Worldwide Railway Skeleton Network: Extraction Methodology and Preliminary Analysis

Sebastian Wandelt, Zezhou Wang, and Xiaoqian Sun

***Abstract*— Understanding and improving global mobility has gained increased interest during the last decades. However, studies on the railway network are spatially limited so far, mostly investigating the domestic network of a country. Data availability is a major limiting factor for the analysis of these networks. Despite the increased open data movement, network operators are often reluctant to publish their infrastructure and passenger data. Existing large-scale studies usually make use of hand-collected data, for instance, based on historical cartographies. In this paper, we develop and implement a methodology to extract the worldwide railway skeleton network from the open data repository OpenStreetMap, where nodes are stations/waypoints and links are weights with information such as spatial distance, gauge, and maximum speed. We describe how we solved several data cleansing and scalability issues and developed network simplification techniques, in order to obtain an adequate representation of the network. We show that the network breaks down into few large and many small components. Furthermore, we show that this public data set can be used for efficient minimum travel time estimation between stations or cities. This paper leads to the development of a new research data set and contributes toward the ability of analyzing global mobility patterns, particularly regarding multimodality and cross-country transportation.**

***Index Terms*— Worldwide railway network, OpenStreetMap, global mobility.**

I. Introduction

EFFICIENT and resilient transportation is fundamental for economic development and it is one of the major challenges of the 21^{st} century. Research on transportation networks has gained increased interests during the last decades. Examples for analysis include, origin-destination demand estimation [1], [2], multi-layer transportation [3], [4], resilience analysis [5], [6], competition and cooperation studies between network operators [7], [8], communication processes [9], [10], and temporal evolution [11]–[13]. For these analysis, transportation networks are modeled as complex networks with nodes (airports, railway stations, route points, etc.) connected by links (flight connections, physical infrastructure links, etc.), at different levels of granularity, seasonality, and type [14]. Most transportation networks, and man-made networks in general, exhibit highly similar statistical characteristics, while displaying substantial non-trivial topological features: The patterns of connection between the elements of the networks are neither purely regular nor purely random [15]. Therefore, it is interesting and important to analyze the properties of these networks.

Studies on the railway network are spatially limited so far, mostly investigating the domestic network of a country, for instance, in China [13], [16], [17], India [18], [19], history of European railway [20], or rather local urban networks [21]. Other researchers have looked at high-speed railway specific research problems, such as, demand estimation [22], and competition/cooperation with air transport [23]. Moreover, one difficulty when dealing with small subsystems is that small-scale data provides poor statistics [24].

Data availability is a major limiting factor for analysis of railway networks. Despite of the increased Open Data-movement, network operators are often reluctant to publish their infrastructure and passenger data. Existing studies at large scales usually make use of hand-collected data, for instance, based on historical cartographies [20]. However, this data is not being made available for public use.

In this paper, we propose a method to extract the Worldwide Railway Network (WRN), where a node is a waypoint/station and a link between two nodes describes whether two nodes are physically connected in the infrastructure network. As a data source, we use OpenStreetMap (OSM), a community project that aims to create a free editable map of the world, started in 2004. Within the last 12 years, OSM data has become an accurate representation of the world, particularly for urban areas [25]–[27]; spurred by the development of cheap, yet accurate GPS devices and the increased usage of smartphones. The major contributions of our study are as follows:

1) We extract the worldwide railway network and its components from an otherwise highly inaccessible (more than 30 GB of compressed data) and rather inconsistent (requiring non-trivial data cleansing) dataset. The extraction is described in detail, addressing several data management problems, for instance, data cleansing and scalability of algorithms. The result of our transformation is a consistent representation of the worldwide railway network, requiring a few dozen MB of storage only and being able to be processed on commodity hardware. Moreover, we publish the source code of our

Manuscript received May 26, 2016; revised September 2, 2016 and November 22, 2016; accepted November 23, 2016. Date of publication December 16, 2016; date of current version July 31, 2017. This work was supported in part by the Research Fund for International Young Scientists, in part by the National Natural Science Foundation of China under Grant 61650110516, in part by the Research Fund for Young Scientists, and in part by the National Natural Science Foundation of China under Grant 61601013. The Associate Editor for this paper was P. Ioannou. *(Corresponding author: Xiaoqian Sun.)*

S. Wandelt and X. Sun are with the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China, and also with the Beijing Key Laboratory for Network-Based Cooperative ATM, Beijing 100191, China (e-mail: wandelt@informatik.hu-berlin.de; sunxq@buaa.edu.cn).

Z. Wang is with the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China (e-mail: wangzezai@buaa.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2016.2632998

extraction technique.[1] Not only does this ensure reproducibility of our study, but also increase the scientific impact [28], [29].

2) We develop a set of network simplification techniques, which aid us to significantly reduce the number of nodes and links in the network, while keeping an accurate representation of inter-station connections. Such a step is necessary, if scalable post-processing is required.
3) We identify and analyze the largest components in the worldwide railway network and find that the network extracted from OSM is a planar network, which breaks down into few very large, and many smaller components. Furthermore, we show that this data is useful to analyze high-speed railways as well as for efficient travel time prediction between stations/cities.
4) Our work is a significant contribution towards the ability of analyzing global mobility patterns, in the face of multi-modal transportation. Particularly, future work can use our methods and algorithms to investigate large-scale cooperation/competition between air transport and railway systems, the analysis of global multi-layer transportation resilience, and more accurate mode-dependent origin-destination demand estimation.

It should be noted, that the major contribution of this paper is not an analysis of the network itself, but rather the presentation and discussion of a technique to extract the worldwide railway network for future use in global transportation research. Clearly, the automated extraction of such a huge network poses tremendous challenges, not all of which can be addressed and solved in a single paper. Therefore, we hope that our work is an initial stimulation towards public usable railway data. We envision that our work contributes to the improvement of railway operations, such as positive train control [30]. Among others, the following use cases of our extraction method, mostly related to the automatic travel time estimation over long distances:

1) Assessment of accessibility of regions to public infrastructure. Examples include the accessibility to jobs, airports, or touristic hot spots.
2) The addition of new railway lines and the increase of maximum speed is always a trade off between cost and benefit. The benefits of new railway infrastructure can be conveniently evaluated with the abstraction of a worldwide network; such experiments cannot be performed with existing APIs to websites such as Google Maps or Baidu Maps.
3) Similarly, estimating the resilience of transportation networks by simulating the removal of infrastructure links and their impacts on overall travel times of passengers is an interesting direction for future work. Again, these experiments need to work on network representations.

The remainder of this paper is structured as follows. In Section II, we describe the data sets and methods for the extraction of the worldwide railway network. The results of our study are presented in Section III, where we perform sensitivity analysis of simplification parameters on smaller regions first, followed by the discussion of the extracted worldwide railway network. We conclude with a summary of our findings and discuss possible direction for future work in Section IV.

[1]We make our source code available for free academic use at Github. The link is: https://github.com/hubsw/OSMRailway.

