



# The Green Paths Route Planning Software for Exposure-Optimised Active Travel

SOFTWARE  
METAPAPER

JOOSE HELLE

AGE POOM

ELIAS WILLBERG

TUULI TOIVONEN

\*Author affiliations can be found in the back matter of this article

]u[ubiquity press

## ABSTRACT

Green Paths is a prototype of route planning software for finding exposure-optimised routes for active travel. It incorporates external data on environmental exposures, including traffic noise levels, air quality, and street-level greenery into the street and paths network produced by the OpenStreetMap project. Written in the Python programming language, the software applies a novel environmental impedance function in the least cost path routing to find exposure-optimised routes. Routes for externally defined origin-destination pairs can be queried via a RESTful API. The API returns alternative routes equipped with rich exposure data. The published version of the software has been applied in population level environmental exposure assessment and in an end-user-oriented web-based route planner application designed for use in the Helsinki Metropolitan Area.

## CORRESPONDING AUTHORS:

### Joose Helle

Digital Geography Lab,  
Department of Geosciences  
and Geography, University of  
Helsinki, Gustaf Hällströmin  
katu 2, 00014 Helsinki, Finland  
[joose.helle@gmail.com](mailto:joose.helle@gmail.com)

### Tuuli Toivonen

Digital Geography Lab,  
Department of Geosciences  
and Geography, University of  
Helsinki, Gustaf Hällströmin  
katu 2, 00014 Helsinki, Finland  
[tuuli.toivonen@helsinki.fi](mailto:tuuli.toivonen@helsinki.fi)

---

## KEYWORDS:

Routing; environmental  
exposure; urban mobility;  
travel; walking; cycling; air  
quality; traffic noise; street-  
level greenery; spatial analysis

## TO CITE THIS ARTICLE:

Helle J, Poom A, Willberg E,  
Toivonen T 2023 The Green  
Paths Route Planning Software  
for Exposure-Optimised  
Active Travel. *Journal of Open  
Research Software*, 11: 10.  
DOI: [https://doi.org/10.5334/  
jors.400](https://doi.org/10.5334/jors.400)

## (1) OVERVIEW

### INTRODUCTION

During urban trips, pedestrians and cyclists are exposed to the harms and benefits of the urban travel environment, such as traffic noise, air pollution, and greenery. As environmental conditions vary geographically in urban spaces, route choice becomes an important factor in determining environmental exposure during active travel [1, 2]. In comparison to taking the shortest or fastest routes, environmentally sensitive route planning is likely to reduce harmful exposures from travel and increase health benefits [3–6].

Transport, environmental, and public health research has witnessed an emergence of exposure-optimised routing studies during the last decade [1]. The research field supports spatial and transport planning in creating sustainable and healthy urban space and travel options [2, 7]. It also helps to advance conventional route planners that have mostly focused on minimising time, distance, or the monetary cost of travel [8]. Informed decision-making among end-users is supported by enabling access to environmental information about possible routes. Namely, pre-trip information about route qualities affects route choice if travellers perceive that the information is useful for their welfare [9, 10].

Most routing studies on exposure optimisation have focused on finding low-exposure routes in respect to air pollution [e.g., 4, 5, 11–14]. Other studies that optimise routes also consider traffic noise levels or the presence of greenery [6, 8, 15–17], allergens [18], or extreme environmental conditions [19, 20]. Several groups of researchers have developed new routing algorithms or prototype route planners for finding exposure-optimised routes [6, 18, 19, 21].

Despite these efforts, real-world evidence on functional route planners that find exposure-optimised routes for active travel is limited. The few examples involve Clean Air Route Finder<sup>1</sup> that provides multiple choices for walking and cycling routes in London. The tool provides information about trip length, travel time, and the relative decrease in air pollution in comparison to the shortest route [22]. An open-source mobile application called *hRouting*, The Health-Optimal Route Planner, enables routes in Zürich with lower exposure to ultrafine particles to be found [23]. Some previous route planners in Canadian cities offered cyclists the opportunity to choose a route with better air quality [24–26]. Another temporarily available routing tool, a Clean Air Route Planner, was introduced in Helsinki, where walking routes were optimised according to precise air quality information [27]. In Amsterdam, the Healthy Urban Route Planner (HURP) project developed a route planner for cyclists and pedestrians to reduce exposure to air pollution.<sup>2</sup> A Healthy Places routing app is soon to be

launched with the aim to minimize air pollution exposure during walking.<sup>3</sup>

To advance the research field with open-source tools, we developed a prototype Green Paths route planning software for active travel. It enables evaluation of environmental exposure during walking and cycling, study access to healthy travel routes, and empower citizens in making health-benefitting route decisions. The open-source routing software demonstrates how least cost path routing can be made sensitive to three environmental variables—traffic noise, air quality, and street-level greenery—in an urban context. It optimises either walking or cycling routes by combining data about environmental conditions with travel time. The Green Paths software provides multiple route suggestions between externally defined origin and destination pairs, from the fastest route to the ones with less noise and air pollution or more green views. The routing results are provided via a RESTful Application Programming Interface (API) to be used either in research or for end-user routing tools and other applications. The routing software has been developed to run in the Helsinki Metropolitan Area. For other urban areas, custom data acquisition and processing steps would be needed prior to running the application. The technical documentation of the software, which is packaged within the sourced code repository and thus available both on GitHub and via the DOI (see the Availability section), includes more detailed requirements and further guidance for preparing the environmental data for other areas.

The Green Paths routing software shares some similarities with the previously developed Clean Air Route Planner for Helsinki [27]. Both applications use similar external data about air quality [28, 29] and facilitate exposure-optimised routing for walking. However, there are significant differences in terms of the scope, functionality, and applicability of the Green Paths software. For example, both walking and cycling are available as travel modes and the routing functionality is based on an original environmental impedance function. The latter combines travel time, environmental cost, and sensitivity coefficients, and provides multiple routing cost alternatives with different trade-offs between exposure benefits and travel time costs as a result. The concept and technical implementation of the environmental impedance function is explained in detail in the Implementation and architecture section of this paper.

The aim of this paper is to describe the methodology and applicability of the Green Paths routing software. The paper provides a comprehensive view of the software, including its key features and technical architecture, and complements the complete technical documentation of the software available from GitHub (included in the source code repository). The paper does not cover the user interface of the Green Paths routing tool. The

paper is organized as follows: the Implementation and architecture section describes the technical context, architecture, external data, packages, and workflow of the software. The Quality control section describes the tests of routing functionality. The Availability section describes the system requirements to run the software, contributors, and software availability. The Reuse potential section describes the current application of the software in the Helsinki Metropolitan Area and replicability potential to other regions.

