The Journal of Open Source Software

GTFS Segments: A Fast and Efficient Library to Generate Bus Stop Spacings

Saipraneeth Devunuri [1*¶] **and Lewis Lehe** [1*]

1 Department of Civil and Environmental Engineering, University of Illinois Urbana-Champaign ¶ Corresponding author * These authors contributed equally.

DOI: 10.21105/joss.06306

Software

- Review
- Repository
- Archive

Editor: Olivia Guest

Reviewers:

- @nagellette
- @xoolive

Submitted: 28 September 2023
Published: 19 March 2024

Summary

The GTFS Segments (gtfs-segments) library is an open-source Python toolkit for computing, visualizing and analyzing bus stop spacings: the distance a transit bus travels between stops. The library reads General Transit Feed Specification (GTFS) data, snaps bus stops to points along routes, divides routes into segments (the pieces of routes between two stops), and then produces a `GeoDataFrame` containing information about each segment. The library also features several functions that act on this GeoDataFrame. It can produce summary statistics of the spacings for a given network, using various weighting schemes (i.e., weighting by frequency of service), as well as histograms of spacings that display their full distribution. In addition to network-level statistics, the package can also compute statistics for each route, such as its length, headway, speed, and average stop spacing. It can draw maps of networks, routes, or segments over a basemap—which allows the user to manually validate data. The segments DataFrame can be exported to `.csv` and `.geojson` file formats. The package can fetch the most up-to-date GTFS data from Mobility Data (MobilityData, 2023) repositories for user convenience.

Statement of need

The choice of bus stop spacing involves a tradeoff between accessibility and speed: wider spacings mean passengers must travel farther to/from stops, but they allow the bus to move faster (Wu et al., 2022). Many US transit agencies have recently carried out *stop consolidation* campaigns that systematically remove stops, due partly to the perception US stop spacings are much narrower than those abroad. However, there are no reliable data sources to obtain current stop spacings despite the wide adoption of General Transit Feed Specification (GTFS) (Voulgaris & Begwani, 2023), because GTFS does not include data on stop spacings directly. Spacings must be computed from route shape geometries, stop locations, and stop sequences. A challenge is that stop locations are not placed on top of route shapes and therefore must be somehow projected onto the route's `LINESTRING`. To make spacings available for analysis, `gtfs-segments` use k-dimensional spatial trees and k-nearest neighbor heuristics to snap stops to routes and divide routes into segments for computation of spacings, as described below.

`gtfs-segments` was designed for researchers, transit planners, students and anyone interested in bus networks. The package has been used in several scholarly articles (Devunuri et al., 2023, 2024; Lehe & Pandey, 2022) and to create databases of spacings for over 550 agencies in the US (Devunuri et al., 2022) and 80 agencies in Canada (Devunuri, 2023). Several transit agencies, such as Regional Transportation District Denver (RTD-Denver), have used the package to visualize the effects of their bus stop consolidation efforts. Filtering functions allow the user to explore datasets, identify errors and compute specialized statistics.

Functionality

`gtfs-segments` has four main functionalities: (1) Downloading GTFS feeds (2) Computing segments (3) Visualizing stop spacings (4) Calculating stop spacing summary statistics. Each is further detailed below.

Downloading GTFS feeds

The package permits the user to search and download recent GTFS feeds from the Mobility Database Catalogs (MobilityData, 2023). It allows for keyword and fuzzy search of GTFS feeds using location (e.g., `Minneapolis`, `San Francisco`) or provider name (e.g., `WMATA`, `Capital Metro`) as input.

Computing segments

The fundamental unit of analysis used by `gtfs-segments` is the *segment*, which is a piece of a bus network defined by three properties: (i) a start stop, (ii) an end stop and (iii) the path that the bus travels along the route in between the two consecutive stops. A segment's *spacing* is the distance of (iii). `gtfs-segments` produces segments by efficiently and robustly snapping stop locations onto route shapes. Figure 1 shows examples where a stop is equidistant from multiple route coordinates. Here, projecting the stop onto the route or snapping to the nearest geo-coordinate (lat, lon) may yield stops that are out-of-order or snapped far from their locations. Also, the time complexity of projection or snapping using brute force is $O(nm)$ for `n` stops and `m` geo-coordinates that represent the route shape.

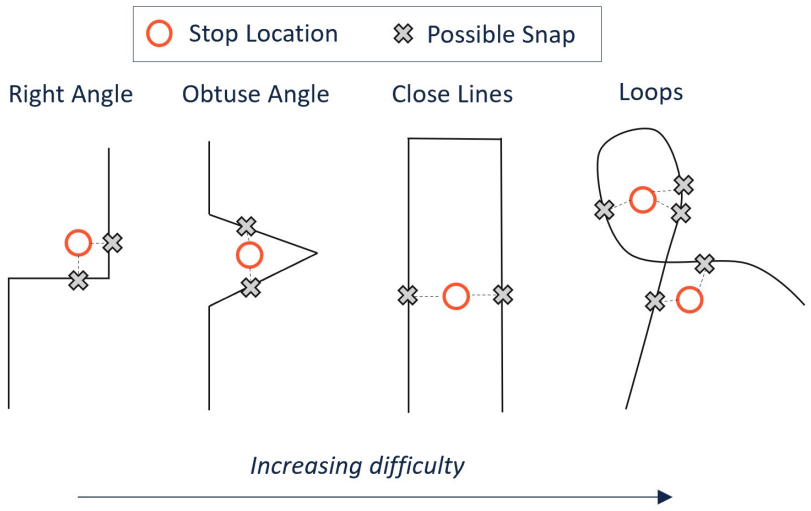


Figure 1: Example route shapes with stop locations that are equidistant from multiple points along the route.