II. Data and Methods

This section describes the data and methodology used in our study. The goal is to extract a consistent model of the worldwide railway network from the planet file of OpenStreetMap (OSM, http://planet.osm.org/). OSM is community project that aims to create a free editable map of the world, started in 2004. Technically, it belongs to the area of crowd-sourcing, referring to how large groups of users can perform functions that are either difficult to automate or expensive to implement; OSM is one of the leading examples of such an effort [31]. While, in general, there is no reliable estimation if a certain object or other detailed attribute information is included in OSM, the amount of coverage in urban areas is very high [32]. As part of the OSM effort is transportation modes throughout the world are being modeled, including stations and physical layout of lines. In our study, we use the data from OSM to extract a global network between railway stations. In our study, we model the WRN as a network $\langle N, L, S, f_N, f_L \rangle$, where N is a set of nodes, L is a set of links between nodes, $S \subseteq N$ is a set of stations, f_N assigns key-value pairs to nodes in the network and f_L assigns key-value pairs to links in the network. We break down our methods for modeling the WRN into four steps as follows:

1) *Data Extraction, Transformation, and Loading (See Section II-A):* During this step, we load data from the planet OSM file into intermediate data structure and fix general inconsistencies, introduced during the mapping process. Particularly, this process introduces some data cleansing tasks, specific to the problem of railway modelling in OSM.
2) *Connecting the Network (See Section II-B):* The WRN, as modeled in OSM, is highly disconnected into smaller components, many of which are unintentionally created. We address this problem by carefully devising an efficient and effective component merging strategy, which connects close nodes in the network, whenever they are appropriate to be connected.
3) *Connecting Stations to the WRN (See Section II-C):* Mappers often model stations as separate nodes or complex areas, which leads to the problem that stations are often disconnected from the network. In order to solve this problem, we identify singular nodes for stations and connect them with their closest infrastructure way segments.
4) *Network Simplification (See Section II-D):* Since we are interested in an inter-station network, several infrastructure elements, such as, spurs and sidings, are not relevant to the skeleton of station network. Moreover, the level of detail in urban areas is very high; higher than necessary in our study. We address the problem by significantly simplifying the network, substantially reducing the number of nodes and links, while still preserving all information relevant for inter-station connections.

A. Data Extraction, Transformation, and Loading

We follow the general process of Extract-Transform-Load, as it is used in many data warehouse scenarios [33]. Data extraction refers to the process of getting data out of the original data source; transformation addresses storing data in a proper data structure for the purposes of analysis, while fixing obvious inconsistencies; and loading is the finalizing process of data creation.

1) Extraction: The original release of the planet file, covering global data, can be found at http://planet.osm.org/. Other mirrors exist which provide local snapshots at city/country level, e.g. at http://download.geofabrik.de/. During the last decade, a set of different file formats have been proposed by the community, each with its own limitations. Originally, the planet file is encoded as a semi-structured XML file. Since this file is rather large (approx. 600 GB), it is often compressed using bzip2, which leads to a file size of around 49 GB. Nevertheless, in order to parse the data, accessing such large semi-structured files is rather slow and cumbersome. Therefore, during the last years, a different file format has been developed: PBF [34]. PBF is a binary format, based on Google protocol buffer, which is 1) a natively compressed format, 2) much faster to process than the semi-structured format, and 3) allows for random access to the data. For our experiments, we used the planet file available at http://planet.osm.org/pbf/planet-latest.osm.pbf (accessed on May 17^{th}, 2016). In order to parse the file, we have used the Python library imposm (2.6.0).

2) Transformation: Many objects modeling railway lines are attached with a tag *maxspeed*. This value of such a tag indicates the maximum legal speed allowed on a segment/relation. Such information is particularly important in a railway context, because it helps to deduce whether a line can be used for high-speed railway service. However, the syntax of values for this key is rather inconsistent, as different mappers use different notations. For instance, among the top values for the key maxspeed, we found the following variants: 'X' (=a number indicating the speed in km/h), 'X mph' (=a number indicating the speed in m/h), 'Xmph' (=as before). In addition, several mappers use the key *highspeed* to indicate that a way segment has a maxspeed of at least 200 km/h. Furthermore, railway segments/relations are often tagged with the key *gauge*, which describes the distance between the inside of the rails. For instance, many gauge values have the keys 1435 or 1520. During this stage we first normalize the values of maxspeed by converting them to km/h (without unit). Our method can accurately process all top 100 frequently occurring variants of maxspeed values. Second, we propagate the values of maxspeed and gauge from each relation to its referenced way segments (values are only replaced if non-existent for way segments).

3) Loading: Finally, the data is loaded according to the structure $\langle N, L, S, f_N, f_L \rangle$. In our implementation, we keep all node-related information from OSM in f_N. These include: Latitude/longitude, level, name (in different languages), operator, uic_refs, and other meta data. We use latitude/longitude pairs for connecting nodes to nearby segments. The other values are mainly used for presentation purposes. In addition, we keep three types of information in f_N for each link: Gauge, maxspeed, and spatial length of the link. The gauge value is used below to decide whether two network segments can be connected; we only connect networks with the same gauge values. The maxspeed information together with the spatial length is used to estimate the time required to travel long a segment, by dividing spatial distance through maxspeed.

Algorithm 1 Finding Overlapping Pairs of Components

Input: $WRN = \langle N, L, S, f_N, f_L \rangle$
Output: Pairs of components
Let C be the components in WRN_{in}
Let $pairs = \emptyset$
for $\langle N_1, L_1, S_1, f_{N_1}, f_{L_1} \rangle \in C$ **do**
  Let bb_1 be the bounding box of N_1, with margin δ
  for $\langle N_2, L_2, S_2, f_{N_2}, f_{L_2} \rangle \in C$ **do**
    Let bb_2 be the bounding box of N_2, with margin δ
    if $bb1$ and $bb2$ overlap **then**
      if $|N_1| > |N_2|$ **then**
        $pairs = pairs \cup \{\langle N_1, L_1, S_1, f_{N_1}, f_{L_1} \rangle, \langle N_2, L_2, S_2, f_{N_2}, f_{L_2} \rangle\}$
      else
        $pairs = pairs \cup \{\langle N_2, L_2, S_2, f_{N_2}, f_{L_2} \rangle, \langle N_1, L_1, S_1, f_{N_1}, f_{L_1} \rangle\}$
      end if
    end if
  end for
end for
return: $pairs$

B. Connecting the Network

The railway network, as modeled in OSM, is rather disconnected, yielding a few large and many smaller components. The major reason is that mappers often create unnecessary nodes when creating new segment endpoints. Since we are interested in creating a global network, we need to come up with a strategy to connect components. Obviously, connecting components in an arbitrary way is not a good idea, since we might connect parts of the network which are not connected in reality. Therefore, we need to design a strategy, which can connect components carefully, yet automatically. Moreover, the method needs to be scalable, if applied to the global network with millions of nodes.

The problem of connecting the network is broken down into a filter-and-refine framework, in order to solve scalability issues: First, we identify overlapping components, based on their geographical proximity. Second, we decide for interesting components whether and how they should be connected. This method is inspired by state-of-the-art techniques in string similarity join, where the goal is to find pairs of strings within a given (edit) distance threshold [35].