## IMPLEMENTATION AND ARCHITECTURE

### Technical context

The Green Paths software is a prototype for exposure-optimised routing engines. It is an open-source software written in the Python programming language. Python provided a suitable technical base for fast prototyping of different routing features and utilising several third-party libraries. The Green Paths software has been built as prototype for demonstrating path finding equipped with rich exposure metrics between selected origin-destination pairs. Thus, it has not been optimised to provide a routing service for extensive production use.

The complete technical documentation of the software has been published along with the source code and can be accessed via its primary README.md file. The technical documentation includes a straightforward installation guide for the software as well as other supporting technical information, such as sample graph data for testing and developing the software.

Table 1 lists the third-party Python libraries that the Green Paths software depends on, excluding the sub-dependencies of the libraries. In addition, the Green Paths software requires the use of the OpenTripPlanner (OTP) open-source routing software for pre-processing OpenStreetMap-based street network data. For further information, see the Graph build package section in this paper.

### Features and software architecture

The Green Paths software has three main features: (i) **scripts and utilities** for combining environmental data on noise, air quality, and greenery with OpenStreetMap-based street network graph; (ii) **the main routing application** for finding fastest and exposure-optimised routes for both walking and cycling; and (iii) **RESTful API** for publishing the environmentally sensitive routing functionalities as a web service. The architecture of the software, data processing flow, and the two use cases of the Green Paths routing software are illustrated in Figure 1.

The software consists of four main packages that are described in more detail in the sections below:

- 1) **graph\_build**: scripts and utilities for handling street network graph and environmental data;
- 2) **aqi\_updater**: utilities for fetching, processing, and sampling either real-time or static air quality index data for the main routing application;
- 3) **gp\_server**: the main routing application that handles routing requests;
- 4) **common** (omitted from the Figure 1): general utilities for handling geometry data and reading and writing street network graph data in the GraphML format (I/O).

### External data

The Green Paths software uses external open-access data of street network, traffic noise, air quality as Air Quality Index (AQI), and street-level greenery as Green View Index (GVI) that are described in the Table 2.

### Graph build package

The routing in the Green Paths software is based on a street network graph, which consists of nodes and edges between the nodes (representing street segments). Each edge is equipped with attributes about the geometry and distance of the edge, allowed travel modes, a biking

PACKAGE	LINK	VERSION (TESTED)	DESCRIPTION AND USE
Flask	<a href="https://github.com/igraph/python-igraph">https://github.com/igraph/python-igraph</a>	1.1.2	A simple web framework for publishing the routing functionality via APIs
GeoPandas	<a href="https://github.com/geopandas/geopandas">https://github.com/geopandas/geopandas</a>	0.8.3	Data structures for spatial data processing and integration
Gunicorn	<a href="https://github.com/benoitc/gunicorn">https://github.com/benoitc/gunicorn</a>	20.0.4	A Python WSGI HTTP Server for UNIX, used for running the APIs of the routing application as a web service
igraph (python-igraph)	<a href="https://github.com/igraph/python-igraph">https://github.com/igraph/python-igraph</a>	0.8.3	Graph data structure and built-in algorithms for solving least cost path problems
Shapely	<a href="https://github.com/Toblerity/Shapely">https://github.com/Toblerity/Shapely</a>	1.7.1	Geometry objects and overlay analysis

