TransitMapper Documentation

Patrick Brosi

16th March 2017

1 Basic Usage

The input file is a graph given as a (simplified) Geo-JSON file, consisting of nodes (represented as "Point"-features) and edges (represented as "LineString"-features). Each edge has a collection of unique "lines" that travel through it. The transitmapper will render these lines in a way that resembles a transit map. The input is read from `stdin`.

```
$ transitmapper -o test.svg < test.json
```
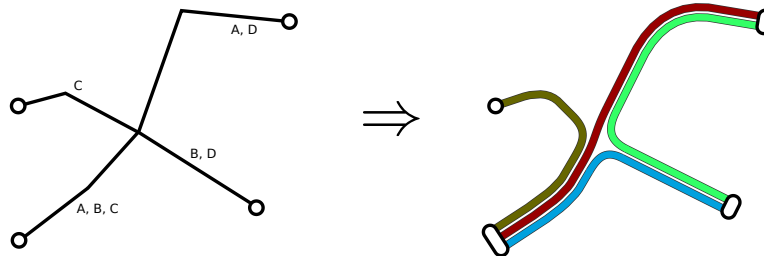
See below for an example input.

Figure 1: Simple example output

2 Command line parameters

The following command line parameters are accepted by `transitmapper` (see also `--help`).

`--line-width=N` The default width of a line, in output units. 20 by default.

`--line-spacing=N` The default spacing between lines, in output units. 10 by default.

`--render-station-names` Output the station names (experimental).

`--render-node-fronts` Output node fronts, useful for debugging.

`--resolution=D` Output resolution. 0.1 by default.

--no-optim (-N) Disable line-ordering optimization.

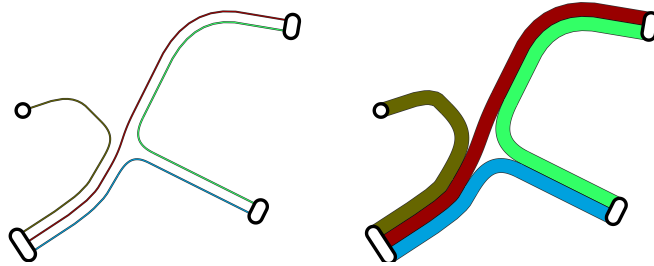--input-smoothing=D Level of input-data smoothing. 3 by default.

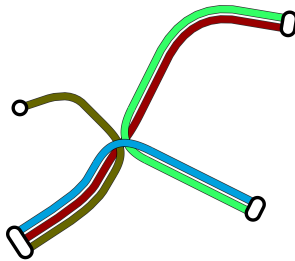Figure 2: Different settings of --line-width and --line-spacing
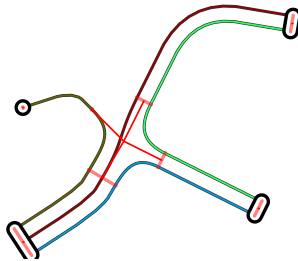

Figure 3: Output without ordering optimization (-N)

Figure 4: Node-front rendering (--render-node-fronts)

3   JSON Format

4   Example Input

```
1  {
2    "type": "FeatureCollection",
3    "features": [
4      {
5        "geometry": {
6          "coordinates": [0, 0],
7          "type": "Point"
```

```
 8          },
 9          "properties": {
10            "id": "1",
11            "station_id": "1"
12          }
13        },
14        {
15          "geometry": {
16            "coordinates": [1000, 1000],
17            "type": "Point"
18          },
19          "properties": {
20            "id": "2",
21            "station_id": "2"
22          },
23          "type": "Feature"
24        },
25        {
26          "geometry": {
27            "coordinates": [
28              [0, 0], [500, 900], [1000, 950]
29            ],
30            "type": "LineString"
31          },
32          "properties": {
33            "from": "1",
34            "to": "2",
35            "lines": [
36              {"color": "00a1de"},
37              {"color": "990000"}
38            ]
39          }
40        }
41    ]
42  }
```