Our algorithm for finding overlapping pairs of components is shown in Algorithm 1. Essentially, this algorithm iterates

over all pairs of components and checks whether their bounding box overlaps. Whenever the bounding boxes overlap, the two components are added to the result. For the computation of the bounding box (extracted from latitude/longitude coordinates of each node in the components), we add a small margin of δ to each value. Without adding the margin, it would be possible that two components have non-overlapping bounding boxes, yet two nodes, one from each component, are spatially close to each other. This cases happens, if both nodes belong to the convex hull of their component. While the time complexity of the algorithm is quadratic in the number of components, the runtime in practice is rather small: There are much fewer components than nodes in the WRN and the check for bounding box overlap takes only a few CPU cycles. It should be noted that the first component in the result set is always the larger component: The rational is that we want to avoid merging small components to other small components, but rather attach small components to the few larger ones. After the termination of Algorithm 1, we have discarded several components pairs, which cannot be connected, because they are two far away from each other.

Given a set of interesting component pairs, we decide next whether and how to connect the components. Intuitively, we want to merge two nodes from different components, if the following two conditions are satisfied:

1) The distance between two nodes is smaller than a threshold $\delta Comp$.
2) The nodes are compatible, i.e., their gauge values are either identical or at least one node's gauge value is missing.

The algorithm for merging components is shown in Algorithm 2. In order to compute the nearest nodes between two components, we could enumerate all pairs of nodes and compute their distance. This process is time consuming, if the number of nodes in each component is large (our initial experiments following this naive approach did not finish to compute within one week on a high-class server with 32 cores and 320 GB RAM). In order to avoid checking all pairs of nodes, we first compute a KD tree [36] for all nodes in the larger component. The KD tree is a variant of binary search trees, where each node in the tree corresponds to a multi-dimensional data point. Given a set of geospatial datapoints, each node in the KD tree creates a hyperplane, which splits the to-be-indexed nodes into two parts. Given a divide-and-conquer strategy, KD trees can search the nearest point over a collection of nodes in logarithmic time, avoiding to iterate all nodes. A visualization of a KD tree for a small set of points is shown in Figure 1.

Since a component is potentially merged with several other interesting components, we compute and store the KD trees of the larger components before processing the components pairs in a loop. During the loop, for each component, we iterate over all nodes in the component and search the nearest nodes in the larger component using the KD tree. For all matching nodes, we check whether the two nodes can be linked together, i.e., their distance is within threshold δ_{Comp} and whether they are sharing the same gauge value, and if yes, we record a new link between both nodes in the variable $NewLinks$. Here, we use the notation $f(n)[gauge] =''$ to state that node n does not have a gauge value assigned. Furthermore, we update the tags of the new link accordingly. After executing Algorithm 2, many small components are merged with the larger components in the WRN.

Algorithm 2 Merging Components

Input: Set of pairs of interesting components $CP = \{(ca_1, cb_1), \ldots, (ca_m, cb_m)\}$
Output: New links for the WRN

Let $NewLinks = \emptyset$
Let $NewLinkAnnotation = \emptyset$
Let $allcomps = \{c \mid \exists X.((c, X) \in CP)\}$
for $c \in allcomps$ **do**
  Let KD_c be the KD-tree computed from lat/lon coordinates of nodes in c
end for
for $(ca, cb) \in CP$ **do**
  Let $\langle N_b, L_b, S_b, f_{N_b}, f_{L_b} \rangle = cb$
  for $n \in N_b$ **do**
    Use KD_{ca} to find the all nodes in ca whose distance to n is within a threshold $\delta Comp$; let the result be $candNodes$
    for $n_{cand} \in candNodes$ **do**
      if $f_{N_b}(n)["gauge"] = f_{N_b}(n_{cand})['gauge']$ or $f_{N_b}(n)['gauge'] =''$ or $f_{N_b}(n_{cand})['gauge'] =''$ **then**
        Add $(n, n_{cand}$ to $NewLinks)$
        Set $NewLinkAnnotation(n, n_{cand})['gauge'] = max(f_{N_b}(n)["gauge"], f_{N_b}(n_{cand})['gauge'])$
      end if
    end for
  end for
end for
return: $pairs$

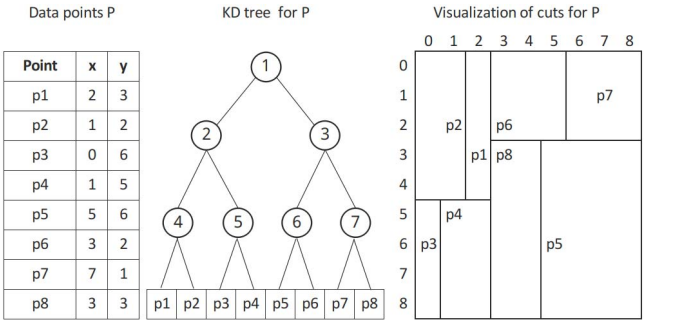

Fig. 1. Example for a KD tree: The input consists of 8 points p1–p8 (left). The KD tree has three hierarchical levels (center). Each cut of the KD tree splits the plane recursively into two halves (right).

C. Connecting Stations to the WRN

Mappers often model stations as nodes, but do not connect them to the network. Moreover, several railway stations are modeled as complex areas, without having a distinct node representing the station. In Figure 2, we show an example for

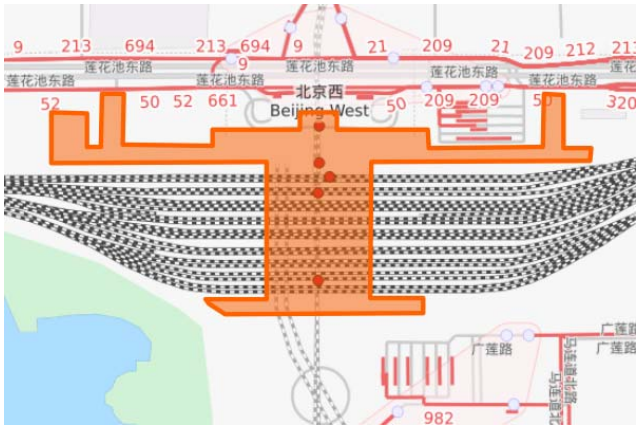

Fig. 2. Modeling of Beijing West railway station as areas, instead of nodes. Data source: http://www.openstreetmap.org/way/30680817

one large railway station (Beijing West) in China modeled as a complex area. We create distinct nodes from these complex areas, by extracting all nodes and computing their mean latitude/longitude values. Afterwards, we create an artificial node pointing to the center of the area, and additionally take over all tags of the complex area to the newly created node. Please note that using the mean coordinates for the new node representing the station is safe only, if the points are close to each other. In the presence of outlier, a better way to compute the representative coordinate could be computed by taking the point with median coordinates or more sophisticated point aggregation techniques. However, we have not found any case in the worldwide railway network, where nodes attached to the station relations/ways could be considered as outlier: The nodes spanning up a way-modeled station are always within a few dozen meters.