**Table 1** The list of third-party Python libraries used in the Green Paths routing software.

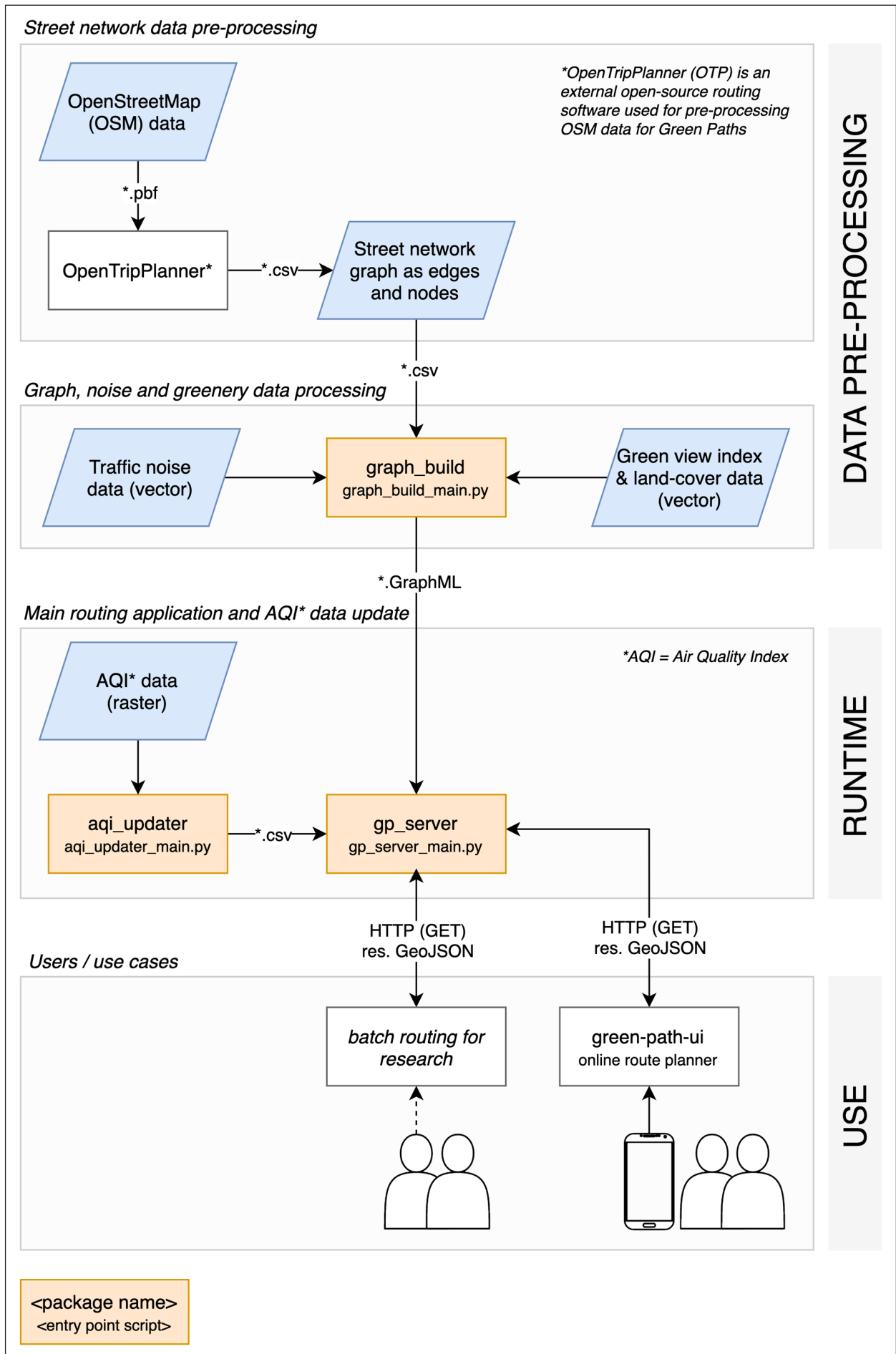


Figure 1 The architecture, data processing workflow, and use cases of the Green Paths routing software.

DATA	CONTENT	MEASURE	VALUE RANGE	FORMAT	SPATIAL EXTENT	SOURCE AND REFERENCES
Street network data	Street and path network	n/a	n/a	Standard compressed Protocolbuffer Binary Format (PBF)	Global	OpenStreetMap, <a href="http://www.openstreetmap.org">www.openstreetmap.org</a> , CC BY 2.0
Traffic noise data	Day, evening, and night A-weighted equivalent continuous sound pressure level ( $L_{den}$ ) from road and rail traffic. Modelled every five years according to the EU Environmental Noise Directive 2002/49/EC.	dB(A)	40...75+	Vector, shapefile	European urban areas	[30], accessible from <a href="https://ckan.ymparisto.fi/dataset/%7B0A7C0CF8-7BAA-49FF-835A-0B97DA89B9D4%7D">ckan.ymparisto.fi/dataset/%7B0A7C0CF8-7BAA-49FF-835A-0B97DA89B9D4%7D</a> , CC BY 4.0
Real-time air quality data	Hourly composite air quality index covering $NO_2$ , $SO_2$ , $O_3$ , $PM_{2.5}$ , $PM_{10}$ , BC, and LDSA	Air Quality Index (AQI) 2.0	1...5 (good... very poor)	Raster, mesh size $13 \times 13$ m, netCDF	Helsinki Metropolitan Region	FMI-ENFUSER 2.0 model, [29, 31], accessible from the Finnish Meteorological Institute (FMI)
Static air quality data	Annual average composite air quality index covering $NO_2$ , $SO_2$ , $O_3$ , $PM_{2.5}$ and $PM_{10}$	Air Quality Index (AQI) 1.0	1...5 (good... very poor)	Raster, mesh size $13 \times 13$ m, netCDF or TIFF	Helsinki Metropolitan Region	FMI-ENFUSER 1.0 model, [29], accessible from the Helsinki Region Environmental Services Authority (HSY)
Greenery data	Proportion of green vegetation visible at street level based on street view panoramas and regional land cover data	Green View Index (GVI)	0...1	Vector, shapefile	Helsinki Metropolitan Region	[32], CC BY 4.0; <a href="https://hri.fi/data/fi/dataset/paakaupunkiseudun-maanpeiteaineisto">hri.fi/data/fi/dataset/paakaupunkiseudun-maanpeiteaineisto</a> , CC BY 4.0

**Table 2** External data currently used in the Green Paths routing software for exposure-optimised routing.

safety factor, and environmental data. The nodes are characterised by only the geometry value. The required attributes and their value types are listed in the technical documentation of the software (see the Availability section below).

As illustrated in Figure 1, raw OpenStreetMap data in PBF format are first processed with OpenTripPlanner to create a street network graph data model. For this purpose, an executable jar-file of OpenTripPlanner is provided along with the source code of the Green Paths software. The resulting graph model with nodes, edges, and respective attributes is imported via comma-separated values (CSV format) to the Green Paths software. The `graph_build` package uses the model to construct a street network graph that fits the internal graph format of the software. The Green Paths software can read and write graph objects via GraphML file format utilising custom I/O functions, which are written on top of the `igraph` library (module `common.igraph`).

During the later stages of the graph build, static external environmental data on traffic noise and greenery are sampled and attached to the edges of the graph via a spatial overlay analysis. These scripts are implemented using mostly procedural programming style.

### AQI updater package

The `aqi_updater` package is needed for fetching, processing, and sampling Air Quality Index (AQI) values from the raster files to the street network graph. It is developed as an independent sibling package next to the `gp_server` package and can be used from the root of the project (`aqi_updater_main.py`). While the implementation of the package includes many generic functionalities, the current solution is adjusted to work in the Helsinki Metropolitan Area with the AQI data formats that are described in Table 2.

The `aqi_updater` reads graph data from the GraphML file and AQI data from a raster file (NetCDF or TIFF) and exports a CSV file that contains a value map of street IDs and respective AQI values. This value map can then be read by the `gp_server` package to update AQI values to its internal graph object at runtime.

There are two ways of using the `aqi_updater`, based on the definition in the configuration file of the `gp_server` package. The options involve either a procedural Python script for processing and sampling a local AQI raster file on demand (see the file `aq_data_import.import_static_aqi_data`) or a standalone application that fetches AQI data from a remote file server and processes and samples it on a scheduled manner. The latter use case is

suitable for using real-time AQI values as demonstrated in the online proof-of-concept instance of the Green Paths routing software for the Helsinki Metropolitan Area (<https://green-paths.web.app/>).

### GP server package

The `gp_server` package forms the main routing application of the Green Paths software and includes the following functionalities.

#### *Reading routing graph data*

The Green Paths software can find five types of routes, including the fastest, safest (for bike, following the OpenTripPlanner capabilities), quietest, and greenest routes, and routes with the freshest air. The three last environmental exposure optimised routing modes are optional. Each enabled routing mode requires the presence of corresponding edge attributes that are listed in the technical documentation. The `gp_server` package reads the necessary attribute values from the street network graph data in GraphML format, except for the air quality attribute values that are read from the CSV format output of the `aqi_updater` package.