`gtfs-segments` overcomes these challenges by increasing the route resolution (i.e., adding points in between geo-coordinates), using spatial k-d trees, and using more than one nearest neighbor. The increase in resolution allows stops to be snapped to nearby points. Using k-d trees reduces the time complexity to $O(nlog(m))$ and makes it possible to compare among several snapping points without added computation. Figure 2 shows an example where initially snapping to the nearest point produces out-of-order stops (3/4/2) and stop 5 is snapped far away from its location. Increasing the resolution (second panel) fixes 5's location problem but the ordering problem persists. By using k=3 nearest neighbors, we find a proper ordering (last

panel). Once every stop has been snapped to a geo-coordinate on the route shape, the shape is segmented between stops and each segment's geometry is stored in a GeoDataFrame.

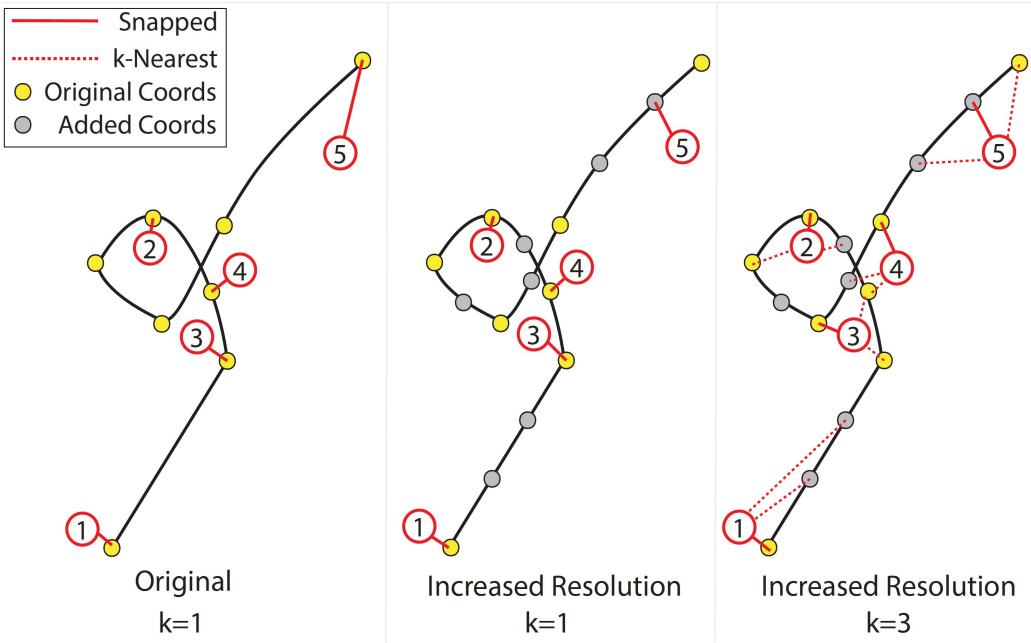


Figure 2: Improvement in snapping due to an increase in resolution and suing k-nearest neighbors.. Adapted from "Bus Stop Spacings Statistics: Theory and Evidence" (Devunuri et al., 2024)

Packages such as `gtfs2gps` (Pereira et al., 2023) and `gtfs_functions` (Toso & Oja, 2023) also compute segments. In addition to its snapping algorithm, visualization, download, and statistical functionalities, `gtfs-segments` is distinguished from those in two ways. First, it has a faster processing rate[1] to compute segments both with and without parallel processing (see Table 1). Second, `gtfs-segments` is tolerant to deviations from GTFS standards. For example, because the Chicago Transit Authority does not have an agency_id in its routes.txt, `gtfs2gps` fails to read it even though this field is not needed for obtaining segments.