Moreover, we need to address the problem of isolated nodes, many of which we have found in the data set. In general, given the KD tree of each component in the network (note that we have already computed those in Section II-B), we could try to find the closest nodes in each component and connect them with the station, if the distance is within a given threshold $\delta Stat$. However, in our experiments, this strategy failed for many stations, with the following reason: In surprisingly many cases, stations are next to a very long way segment, whose nodes are up to several km away from the station; particularly in rural areas: The infrastructure network in these regions is not modeled as accurate as inside cities, where we can often find many nodes next to stations. Thus, if a way segment follows roughly a straight line in reality, they are often modeled with only few nodes. In summary, the simple strategy of connecting closest nodes to stations does not work.

Therefore, we have developed a different approach in our study. As described above, we are actually interested in those segments, whose perpendicular distance to the node is small enough to connect the nodes, while the distance to the nodes spanning up the segment is not relevant. In our implementation, we have computed the four closest nodes to a station. For each way segment connected to one of these four nodes, we compute the perpendicular distance between the station and that way segment. Once the distance is smaller than a given threshold $\delta Stat$, we connect the station to that way segment.

Algorithm 3 Triangle Filtering

Input: $WRN = \langle N, L, S, f_N, f_L \rangle$
Output: Network $\langle N, L, S, f_N, f_L \rangle$ without triangles

```
for n ∈ N do
  if deg(n) == 2 then
    Let a, b be the neighbors of n in WRN
    if (a, b) ∈ L and length((a, b)) ≤ δTri and n ∉ S then
      Remove n from WRN (update N, L, f_N, f_L accordingly)
    end if
  end if
end for
return: WRN
```

D. Network Simplification

The goal of our study is to extract a global station-network from OSM. So far, we have extracted a detailed infrastructure model, containing several details not contributing to the station network. These details include, for instance, exact curves of railway lines inside cities at a resolution of few meters, redundant connections between nodes introduced by mappers, leftover railway spurs, yards, and sidings, off the main railway track. Within the process of network simplification, we aim to remove these details from the infrastructure network, in order to obtain a skeleton network between stations. The simplification consists of five steps, which are motivated and described below.

1) Coordinate Reduction: When mappers add new segments into the OSM database, there is a trade-off between accuracy and map coverage. In general, the goal is to have a wide coverage and high accuracy; yet, often coverage comes before accuracy: Initial efforts to add a new area to OSM might introduce low accuracy data, while updating denser areas should focus on accuracy.[2] Therefore, identical nodes inside OSM often have slightly different coordinates and identical ways are often mapped as parallel. Therefore, we perform an initial clustering of nodes, which merges nodes within a distance threshold into a single node. In our implementation, we merge nodes by rounding coordinates to a fixed number of digits σRes; i.e. the smaller σRes, the more nodes will be merged. The choice of the parameter σRes is discussed in Section III. In general, this step can be skipped, at the price of obtaining a significantly larger (=number of nodes/links) network in much greater detail. However, for the station network we do not care about resolution of a few meters.

2) Triangle Filtering: After the process of coordinate reduction, we found that network contains many small triangles. These triangles consist of one node n of degree two, opposite to the (longest) base of the triangle and two other nodes which

[2]See http://wiki.openstreetmap.org/wiki/Accuracy for discussion.

TABLE I
PARAMETERS FOR NETWORK SIMPLIFICATION AND THEIR ASSIGNED VALUES

Parameter	Description	Value
$\delta Comp$	Max. distance between components (in km)	0.1
$\delta Stat$	Max. distance between stations and ways (in km)	0.2
σRes	Rounding value for coordinates	2
δTri	Max. distance for triangle elimination (in km)	3
$\delta Rewr$	Max. distance for path rewriting (in km)	50
$\delta Elim$	Max. distance for leftover path removal (in km)	5
$minN$	Min. number of nodes per remained component	100
$minA$	Min. bounding box area per component (in km^2)	10

are connected to the remaining part of the network. Unless n is a station, we argue that these links can be removed from the network, since all connections can go through the base of the triangle. Note that for some analysis tasks, e.g., network robustness, it might be necessary to keep all triangles inside: During the failure at the base of the triangle, the other two connections can be used as backup/alternative. The process of triangle filtering is formalized in Algorithm 3. We iterate over all nodes and remove the nodes, which are linked to exactly two connected neighbors. For our station network, we have removed the shorter sides of all those triangles, as long as the base is shorter than a threshold δTri. If an analysis task requires a high level of detail, δTri should be chosen as small, or triangle filtering skipped. For the purpose of deriving a station network (where only stations are nodes), δTri can be even set to several kilometers.

3) Path Rewriting: While coordinate reduction mainly addresses mapping errors, path rewriting addresses a related problem: Given a path between two nodes (e.g., stations), we only care whether the two nodes are connected and how long the distance is. The actual physical layout of a path into segments is not important for our study; particularly small curves at the resolution of a few hundred meters. Therefore, within path rewriting, we identify maximum, cycle-free paths P in a network, such that all nodes on the path have at most a degree of two, none of the nodes is a station, gauge information along the path is compatible, and maxspeed is compatible. We replace links only, for which these two attributes coincide. If two consecutive links do not share the same maxspeed and gauge value, then we will not merge them during path rewriting. After discarding links with different maxspeed/gauge values, for each $p \in P$, we remove all segments from p and replace them by a single link from the start of p to the end of p. While replacing paths, we keep track of the spatial distance of all links on the path, in order to have accurate representations of distances in the network. In our implementation, we rewrite all paths of length at most $\delta Rewr$. Moreover, in case of multiple paths between the same pair of nodes, we only keep the shortest path.

4) Leftover Path Removal: After performing previous simplification steps, the network contains several links (a, b), such that a is a node with degree one and b is either a station or a node with degree of at least two, i.e., paths yielded by path rewriting. As long as node a is not a station, it is safe to remove the link (a, b) from the network, since no inter-station connection will be affected. Therefore, we remove all such links with a distance below threshold $\delta Elim$. The case of (b, a) is processed in the same way as (a, b).

5) Removal of Irrelevant Components: After merging components in the network (Section II-B), we still have (usually very small) components, which are not connected to the main network. These components do not significantly contribute to the skeleton network, particularly if they do not contain any stations. Therefore, we remove all components from the network, which satisfy any of the three constraints below:

1) There are less than two stations connected to the component c.
2) The number of nodes in the component is less than threshold $minN$.
3) The area of the bounding box of the component is less than threshold $minA$.

After performing these simplification steps, we obtained the final WRN skeleton; please note that we have also removed all redundant links, i.e. links connecting the same pairs of nodes. In Table I, we give an overview about the parameters for network simplification and the values that were assigned in our evaluation (see below).

III. RESULTS

In this section, we report the results for our preliminary analysis of the WRN. In Section III-A, we evaluate and discuss the influence of parameters on the extraction of the railway network from OpenStreetMap. Section III-B presents the largest components of the WRN and discusses the applicability of the network for high-speed railway analysis. All experiments were executed on a server with 32 cores and 320 GB RAM, running Fedora 21 (Linux 4.1.13-100.fc21.x86_64).