#### *Environmental impedance function*

The software uses Dijkstra's least cost path routing algorithm, which by default is based on travel time. Travel time is derived from the distance of the edges of the graph and travel mode specific speed coefficients that are defined in the configuration file. To support exposure-optimised routing, we integrated an environmental impedance function to the travel-time-optimised routing algorithm. The function enables calculation of exposure-adjusted edge cost values (Eq. 1) that complement the travel time related base cost of edges:

$$C_e = C_t + C_t * c_e * s \quad (1)$$

where  $C_e$  is the total (composite) cost of the edge,  $C_t$  is the base cost of the edge that is proportional to travel distance and defined speed,  $c_e$  is an environmental cost coefficient, and  $s$  is an arbitrary sensitivity coefficient for assigning applicable weight(s) for the environmental cost component. The additional environmental cost  $C_t * c_e * s$  is thus proportional to the value of the environmental variable and travel time. The definition of the environmental cost coefficient  $c_e$  depends on the exposure-optimised routing mode. It reflects either the traffic noise level, AQI value, or GVI value in the value range of respective environmental data (see Table 2).

As the units of the cost components are not the same, the combined edge cost is unitless. Edge costs are used only for route optimization, whereas for example route comparison can be done with properties that have a real-world meaning, such as mean noise level.

The sensitivity coefficient  $s$  enables multiple alternative routes to be found for a single origin-destination pair. Multiple route suggestions enable responses to the variation of personal preferences, including varying sensitivity to environmental properties or different valuations of travel time. In other words, as it is challenging to define a single best route for everyone, the environmental impedance factor needs to include a sensitivity component that enables the environmental cost and travel time to be balanced. The `gp_server` package uses a configurable list of sensitivity coefficients for routing several alternative routes. Higher sensitivity coefficients increase the weight of the environmental properties in routing, whereas lower sensitivities result in routes with travel times close to the fastest route. Routing results always include the fastest route by applying a sensitivity coefficient equal to zero. Ideally, exposure-optimised routing can find route(s) of significant improvement in environmental exposure yet with only a limited increase in travel time.

The appropriate list of sensitivity coefficients for each exposure-optimised routing mode varies depending on the value range and spatial variation of the environmental variable. Environmental variables with limited spatial variation require higher sensitivity coefficients to find alternative, exposure-optimised routes.

#### *Configuration*

Some of the runtime settings of the Green Paths software can be changed via a simple configuration module in the `gp_server` package (`gp_server.conf`). With the configurations, users can (i) define, which graph file is used, (ii) disable or enable either of the two travel modes (walking/biking), (iii) disable or enable exposure-optimised routing modes one by one (traffic noise, greenery or air quality optimized routing), (iv) set the sensitivity coefficients that are used in exposure-optimised routing, (v) set default speed coefficients for travel modes, and (vi) disable or enable the research mode. The latter provides additional exposure information for each route in the results. The differences between the research mode for batch routing and the end-user mode for routing single routes are explained in the technical documentation of the API (see the Availability section).

#### *Start-up of the routing application and routing workflow*

To find walking and cycling routes via the least cost path routing algorithm, the Green Paths software applies (edge) cost calculation principles to different travel modes and street types adopted and simplified from the OpenTripPlanner. We implemented a straightforward least cost path routing method: all cost attributes are pre-calculated and assigned to edges during the start-up of the application, i.e., prior to solving the least cost path problem with Dijkstra's algorithm. This approach makes executing Dijkstra's algorithm very simple. The downside

is that the memory consumption of the application grows quickly along with the number of pre-calculated edge attributes, as an independent cost attribute is needed for each unique combination of travel mode, routing mode, and sensitivity coefficient. The basic edge cost calculation considers *length*, *allows\_walking*, and *allows\_biking* attributes parsed by OpenTripPlanner for assigning travel time-based costs (*base costs*) for the two travel modes. Walking with bike is also permitted, to allow streets to be crossed and reaching destinations in walk-only parts of the network (as in OpenTripPlanner). Next, exposure-adjusted edge costs are calculated from edge-level base costs and environmental exposure data using the environmental impedance function. Due to the pre-calculation of cost attributes, several advanced routing features that require dynamic cost calculation within Dijkstra’s algorithm are not supported. These include turning costs for cycling and traffic light penalties.

The start-up of the routing application (*gp\_server\_main.py*) initiates the following sequence of actions to prepare the application for handling routing requests:

- 1) Graph data from an external GraphML file is read into an in-memory graph object;
- 2) Edge and node geometries are read from the graph into GeoPandas GeoDataFrames for fast lookup of the nearest nodes and edges for origins and destinations;

- 3) Travel time-based costs are calculated as edge attributes separately for both travel modes (walking & cycling);
- 4) Traffic noise-based costs are calculated for both travel modes at different sensitivities and set as edge attributes;
- 5) GVI-based costs are calculated for both travel modes at different sensitivities and set as edge attributes;
- 6) AQI update data are read and updated to the graph as edge attributes if the CSV-format output either for static or real-time AQI values from the *aqi\_updater* application is available;
- 7) AQI-based costs are calculated for both travel modes at different sensitivities and set as edge attributes if the AQI values were updated to the graph under the step 6;
- 8) Steps 6 & 7 are repeated in a scheduled manner if real-time AQI data are available.

Steps 4–8 can be disabled from the configuration file of the *gp\_server* package. After successfully executing these steps, the main application is ready to accept routing requests via the API endpoints defined in the module *gp\_server\_main.py* (start-up script of the routing application). The sequence of actions that is executed during handling a single routing request is visualised in Figure 2.

