Tour4Me: A Framework for Customized Tour Planning Algorithms

Kevin Buchin
TU Dortmund
Dortmund, Germany
kevin.buchin@tu-dortmund.de

Mart Hagedoorn
TU Dortmund
Dortmund, Germany
mart.hagedoorn@tu-dortmund.de

Guangping Li
TU Dortmund
Dortmund, Germany
guangping.li@tu-dortmund.de

ABSTRACT

The arc orienteering problem aims to find a route which maximizes the total edge profit without exceeding a given budget. We introduce the touring problem, which additionally optimizes attributes not depending only on single edges, for example maximizing the total area covered by an tour. In this demonstration paper we provide a framework that allows for straightforward integration of new algorithms for the touring problem. We have included a new exact solver, a heuristic, and two greedy methods. Furthermore, a GUI is provided in the form of a webpage, allowing for testing of algorithms even by end users.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

datasets, neural networks, gaze detection, text tagging

ACM Reference Format:
Kevin Buchin, Mart Hagedoorn, and Guangping Li. 2018. Tour4Me: A Framework for Customized Tour Planning Algorithms. *J. ACM* 37, 4, Article 111 (August 2018), 1 page. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Most people who do outdoor activities run into the problem of finding an appropiate route. Depending on the activity from hiking and jogging to gravel and road cycling, requirements from users can greatly vary. To this end we have developed tourGenerator. The tool tourGeneratorconstists out of an intuitive UI that allows users to create tours customed to their specific demands in their own webbrowser. Furthermore, tourGeneratorcontains a few algorithms for computing solutions for the arc orienteering problem (AOP) and the more general touring problem.

Authors' addresses: Kevin Buchin, TU Dortmund, Dortmund, Germany, kevin.buchin@tu-dortmund.de; Mart Hagedoorn, TU Dortmund, Dortmund, Germany, mart.hagedoorn@tu-dortmund.de; Guangping Li, TU Dortmund, Dortmund, Germany, guangping.li@tu-dortmund.de.

0004-5411/2018/8-ART111 $15.00
https://doi.org/XXXXXXX.XXXXXXX

1.1 Related Work

1.2 Contribution

2 SYSTEM

2.1 Architecture

2.2 Data

2.3 Interface

3 ALGORITHM

3.1 Greedy Selection

3.2 Jogging Tour

3.3 Iterative Local Search

3.4 Integer Linear Programming

The integer linear program (ILP) gives the optimal solution for an instance of the AOP. The ILP used in tourGeneratoris a modified version from Verbeeck et al. The ILP from [] introduces a constraint for every subset of the vertices in order to avoind disconnected components, resulting in $O(2^n)$ constrains.

$$somethingBad \tag{1}$$

The ILP from [] uses Equation 1 to avoid subcycles. Instead we introduce a variable ρ_{kij}, for $1 \leq k \leq L$ and $1 \leq i, j \leq m$. Variable ρ_{kij} denotes whether edge e_{ij} is included in the path at location k.

$$\sum_{i=1}^{m} \sum_{j=1}^{m} \rho_{kij} = 1 \qquad \forall 1 \leq k \leq L \tag{2}$$

$$\sum_{k=1}^{L} \rho_{kij} = \begin{cases} h_{ij} & \text{if } e_{ij} \text{ is an edge} \\ 0 & \text{otherwise} \end{cases} \qquad \forall 1 \leq i, j \leq m \tag{3}$$

$$2 \cdot \rho_{kij} \leq p[k][i] + p[k+1][j] \tag{4}$$

We include Constraint 2 for every $1 \leq k \leq L$ so that the path only has one edge at every position.

Constraint 3.

4 CONCLUSION