A. Parameter Estimation

Our method for extracting the WRN from OpenStreetMap has several parameters, which guide the quality (level of detail in the network) and efficiency (in terms of running time of the algorithm). In this subsection, we analyze the sensitivity of all parameters used in our study. Since the computation of the WRN is rather time consuming, computing a single network snapshot takes between a few hours and more than a day, we do not perform sensitivity analysis on the whole network, but on selected subnetworks. For our study, we have chosen three regional networks: 1) the network of Berlin-Brandenburg area, 2) the complete network of China, and 3) the complete network of Australia-Oceania. Given the difference in size and geographical position, these three examples serve well for our purpose of discussing the effects of different parameter choices.

1) Merging Components: The maximum distance $\delta Comp$ decides whether two nodes from different components should be connected. If this parameter is too small, then the resulting network contains many components; several of which are introduced by mapping errors. On the other hand, if we chose this parameter too large, then we merge components that are not supposed to be merged, because they represent different subnetworks. In Figure 3, we show the number of

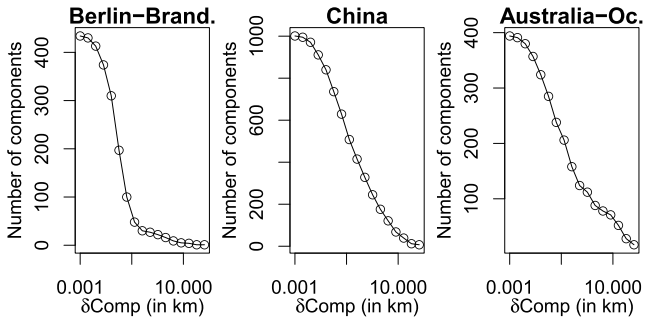

Fig. 3. Distance threshold $\delta Comp$ plotted against the number of components in the networks.

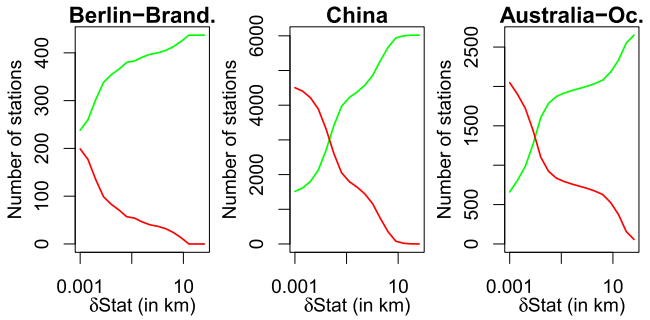

Fig. 4. Distance threshold $\delta Stat$ plotted against the number of stations in the networks (connected=green, disconnected=red).

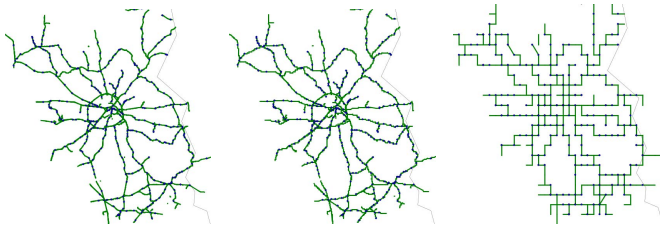

Fig. 5. Sensitivity analysis of the parameter σRes. Decreasing σRes from 3 to 1, the network develops from a fine-grained representation into a coarser structure. (a) $\sigma Res = 3$. (b) $\sigma Res = 2$. (c) $\sigma Res = 1$.

components in the three regional networks, with a varying distance threshold $\delta Comp$. The values can be fitted by an S-curve, with a slow initial decrease for small $\delta Comp$, a rapid decrease with medium distances and a rather slow decrease for large $\delta Comp$. For the remainder of our study, we chose $\delta Comp = 0.1$, which means that parts of the network which are within 100 m, are considered to be connectable. For some applications, this threshold might be too high; yet for our purpose of creating the station network, it turned out to be appropriate.

2) Connecting Stations: The maximum distance $\delta_{Station}$ decides how stations are connected to the surrounding infrastructure. Again, choosing this threshold is a tradeoff between the elimination of mapping errors and the merging of physically unconnected stations. In Figure 4, we show the number of connected and unconnected stations in the three regional networks, with a varying distance threshold $\delta Stat$. We can observe a similar S-curve as with the merging of

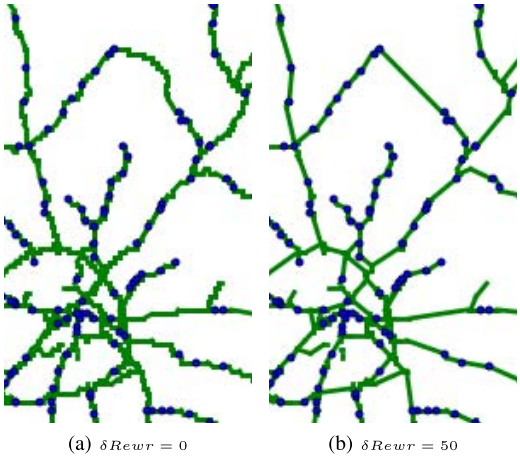

Fig. 6. Effect analysis of the parameter $\delta Rewr$. Increasing $\delta Rewr$, many links between stations and infrastructure crossings become straight lines; thereby abstracting the network structure into a skeleton.

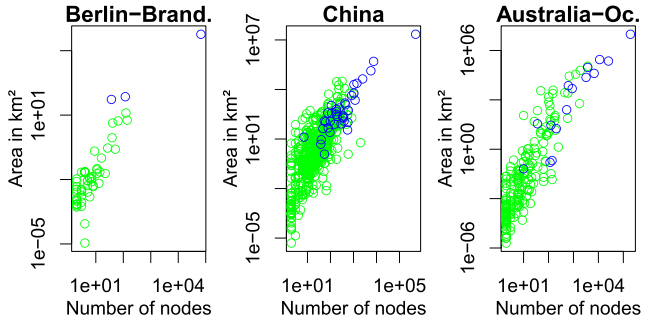

Fig. 7. Number of nodes in components plotted against the area as induced by the bounding box (Blue: components contains stations, Green: component without stations). It can be seen that there exists a few larger components containing stations and many smaller components without stations connected to them. Components without stations can be discarded in our study, since they do not contribute to the connectivity between stations.

components. For the rest of our study, we chose $\delta Stat = 0.2$, which means that a station is connected to infrastructure within 200 m. This resolution is sufficient in our case; with a focus on the global station network. If we let $\delta Stat = 0.1$, we found that several larger railway stations, for instance, Hankou in Wuhan City, China, are not connected to the network, because the (center of the) station building is too far away from the railway infrastructure network.