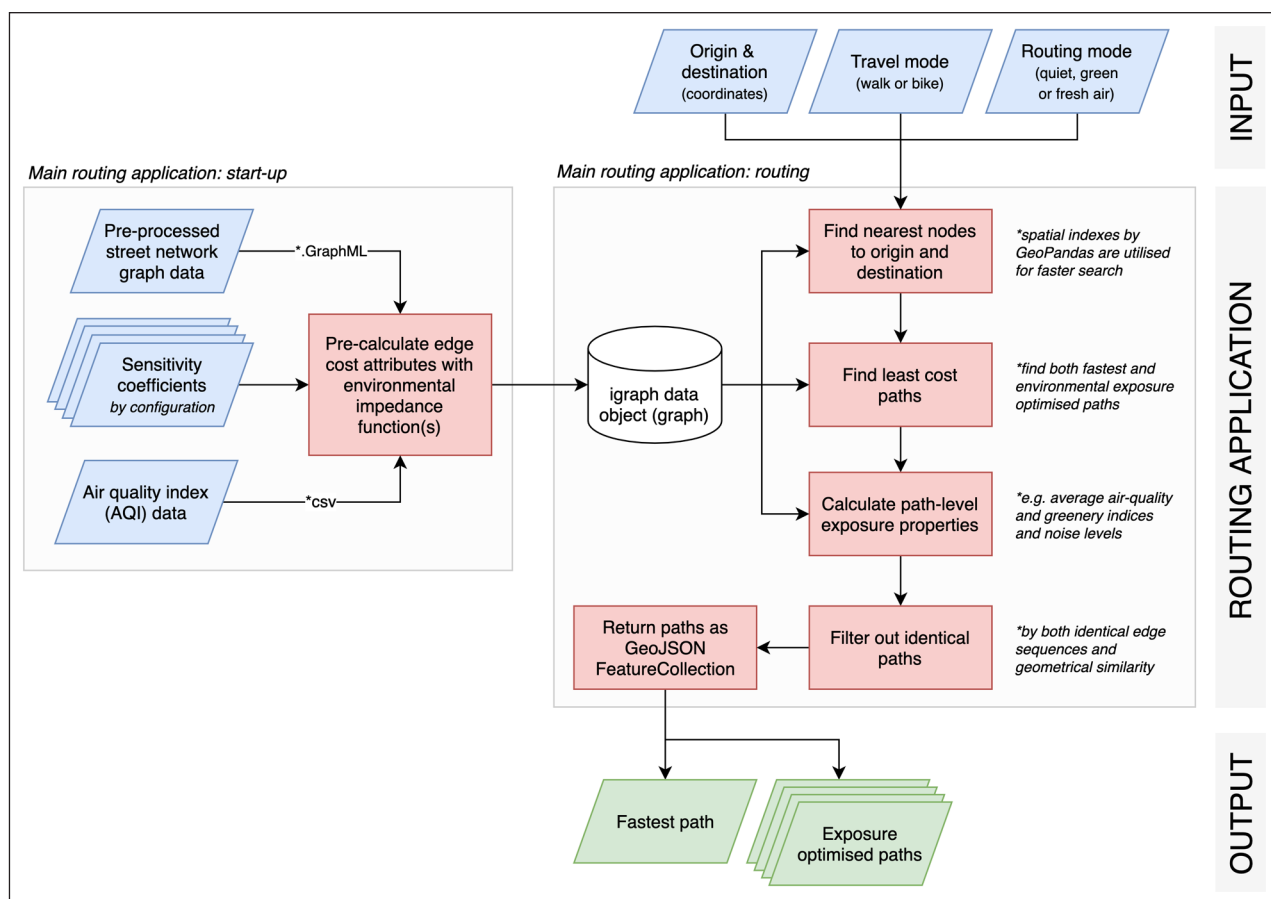


Figure 2 Flowchart of the start-up and the routing workflow of the main routing application (*gp\_server/gp\_server\_main.py*).

Prior to finding routes, the application finds the nearest edges and nodes to the user-defined origin and destination locations. If the nearest edge is closer than the nearest node, a new temporary node and linking edges are created on the nearest point of the nearest edge. Optimising the resource use is achieved by using a distance difference threshold: if the nearest node is no more than 30 m further than the nearest edge, that node is used as the origin or destination location as opposed to creating a new temporary node.

Since the routing logic is embedded in the pre-calculation of edge costs, routing is executed simply by solving one least cost path problem per each pre-calculated cost attribute as per the user-defined routing options (OD, travel mode, and exposure mode). Routing includes finding both fastest and several exposure optimised routes, as illustrated in [Figure 2](#).

After routing, the identified routes are initially only known as sequences of edge IDs. Complete route information, including the length and environmental exposure properties, are collected and calculated from the graph data by edge IDs. Averaged environmental exposure properties such as mean traffic noise, AQI, and GVI values are then calculated from edge-level properties. Also, accumulated distances within each traffic noise, AQI, and GVI range are calculated and added as route properties.

The maximum number of exposure optimised routes for one origin-destination pair is bounded by the number of sensitivity coefficients that are in use. Since employing the sensitivity coefficients aims to result in routes of little variance in travel time but significant improvement in environmental exposure, many of the identified routes are almost identical by geometry. Therefore, prior to returning the routes to the user, identical and nearly identical routes are filtered out using a geometrical overlay analysis (with 50 m buffers).

#### *API and response schema*

The Green Paths software publishes its routing functionality as a RESTful API. The API accepts HTTP GET requests that contain the necessary routing settings as (URL) path variables.

In the case of a routing error, the response status code indicates the type of the error, and the response data contains a descriptive error message. After successful routing, the API returns all route information in GeoJSON FeatureCollection format ([Figure 2](#)). Each route features several descriptive properties, including exposure profiles for traffic noise, air quality, and greenery, and general information such as total distance. The API and its usage are documented in more detail in the technical documentation.

## QUALITY CONTROL

The software code features several API tests and some unit tests that ensure the basic routing

functionality. The testing approach focuses mainly on higher level regression testing (API tests), which allowed heavy refactoring of the architecture as well as implementation details whenever necessary. We automated the test runs as continuous integration pipelines using GitHub Actions. This ensured that the software remained functional before changes were pushed to the main branch.

## (2) AVAILABILITY

### OPERATING SYSTEM

Linux, OS X, and Windows.

### PROGRAMMING LANGUAGE

Python 3.8.

### ADDITIONAL SYSTEM REQUIREMENTS

The CPU and memory requirements for the Green Paths software depend primarily on the size of the street network graph data, as the complete graph is loaded to memory at runtime. The size of the graph data file for HMA, including greenery and noise attributes, is roughly 500 MB. Hence, at least 500 + 100 MB memory is needed to run the application successfully. In practice, running several parallel instances of the application is recommended to allow a larger number and concurrent routing requests. Parallelisation can be achieved either with running the application in Docker containers, using multiple Gunicorn workers, or with a mixture of the two.

### DEPENDENCIES

The Green Paths software requires installation of the following Python libraries: *igraph*, *GeoPandas*, *Shapely*, *Flask*, *Gunicorn*, and the sub-dependencies of these. The complete Python environment is described as a YAML environment file and can be easily installed with the Conda package manager.

### LIST OF CONTRIBUTORS

This software was developed and designed by Joose Helle (UH & SYKE) with support from Age Poom (UH, project management, conceptualisation, testing, and research), Elias Willberg (UH, testing, and research), Tuomas Väisänen (UH, technical support), and Tuuli Toivonen (UH, conceptualisation, and supervision).

