3,n_5)(n_5,n_4)(n_4,t) \rangle$

   ***Path p1 is not an alternative***

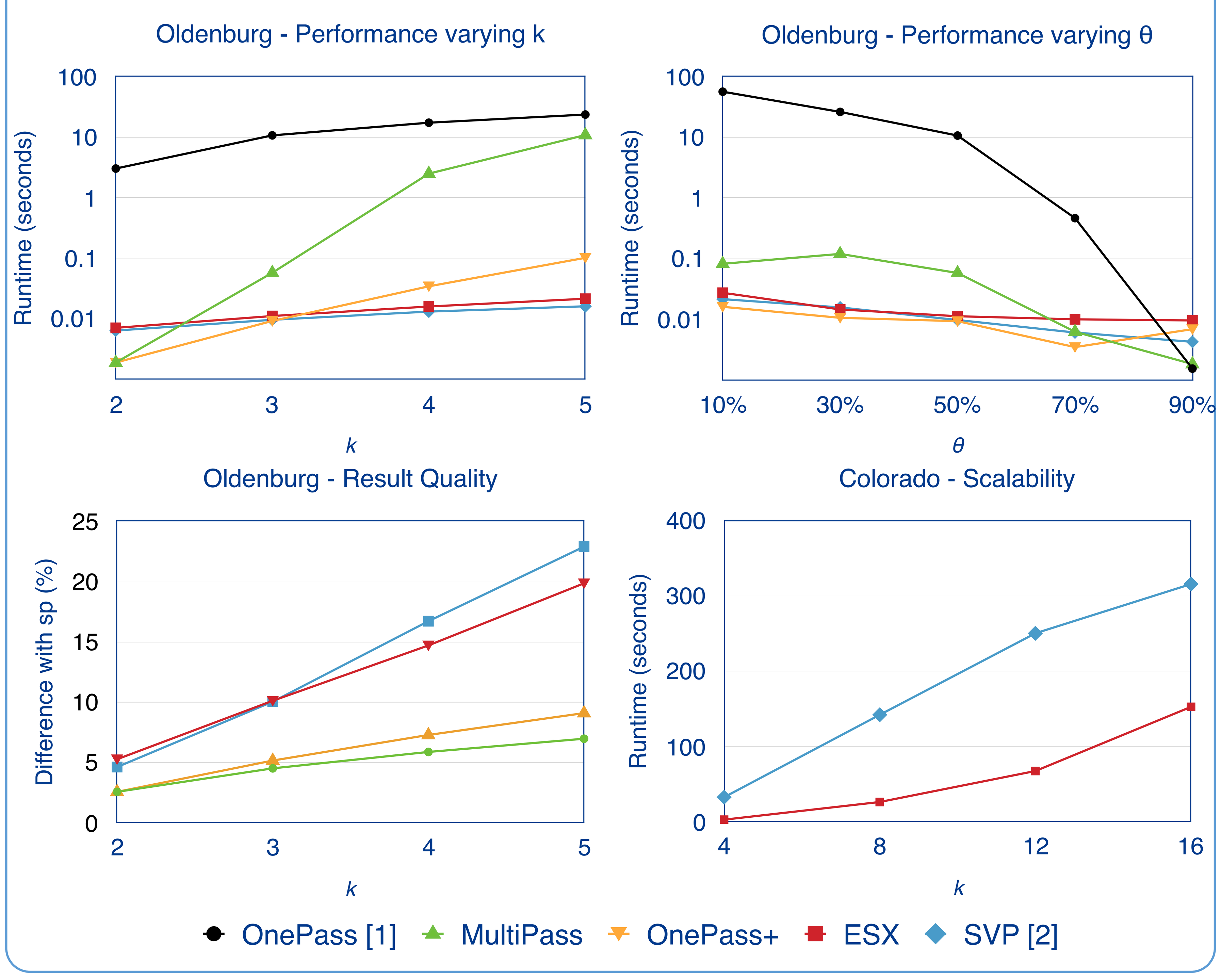
2. **Remove edge ($n_3$,$n_5$)**

   $p_2 = \langle (s,n_3)(n_3,n_5)(n_5,n_4)(n_4,t) \rangle$

   ***Path p2 is an alternative***

Sample execution of ESX ($\theta$=50%)

## EXPERIMENTAL EVALUATION





## SUMMARY

**MultiPass:** computes the optimal result but is practical only for small road networks

**OnePass+:** good approximation and practical for larger road networks than MultiPass

**ESX:** less accurate but practical even for large road networks and large values of *k*


[1] T. Chondrogiannis, P. Bouros, J. Gamper, U. Leser. Alternative routing: k-shortest paths with limited overlap. *In Proc. of the 23rd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 68:1–68:4, 2015.

[2] I. Abraham, D. Delling, A.V. Goldberg, R.F. Werneck. Alternative routes in road networks. *Journal of Experimental Algorithmics*, 18:1–17, 2013.