3) Network Simplification: Next, we evaluate the influence of the two network simplification parameters σRes and $\delta Rewr$ on the network. In Figure 5, we show the results for $\sigma Res \in \{1, 2, 3\}$ with fixed $\delta Rewr = 0$. It can be seen that the networks $\sigma Res = 1$ are significantly coarser than for the other values, which makes the network rather unusable. Starting from $\sigma Res = 2$, the overall network structure is preserved, Moreover, in Figure 6, we can clearly see the effect of path rewriting for $\delta Rewr \in \{0, 50\}$ with fixed $\sigma Res = 2$. For $\delta Rewr = 0$, the links between stations follow the real infrastructure. For $\delta Rewr = 50$, the links between stations (and important infrastructure nodes) are mostly reduced to straight lines. It should be noted that the number of nodes and links varies significantly between these networks. For the

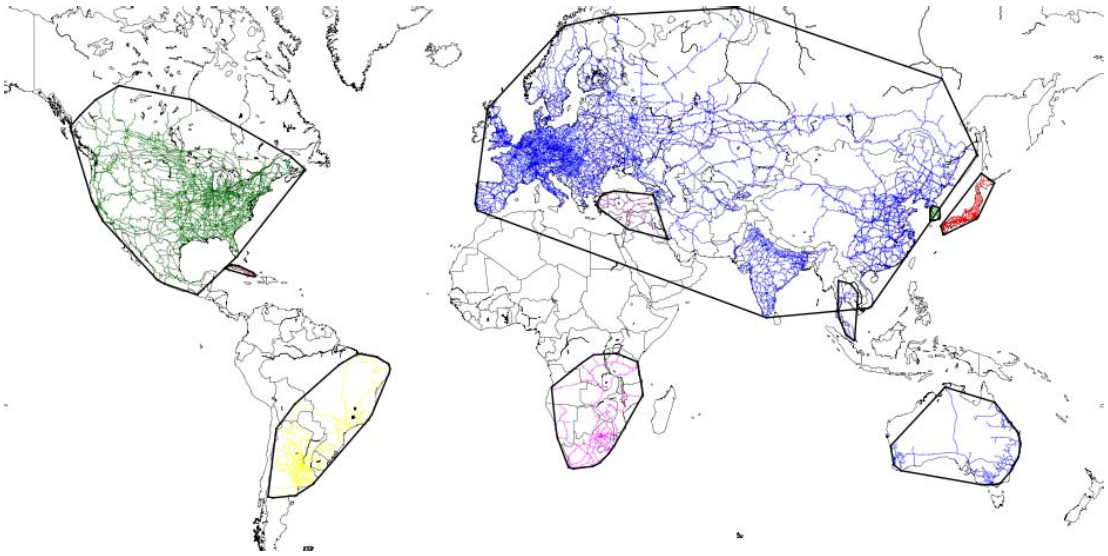

Fig. 8. The ten largest components in the worldwide railway network.

TABLE II
TOP TEN COMPONENTS OF THE WORLDWIDE RAILWAY NETWORK, ORDERED BY THE NUMBER OF NODES. BB STANDS FOR BOUNDING BOX OF THE COMPONENT

Region	BB (km^2)	$\|N\|$	$\|L\|$	Density
Europe/Asia	112,143,684	151,910	174,011	0.00001
North America	38,356,639	21,326	25,527	0.00011
Japan	2,409,088	21,265	23,003	0.00010
South Africa	8,841,590	5,401	5,880	0.00040
Australia	9,675,105	5,012	5,504	0.00043
Turkey	2,582,037	2,300	2,647	0.00100
South Korea	104,832	1,292	1,508	0.00180
South Asia	1,251,721	826	903	0.00260
Indonesia	245,752	814	907	0.00274

remainder of our study, we chose $\sigma Res = 2$ and $\delta Rewr = 50$. Moreover, we let $\delta Tri = 3$, in order to remove smaller triangles from the network (to improve the results of path rewriting) and $\delta Elim = 5$, in order to remove station-less links in the periphery of the network.

4) Unnecessary Components Filtering: We evaluate the two remaining parameters for filtering unnecessary components, $minN$ and $minA$, in Figure 7. We can see that there are usually a few components covering a large area and covering the large majority of nodes, while many smaller components are composed of 10–100 nodes only. It is interesting to notice that stations occur usually in larger components. Therefore, in the remaining part, we set $minN = 100$ and $minA = 10$.

B. Worldwide Railway Network: Preliminary Analysis

After tuning the parameters according to previous sections, we have extracted the worldwide railway network. In Figure 8, the ten largest components are depicted with different color. Table II, summarizes statistical properties of these components. The largest components covers Europe and large part of East Asia (plus India). The second largest component covers North America, followed by Japan and South America. The number of nodes in the top three networks is significantly larger than for the remaining components. However, statistical properties are quite similar, with a very low density. We have further analyzed the degree distribution. Most of the nodes have exactly two neighbors: Either being constituted by a station (most frequently) or as a node not yet eliminated by path rewriting.

In Figure 9, we plot all railway links with a maxspeed value larger than 200 km/h. As it can be seen, the result closely resembles the well known high-speed railway regions across the world, including larger areas in Spain, France, Germany, China, Japan, and South Korea. Nevertheless, it should be kept in mind, that the information in OSM might be incomplete. The results of our analysis together with the code, can be used by other researchers and mappers to identify regions for which the data about railway transportation is incomplete.

The worldwide railway network, as extracted in our study, can be used to predict the travel time between stations (or cities) in the network. In order to estimate the travel time between two stations, we compute the weights in the network as follows: For each link, we assign the weight $w = \frac{length}{speed}$, where $length$ is the physical length of the segment in km and $speed$ is the maxspeed for the segment (we use 100 km/h, if no data is available). In Figure 10, we plot the predicted travel times for four different regions: Australia, China, Europe, and USA. For Australia, we analyzed the five stations: North Melbourne, Sydney Redfern, Canberra, Wagga-Wagga, and Newcastle Broadmeadow. For China, we have used the following five major stations: Beijing West, Xi'an North, Zhengzhou East, Hankou, and Changsha South. For Central Europe, we plot the predicted and scheduled travel time between the following five major stations: Paris Nord, Munich Central Station, Berlin Hauptbahnhof (tief), Strasbourg, and Bruxelles-Centrale. For USA, we have used main stations in the following five cities along the East Coast: New York, Baltimore, Philadelphia, Washington DC, and Boston. Scheduled travel times were looked up on train operator webpages (http://www.

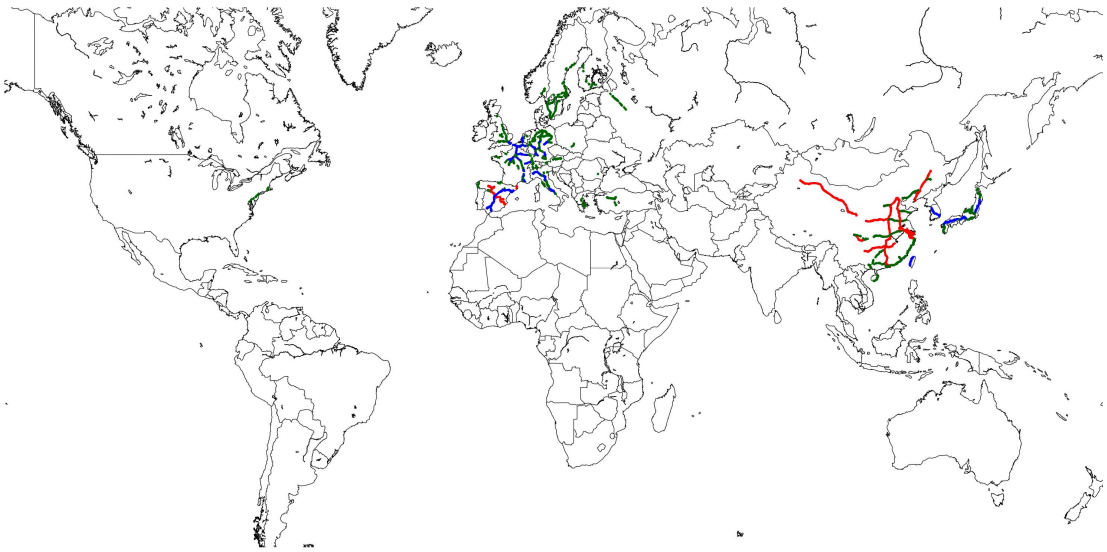

Fig. 9. Extract of the high-speed railway network. Speed limits are indicated as follows: Green (200–279 km/h), blue (280–349 km/h), and red ≥350 km/h). The snapshot correctly assembles the important HSR regions, such as, Spain, France, Germany, China, Japan, South Korea, and along the east coast of US.