### SOFTWARE LOCATION

Archive

**Name:** Zenodo

**Persistent identifier:** DOI: [10.5281/zenodo.5409403](https://doi.org/10.5281/zenodo.5409403)

**Licence:** MIT licence

**Version published:** 0.8.0

**Date published:** 03/09/2021



Code repository

**Name:** GitHub  
**Persistent identifier:** <https://github.com/DigitalGeographyLab/green-path-server>  
**Licence:** MIT licence  
**Version published:** 0.8.0  
**Date published:** 03/09/2021 (v0.8.0)

LANGUAGE

The software and its technical documentation are written in English.

(3) REUSE POTENTIAL

The published version of the proof-of-concept software for exposure-optimised routing has been developed to work in the Helsinki Metropolitan Area based on the locally available environmental data. A complete installation guide is provided along with the source code, including data download steps for the Helsinki Metropolitan Area. The ready-made sample street network graph for the region includes traffic noise and street-level greenery information.

To reuse the Green Paths software in spatial contexts other than in the Helsinki Metropolitan Area, both OpenStreetMap-based street network data and environmental data are needed. Provided that OpenStreetMap data are available for many cities in a standard format and the application is written with modular Python, the software can be adopted and applied elsewhere, both for research and route planner applications.

A substantial proportion of the software functionality of handling environmental data requires localisation, depending on the available data and their characteristics, such as file formats, value ranges, and spatial extents. For adopting the software to other locations, the need for modification or developing new implementation solutions depends on the similarity of the available data in those locations with the data applied in the current version of the software, specific to the Helsinki Metropolitan Area (sample data and installation guide are provided in the technical documentation). Specifically, local implementation requires adaptation of the packages `graph_builder` and `aqi_updater` to match the available data formats and structures. The broad steps for data pre-processing and integration are described in the technical documentation (see the Availability section).

The software has been used in research to analyse multiple exposures during realised mobility and the accessibility of healthy travel environments in Helsinki [see 33]. The software is also applied in a prototype end-user route planner Green Paths, accessible at <https://green-paths.web.app/>. The end-user web application was developed in the European Union funded project Healthy Outdoor Premises for Everyone (UIA HOPE).<sup>4</sup> It visualises the results returned by the API by suggesting multiple routes between the user-defined origin-destination pair. It presents the results separately for walking and cycling, and for quiet, green, and fresh air routes (Figure 3). The development of Green Paths continues now under projects GREENTRAVEL and Urban Airquality 2.0.

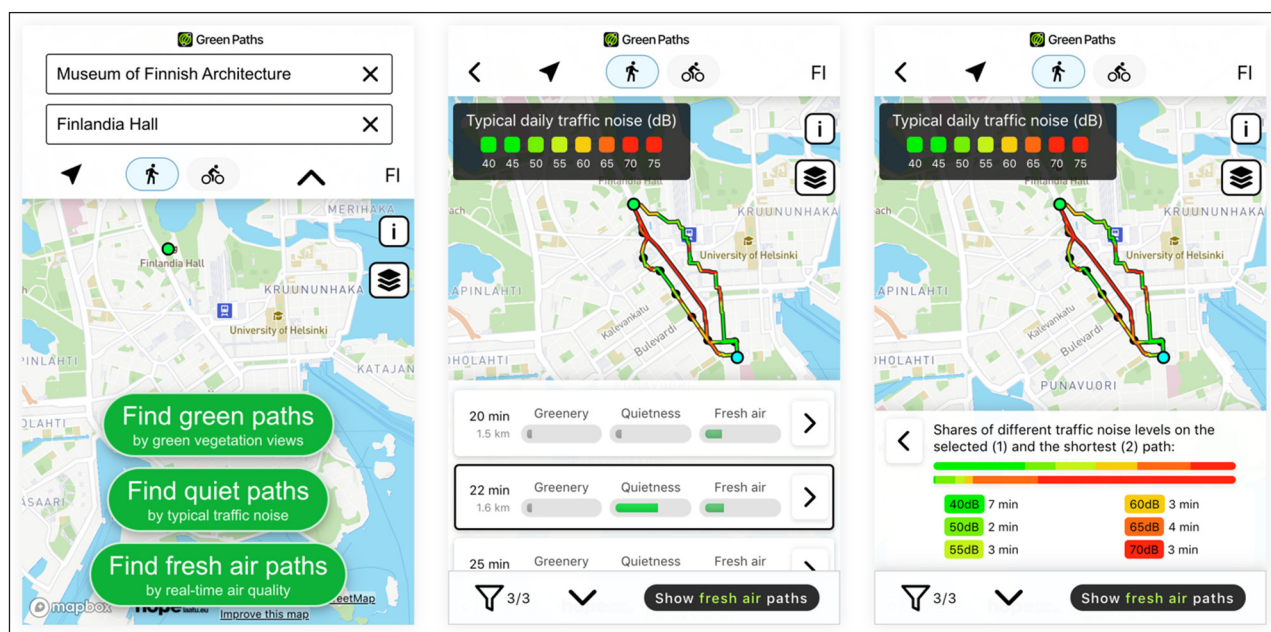


Figure 3 Selected screenshots of the Green Paths routing tool user interface for finding and comparing quiet routes for walking. Source: <https://green-paths.web.app>.

## NOTES

- 1 <https://www.cleanairroutes.london/>.
- 2 <https://www.ams-institute.org/about-ams/>.
- 3 <https://www.healthylplaces.eu/>.
- 4 <https://ilmanlaatu.eu/briefly-in-english/>.

## ACKNOWLEDGEMENTS

We would like to thank the Finnish Meteorological Institute, the Helsinki Region Environmental Services Authority, the City of Helsinki, Forum Virium Helsinki, and other partners from the UIA HOPE project, and the members of the Digital Geography Lab, University of Helsinki, for their input and support. We also wish to pass on our gratitude to all parties who made environmental data on street and path network, traffic noise, air quality, and street-level greenery in the Helsinki Metropolitan Area openly available and thus enabled the demonstration of exposure-optimised routing features of the Green Paths software: Akseli Toikka, Ville Mäkinen, Juha Oksanen, Lasse Johansson, the cities of Helsinki, Espoo, Vantaa, and Kauniainen, Finnish Meteorological Institute, Helsinki Region Environmental Services Authority, Finnish Environment Institute, and the OpenStreetMap project.