			Average Processing Rate (Trips/Second) [n=3]			
		Package	*gtfs2gps*	*gtfs_functions*	*gtfs_segments*	
		Parallel Processing	*Y*	*N*	*N*	*Y*
Agency	**City**	**File Size (MB)**	v2.1.0	v2.3.0	v2.1.0	
SFMTA	San Francisco	10.0	41	621	625	**716**
MBTA	Boston	14.9	87	325	388	**431**
TriMet	Portland	35.5	64	85	103	**106**

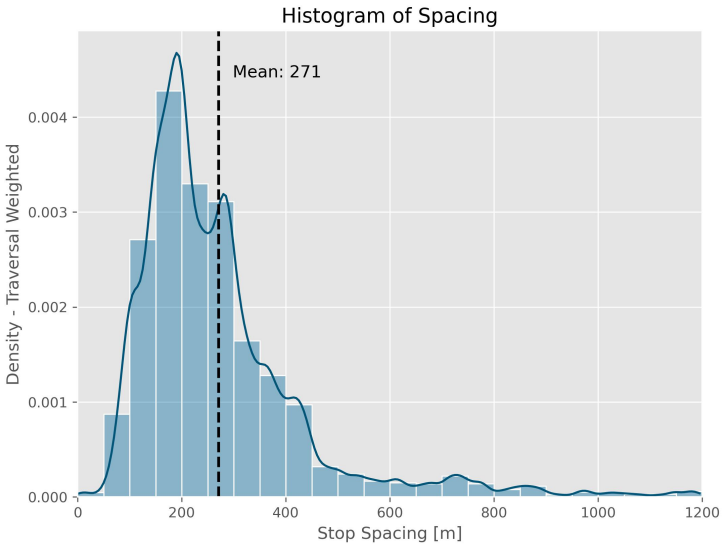
Table 1: Comparison of average processing rates for gtfs2gps, gtfs_functions and gtfs_segments.

[1]The average processing rate is the average number of trips processed per second, averaged over 3 independent runs. The experiments were run with an `Intel(R) Core(TM) i9-10920X` processor at 3.50GHz with 12 hyperthreaded CPU cores and 64 GB RAM, running on Windows. The most recent GTFS feeds (as of February 2024) for the respective agencies were used.

Visualizing stop spacings

The package can create maps of stops and segments (with basemap), including interactive maps. See Figure 3a, which colors segments by spacing. The package can also produce histograms of stop spacings (see Figure 3b), which can inform strategic decisions about network design.

(a) Interactive Heatmap



(b) Histogram of stop spacings

Figure 3: Other visualization features in the package. SFMTA GTFS feed was used to generate these.

Calculating stop spacing summary statistics

Discussions about stop spacings, commonly include statistical metrics such as means and medians, used to spacings between different agencies or track changes within an agency over time. `gtfs-segments` can produce weighted mean, median, and standard deviations for an agency, using different weighting systems (e.g., weighting segments by the number of times a bus traverses it or the number of routes that include it) as outlined by Devunuri et al. (2024). For each route, `gtfs-segments` can give metrics such as mean spacing, headways, speeds, number of buses in operation and route lengths.

Acknowledgments

The `gtfs-segments` package draws its inspiration from `gtfs_functions` (Toso & Oja, 2023), `gtfs2gps` (Pereira et al., 2023), and `partridge` (Whalen, 2023) repositories. We thank the contributors of these packages for their excellent work. We also extend our thanks to Mobility Data (MobilityData, 2023) for compiling GTFS from around the globe and constantly maintaining them.

References

Devunuri, S. (2023). *Bus Stop Spacings for Transit Providers in Canada* (Version V2) [Data set]. Harvard Dataverse. https://doi.org/10.7910/DVN/QFTAPM

Devunuri, S., Lehe, L. J., Qiam, S., Pandey, A., & Monzer, D. (2024). Bus stop spacing statistics: Theory and evidence. *Journal of Public Transportation*, *26*, 100083. https://doi.org/10.1016/j.jpubtr.2024.100083

Devunuri, S., Qiam, S., & Lehe, L. (2022). *Bus Stop Spacings for Transit Providers in the US* (Version V3) [Data set]. Harvard Dataverse. https://doi.org/10.7910/DVN/SFBIVU

Devunuri, S., Qiam, S., & Lehe, L. (2023). *ChatGPT for GTFS: From Words to Information*. https://doi.org/10.48550/arXiv.2308.02618

Lehe, L., & Pandey, A. (2022). Bus stop spacing with heterogeneous trip lengths and elastic demand. *Available at SSRN 4135394*. https://doi.org/10.2139/ssrn.4135394

MobilityData. (2023). *Mobility Database*. https://database.mobilitydata.org/

Pereira, R. H., Andrade, P. R., & Vieira, J. P. B. (2023). Exploring the time geography of public transport networks with the gtfs2gps package. *Journal of Geographical Systems*, *25*(3), 453–466. https://doi.org/10.31235/osf.io/qydr6

Toso, S., & Oja, R. (2023). *Gtfs_functions: Package with useful functions to create geo-spatial visualizations from a GTFS.* GitHub. https://github.com/Bondify/gtfs_functions

Voulgaris, C. T., & Begwani, C. (2023). Predictors of early adoption of the general transit feed specification. *Findings*. https://doi.org/10.32866/001c.57722

Whalen, D. (2023). *Partridge: A fast, forgiving GTFS reader built on pandas DataFrames*. GitHub. https://github.com/remix/partridge

Wu, T., Jin, H., & Yang, X. (2022). To what extent may transit stop spacing be increased before driving away riders? Referring to evidence of the 2017 NHTS in the united states. *Sustainability*, *14*(10). https://doi.org/10.3390/su14106148