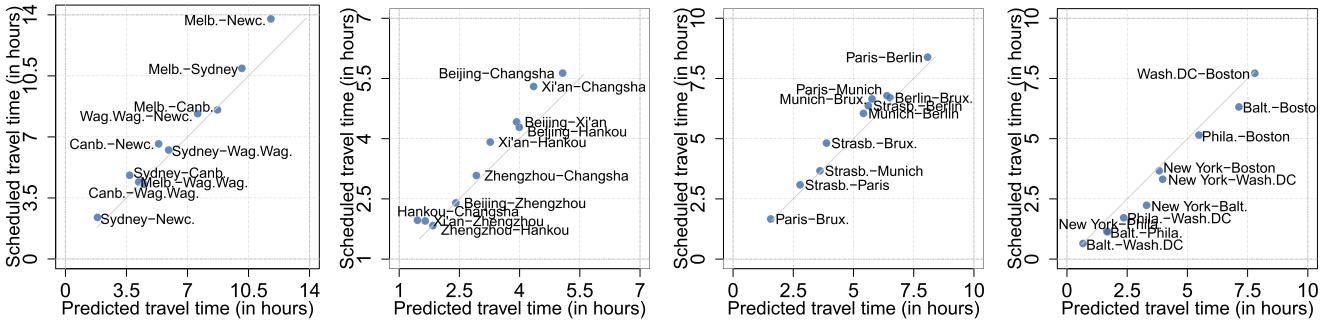

Fig. 10. Predicted travel time plotted against scheduled travel time for ten city pairs in Australia (a), China (b), and Europe (c), and USA (d).

nswtrainlink.info for Australia, https://kyfw.12306.cn/otn/leftTicket/init for China, http://www.bahn.de for Europe, and https://tickets.amtrak.com/itd/amtrak for USA); for each connection we selected the fastest available train connection.

It can be seen that the data provides an accurate estimation of the scheduled travel time. In fact the predicted travel times is often smaller, since in our simplified model a) we do not consider the time spent for stops at trains stations, b) we assume that a train always travels at maximal possible speed, c) that a train always follows the best possible path (i.e., shortest path in the travel time network), and d) that a passenger does not need to change trains. For USA, the predicted travel times are slightly higher than the real travel times. The reason is that the maxspeed value for railway lines of Amtrak is sometimes not tagged with a maxspeed value in OSM: In these cases our algorithm falls back to the normal railway speed, which is less than 50% of the actual speed for large parts of the infrastructure. The deviation from predicted travel times to actual travel times could be used to improve OSM data in the future, by adding appropriate maxspeed tags. It should be noted, that we have found some rare cases, for which the predicted travel time is significantly lower than the scheduled travel time. One such case is the connection between Xi'an North and Shanghai Hongqiao, whose predicted travel time is 5 hours, but the scheduled travel time is 12 hours. However, in June 2016, a new connection will be established between these two train stations, with a scheduled travel time of 5 hours. This example should also again underline that the travel times from the data are indeed a lower bound on the minimum reachability between stations/cities, not the actual travel time.

We have performed some additional experiments which show the impact of the network simplification on the performance of travel time estimation. First, we have measured the running time of travel distance estimations. On average, a single query to determine the distance between two random stations on the original graph takes around 120 seconds. However, on the reduced graph, these query times are reduced to 3-5 seconds for a station pair. This 20-fold reduction in running time justifies the simplification already. The second reason for reducing the graph is the memory consumption. The original graph representation of the worldwide network needs approx. 1 GB of memory without any indexing structures. Once loaded into main memory, using the network toolkit NetworkX, the graph requires 40 GB. The reduced graph, on the other hand, requires only 45 MB on the hard disk

and less than 2 GB of main memory. Therefore, we believe that the reduction of nodes/links is of practical relevance.

IV. Conclusions

In this paper, we proposed a method to extract the worldwide railway network, where a node is a waypoint/station and a link between two nodes describes whether two nodes are physically connected in the infrastructure network. As a data source, we used the free OpenStreetMap. We solved several data management problems, such as data cleansing and scalability. Our work is a contribution towards the ability of better understanding global mobility patterns, in the face of multimodal transportation. Researchers can use our methods and algorithms to investigate large-scale cooperation/competition between air transport and railway systems, the analysis of global multi-layer transportation resilience, and more accurate mode-dependent origin-destination demand estimation.

For future work, it would be interesting, yet challenging, to enhance the infrastructure data with real passenger data and schedule information. One way would be to regularly scrape webpages of railway operators or try to use other open data sets, such as GTFS-based releases. However, to the best of our knowledge, most of such datasets are for local regions only. Merging data into a consistent global database is a challenging task; yet, the gains of such a scientific data set clearly outweigh the efforts. Overall, we hope that our work spurs the development of a global railway database, free to use for academic research.

References

[1] W. Wei and M. Hansen, "An aggregate demand model for air passenger traffic in the hub-and-spoke network," *Transp. Res. A, Policy Pract.*, vol. 40, no. 10, pp. 841–851, Dec. 2006.

[2] E. van der Hurk, L. Kroon, G. Maróti, and P. Vervest, "Deduction of passengers' route choices from smart card data," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 430–440, Feb. 2015.

[3] M. Givoni and D. Banister, "Airline and railway integration," *Transp. Policy*, vol. 13, no. 5, pp. 386–397, Sep. 2006.

[4] M. Zanin, "Can we neglect the multi-layer structure of functional networks?" *Phys. A, Statist. Mech. Appl.*, vol. 430, pp. 184–192, Jul. 2015.

[5] O. Lordan, J. M. Sallan, P. Simo, and D. Gonzalez-Prieto, "Robustness of the air transport network," *Transp. Res. E, Logistics Transp. Rev.*, vol. 68, pp. 155–163, Aug. 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1366554514000805

[6] L. Cadarso, G. Maróti, and Á. Marín, "Smooth and controlled recovery planning of disruptions in rapid transit networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 2192–2202, Aug. 2015.

[7] C. Behrens and E. Pels, "Intermodal competition in the London–Paris passenger market: High-speed rail and air transport," *J. Urban Econ.*, vol. 71, no. 3, pp. 278–288, May 2012.