## FUNDING INFORMATION

This work was supported by the European Regional Development Fund via the Urban Innovation Action (Healthy Outdoor Premises for Everyone, UIA03-240), the Amer Foundation (Sporttia Stadiin/Urban Exerciser Project), the Estonian Research Council (grant No MOBTP1003), and the European Research Council Consolidator Grant (GREENTRAVEL, 101044906).

## COMPETING INTERESTS

The authors have no competing interests to declare.

## AUTHOR CONTRIBUTIONS

JH developed and designed the software with support from other authors. JH and AP wrote the manuscript with critical reading and revision support from EW and TT. AP contributed to project management, testing, and research. EW contributed to testing and research. TT contributed to funding acquisition and project management. All authors approved the submitted and the revised version.

## AUTHOR AFFILIATIONS

**Joose Helle**  [orcid.org/0000-0003-3774-9739](https://orcid.org/0000-0003-3774-9739)

Digital Geography Lab, Department of Geosciences and Geography, University of Helsinki, Gustaf Hällströmin katu 2, 00014 Helsinki, Finland

**Age Poom**  [orcid.org/0000-0003-3592-7837](https://orcid.org/0000-0003-3592-7837)

Digital Geography Lab, Department of Geosciences and Geography, University of Helsinki, Gustaf Hällströmin katu 2, 00014 Helsinki, Finland; Mobility Lab, Department of Geography, University of Tartu, Vanemuise 46, 50114 Tartu, Estonia

**Elias Willberg**  [orcid.org/0000-0003-0159-0084](https://orcid.org/0000-0003-0159-0084)

Digital Geography Lab, Department of Geosciences and Geography, University of Helsinki, Gustaf Hällströmin katu 2, 00014 Helsinki, Finland

**Tuuli Toivonen**  [orcid.org/0000-0002-6625-4922](https://orcid.org/0000-0002-6625-4922)

Digital Geography Lab, Department of Geosciences and Geography, University of Helsinki, Gustaf Hällströmin katu 2, 00014 Helsinki, Finland

## REFERENCES

1. **Poom A, Willberg E, Toivonen T.** Environmental exposure during travel: A research review and suggestions forward. *Health & Place*. 2021; 70: 102584. DOI: <https://doi.org/10.1016/j.healthplace.2021.102584>
2. **Doorley R, Pakrashi V, Szeto WY, Ghosh B.** Designing cycle networks to maximize health, environmental, and travel time impacts: An optimization-based approach. *International Journal of Sustainable Transportation*. 2020; 14(5): 361–374. DOI: <https://doi.org/10.1080/15568318.2018.1559899>
3. **Brand VS, Kumar P, Damascena AS, Pritchard JP, Geurs KT, de F. Andrade M.** Impact of route choice and period of the day on cyclists' exposure to black carbon in London, Rotterdam and São Paulo. *Journal of Transport Geography*. 2019; 76: 153–165. DOI: <https://doi.org/10.1016/j.jtrangeo.2019.03.007>
4. **Alam MS, Perugu H, McNabola, A.** A comparison of route-choice navigation across air pollution exposure, CO<sub>2</sub> emission and traditional travel cost factors. *Transportation Research Part D: Transport and Environment*. 2018; 65: 82–100. DOI: <https://doi.org/10.1016/j.trd.2018.08.007>
5. **Mölter A, Lindley S.** Influence of walking route choice on primary school children's exposure to air pollution – A proof of concept study using simulation. *Science of the Total Environment*. 2015; 530–531: 257–262. DOI: <https://doi.org/10.1016/j.scitotenv.2015.05.118>
6. **Novack T, Wang Z, Zipf A.** A system for generating customized pleasant pedestrian routes based on openstreetmap data. *Sensors*. 2018; 18(11). DOI: <https://doi.org/10.3390/s18113794>
7. **Nieuwenhuijsen MJ.** Urban and transport planning pathways to carbon neutral, liveable and healthy cities; A