[8] H. Yang and A. Zhang, "Effects of high-speed rail and air transport competition on prices, profits and welfare," *Transp. Res. B, Methodol.*, vol. 46, no. 10, pp. 1322–1333, Dec. 2012.

[9] B. Ai *et al.*, "Challenges toward wireless communications for high-speed railway," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2143–2158, Oct. 2014.

[10] L. Zhao, B. Cai, J. Xu, and Y. Ran, "Study of the track–train continuous information transmission process in a high-speed railway," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 112–121, Feb. 2014.

[11] F. Xie and D. Levinson, "Modeling the growth of transportation networks: A comprehensive review," *Netw. Spatial Econ.*, vol. 9, no. 3, pp. 291–307, Sep. 2009. [Online]. Available: http://dx.doi.org/10.1007/s11067-007-9037-4

[12] R. Louf, C. Roth, and M. Barthelemy, "Scaling in transportation networks," *PLoS ONE*, vol. 9, no. 7, p. e102007, 2014.

[13] J. Wang, F. Jin, H. Mo, and F. Wang, "Spatiotemporal evolution of China's railway network in the 20th century: An accessibility approach," *Transp. Res. A, Policy Pract.*, vol. 43, no. 8, pp. 765–778, Oct. 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S096585640900072X

[14] M. Zanin and F. Lillo, "Modelling the air transport with complex networks: A short review," *Eur. Phys. J. Special Topics*, vol. 215, no. 1, pp. 5–21, Jan. 2013.

[15] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Rev. Modern Phys.*, vol. 74, no. 1, pp. 47–97, Jan. 2002. [Online]. Available: http://link.aps.org/doi/10.1103/RevModPhys.74.47

[16] W. Li and X. Cai, "Empirical analysis of a scale-free railway network in China," *Phys. A, Statist. Mech. Appl.*, vol. 382, no. 2, pp. 693–703, Aug. 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378437107003949

[17] M. Ouyang, L. Zhao, L. Hong, and Z. Pan, "Comparisons of complex network based models and real train flow model to analyze chinese railway vulnerability," *Rel. Eng. Syst. Safety*, vol. 123, pp. 38–46, Mar. 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0951832013002792

[18] P. Sen, S. Dasgupta, A. Chatterjee, P. A. Sreeram, G. Mukherjee, and S. S. Manna, "Small-world properties of the indian railway network," *Phys. Rev. E*, vol. 67, no. 3, p. 036106, Mar. 2003. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevE.67.036106

[19] S. Ghosh *et al.*, "Statistical analysis of the indian railway network: A complex network approach," *Acta Phys. Polonica B Proc. Suppl.*, vol. 4, no. 2, pp. 123–138, 2011.

[20] J. Martí-Henneberg, "European integration and national models for railway networks (1840–2010)," *J. Transp. Geogr.*, vol. 26, pp. 126–138, Jan. 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0966692312002517

[21] K. A. Seaton and L. M. Hackett, "Stations, trains and small-world networks," *Phys. A, Statist. Mech. Appl.*, vol. 339, nos. 3–4, pp. 635–644, Aug. 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378437104003036

[22] J. Martí-Henneberg, "Attracting travellers to the high-speed train: A methodology for comparing potential demand between stations," *J. Transp. Geogr.*, vol. 42, pp. 145–156, Jan. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0966692314002397

[23] J. I. Castillo-Manzano, R. Pozo-Barajas, and J. R. Trapero, "Measuring the substitution effects between high speed rail and air transport in Spain," *J. Transp. Geogr.*, vol. 43, pp. 59–65, Feb. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0966692315000101

[24] W. Li, Q. A. Wang, L. Nivanen, and A. L. Méhauté, "How to fit the degree distribution of the air network?" *Phys. A, Statist. Mech. Appl.*, vol. 368, no. 1, pp. 262–272, Aug. 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378437106001051

[25] D. Zielstra and A. Zipf, "Quantitative studies on the data quality of openstreetmap in Germany," in *Proc. GISci.*, 2010, pp. 1–7.

[26] S. Zheng and J. Zheng, "Assessing the completeness and positional accuracy of openstreetmap in China," in *Thematic Cartography for the Society*. Cham, Switzerland: Springer, 2014, pp. 171–189.

[27] M. A. Brovelli, M. Minghini, M. Molinari, and P. Mooney, "Towards an automated comparison of openstreetmap with authoritative road datasets," *Trans. GIS*, doi: 10.1111/tgis.12182. [Online]. Available: http://dx.doi.org/10.1111/tgis.12182

[28] D. Partha and P. A. David, "Toward a new economics of science," *Res. Policy*, vol. 23, no. 5, pp. 487–521, Sep. 1994.

[29] J. C. Molloy, "The open knowledge foundation: Open data means better science," *PLoS Biol.*, vol. 9, no. 12, p. e1001195, 2011. [Online]. Available: http://dx.doi.org/10.1371%2Fjournal.pbio.1001195

[30] Y. Zhao and P. Ioannou, "Positive train control with dynamic headway based on an active communication system," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 3095–3103, Dec. 2015.

[31] M. Haklay and P. Weber, "OpenStreetMap: User-generated street maps," *IEEE Pervasive Comput.*, vol. 7, no. 4, pp. 12–18, Oct. 2008.

[32] P. Neis and D. Zielstra, "Recent developments and future trends in volunteered geographic information research: The case of openstreetmap," *Future Internet*, vol. 6, no. 1, pp. 76–106, 2014.

[33] P. Vassiliadis, "A survey of extract–transform–load technology," *Int. J. Data Warehousing Mining*, vol. 5, no. 3, pp. 1–27, 2009.

[34] K. Varda, "Protocol buffers: Google's data interchange format," Google, Mountain View, CA, USA, Tech. Rep., Jun. 2008. [Online]. Available: http://google-opensource.blogspot.com/2008/07/protocol-buffers-googles-data.html

[35] S. Wandelt *et al.*, "State-of-the-art in string similarity search and join," *SIGMOD Rec.*, vol. 43, no. 1, pp. 64–76, Mar. 2014. [Online]. Available: http://doi.acm.org/10.1145/2627692.2627706
[36] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Commun. ACM*, vol. 18, no. 9, pp. 509–517, 1975.

Sebastian Wandelt received the Ph.D. degree in computer science from Hamburg University of Technology. He is a Professor with the School of Electronic and Information Engineering, Beihang University, and the Beijing Key Laboratory for Network-Based Cooperative ATM, Beijing. His research interests are transportation systems, scalable data management, and compressing/searching large collections of objects.

Zezhou Wang is the bachelor's degree with the School of Electronic and Information Engineering, Beihang University. His major research interest is in railway and communication systems engineering.

Xiaoqian Sun received the Ph.D. degree in aerospace engineering from Hamburg University of Technology in 2012. She is an Associate Professor with the School of Electronic and Information Engineering, Beihang University, and the Beijing Key Laboratory for Network-Based Cooperative ATM, Beijing. Her research interests mainly include air transportation networks, multimodal transportation, and multicriteria decision analysis.