- review of the current evidence. *Environment International*. 2020; 140: 105661. DOI: <https://doi.org/10.1016/j.envint.2020.105661>
8. **Ribeiro P, Mendes JFG.** Route planning for soft modes of transport: Healthy routes. In: *Urban Transport XVII. WIT Transactions on the Built Environment*; 2011. WIT Press. DOI: <https://doi.org/10.2495/UT110571>
  9. **Rapoport A, Gisches EJ, Daniel T, Lindsey R.** Pre-trip information and route-choice decisions with stochastic travel conditions: Experiment. *Transportation Research Part B: Methodological*. 2014; 68: 154–172. DOI: <https://doi.org/10.1016/j.trb.2014.06.007>
  10. **Jiang G, Fosgerau M, Lo HK.** Route choice, travel time variability, and rational inattention. *Transportation Research Part B: Methodological*. 2020; 132: 188–207. DOI: <https://doi.org/10.1016/j.trb.2019.05.020>
  11. **Davies G, Whyatt JD.** A network-based approach for estimating pedestrian journey-time exposure to air pollution. *Science of the Total Environment*. 2014; 485: 62–70. DOI: <https://doi.org/10.1016/j.scitotenv.2014.03.038>
  12. **Zou B, Li S, Zheng Z, Zhan BF, Yang Z, Wan N.** Healthier routes planning: A new method and online implementation for minimizing air pollution exposure risk. *Computers, Environment and Urban Systems*. 2020; 80: 101456. DOI: <https://doi.org/10.1016/j.compenvurbsys.2019.101456>
  13. **Van den Hove A, Verwaeren J, Van den Bossche J, Theunis J, De Baets B.** Development of a land use regression model for black carbon using mobile monitoring data and its application to pollution-avoiding routing. *Environmental Research*. 2020; 183: 108619. DOI: <https://doi.org/10.1016/j.envres.2019.108619>
  14. **Gao L-N, Tao F, Ma P-L, Wang C-Y Kong W, Chen W-K Zhou T.** A short-distance healthy route planning approach. *Journal of Transport & Health*. 2022; 24: 101314. DOI: <https://doi.org/10.1016/j.jth.2021.101314>
  15. **Ludwig C, Lautenbach S, Schömann E-M, Zipf A.** Comparison of Simulated Fast and Green Routes for Cyclists and Pedestrians. In: *11th International Conference on Geographic Information Science*; September 27–30, 2021. URL <https://drops.dagstuhl.de/opus/volltexte/2021/14758/pdf/lipics-vol208-giscience2021-complete.pdf>.
  16. **Caggiani L, Camporeale R, Ottomanelli M.** A real time multi-objective cyclists route choice model for a bike-sharing mobile application. In: *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2017 – Proceedings. Institute of Electrical and Electronics Engineers Inc*; 2017. DOI: <https://doi.org/10.1109/MTITS.2017.8005593>
  17. **Ribeiro PJG, Dias GJC, Mendes JFG.** Health-oriented routes for active mobility. *Journal of Transport & Health*. 2022; 26: 101410. DOI: <https://doi.org/10.1016/j.jth.2022.101410>
  18. **García-Díaz JA, Noguera-Arnaldos JÁ, Hernández-Alcaraz ML, Robles-Marín IM, García-Sánchez F, Valencia-García R.** AllergyLESS. An intelligent recommender system to reduce exposition time to allergens in smart-cities. In: *Distributed Computing and Artificial Intelligence, 15th International Conference*; 2019. Springer Verlag. DOI: [https://doi.org/10.1007/978-3-319-94649-8\\_8](https://doi.org/10.1007/978-3-319-94649-8_8)
  19. **Rußig, J, Bruns J.** Reducing Individual Heat Stress through Path Planning. *GI\_Forum*. 2017; 1: 327–340. DOI: [https://doi.org/10.1553/giscience2017\\_01\\_s327](https://doi.org/10.1553/giscience2017_01_s327)
  20. **Opach T, Navarra C, Rød JK, Neset T-S.** Towards a Route Planner Supporting Pedestrian Navigation in Hazard Exposed Urban Areas. In: *ISCRAM 2020 Conference Proceedings – 17th International Conference on Information Systems for Crisis Response and Management*; 2020. Virginia Tech. URL <http://citizensensing.eu/>. [Online; accessed 5-October-2020]
  21. **Mahajan S, Tang Y-S, Wu D-Y, Tsai T-C, Chen L-J.** CAR: The Clean Air Routing Algorithm for Path Navigation with Minimal PM2.5 Exposure on the Move. *IEEE Access*. 2019; 7: 147373–147382. DOI: <https://doi.org/10.1109/ACCESS.2019.2946419>
  22. **Grieve A.** Clean Air Walking Routes. Monitoring Report for Cross River Partnership; 2017. URL <https://crossriverpartnership.org/wp-content/uploads/2019/04/clean-air-walking-routes-air-quality-monitoring-report-1-1.pdf>.
  23. **Hasenfratz D, Arn T, De Concini I, Saukh, O, Thiele L.** Health-optimal routing in urban areas. In: *IPSN 2015 – Proceedings of the 14th International Symposium on Information Processing in Sensor Networks. Association for Computing Machinery, Inc*; 2015. DOI: <https://doi.org/10.1145/2737095.2737135>
  24. **Brauer M, Nunes M, Winters M, Su J.** An Interactive Route Planner Incorporating Air Pollution and Cycling Determinants to Facilitate and Promote Cycling in Metro Vancouver, Canada. *Epidemiology*. 2011; 22: S78. DOI: <https://doi.org/10.1097/01.ede.0000391902.48815.e4>
  25. **Su JG, Winters M, Nunes M, Brauer M.** Designing a route planner to facilitate and promote cycling in Metro Vancouver, Canada. *Transportation Research Part A: Policy and Practice*. 2010; 44(7): 495–505. DOI: <https://doi.org/10.1016/j.tra.2010.03.015>
  26. **Hatzopoulou M, Weichenthal S, Barreau G, Goldberg M, Farrell W, Crouse D, Ross N.** A web-based route planning tool to reduce cyclists' exposures to traffic pollution: A case study in Montreal, Canada. *Environmental Research*. 2013; 123: 58–61. DOI: <https://doi.org/10.1016/j.envres.2013.03.004>
  27. **Nurminen A, Malhi A, Johansson L, Framling K.** A Clean Air Journey Planner for pedestrians using high resolution near real time air quality data. In: *IEEE Xplore*; 2020. Institute of Electrical and Electronics Engineers (IEEE). DOI: <https://doi.org/10.1109/IE49459.2020.9155068>

28. **Karppinen A, Johansson L.** Fusion of Air Quality Information: Evaluation of the Enfuser-Methodology in Finland and a Case Study in China. In: *Air Pollution Modeling and its Application XXV*; 2018. ITM 2016. Springer Proceedings in Complexity. Springer. DOI: [https://doi.org/10.1007/978-3-319-57645-9\\_34](https://doi.org/10.1007/978-3-319-57645-9_34)
29. **Johansson L, Karppinen A, Kurppa M, Kousa A, Niemi JV, Kukkonen J.** An operational urban air quality model ENFUSER, based on dispersion modelling and data assimilation. *Environmental Modelling & Software*. 2022; 156: 105460. DOI: <https://doi.org/10.1016/j.envsoft.2022.105460>
30. **Sito OY.** *Helsingin kaupungin meluselvitys 2017*. Helsingin kaupunki; 2017. URL <https://www.hel.fi/static/liitteet/kaupunkiymparisto/liikenne-ja-kartat/ymparistovaikutukset/julkaisu-04-17.pdf>.
31. **Johansson L, Poom A, Niemi J, Timonen H, Karppinen A.** On the Utilization of Real-Time Activity and Air Quality Sensor Data in a Local-Scale Operative Dispersion Model in Helsinki. In: *Air Pollution Modeling and its Application XXVIII*; 2022. Springer International Publishing. DOI: [https://doi.org/10.1007/978-3-031-12786-1\\_6](https://doi.org/10.1007/978-3-031-12786-1_6)
32. **Toikka A, Willberg E, Mäkinen V, Toivonen T, Oksanen J.** The green view dataset for the capital of Finland, Helsinki. *Data in Brief*. 2020; 30: 105601. DOI: <https://doi.org/10.1016/j.dib.2020.105601>
33. **Willberg E, Poom A, Helle J, Toivonen T.** Cyclists' exposure to air pollution, noise, and greenery: a population-level spatial analysis approach. *International Journal of Health Geographics*. 2023; 22(1): 1–21. DOI: <https://doi.org/10.1186/s12942-023-00326-7>

---

#### TO CITE THIS ARTICLE:

Helle J, Poom A, Willberg E, Toivonen T 2023 The Green Paths Route Planning Software for Exposure-Optimised Active Travel. *Journal of Open Research Software*, 11: 10. DOI: <https://doi.org/10.5334/jors.400>

**Submitted:** 08 October 2021    **Accepted:** 29 September 2023    **Published:** 13 October 2023

#### COPYRIGHT:

© 2023 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

*Journal of Open Research Software* is a peer-reviewed open access journal published by Ubiquity Press.