ISPRS Int. J. Geo-Inf. **2012**, *1*, 186-208; doi:10.3390/ijgi1020186

OPEN ACCESS

ISPRS International Journal of **Geo-Information**

ISSN 2220-9964

www.mdpi.com/journal/ijgi/

Article

Using Crowdsourced Geodata for Agent-Based Indoor Evacuation Simulations

Marcus Goetz * and Alexander Zipf

GIScience Research Group, Institute of Geography, University of Heidelberg, Berliner Straße 48, D-69120 Heidelberg, Germany; E-Mail: zipf@uni-heidelberg.de

* Author to whom correspondence should be addressed; E-Mail: m.goetz@uni-heidelberg.de; Tel.: +49-6221-54-5501; Fax: +49-6221-54-4529.

Received: 4 June 2012; in revised form: 20 August 2012 / Accepted: 21 August 2012 / Published: 29 August 2012

Abstract: Crowdsourced geodata has been proven to be a rich and major data source for environmental simulations and analysis, as well as the visualization of spatial phenomena. With the increasing size and complexity of public buildings, such as universities or hotels, there is also an increasing demand for information about indoor spaces. Trying to stimulate this growing demand, both researchers and Volunteered Geographic Information (VGI) communities envision to extend established communities towards indoors. It has already been showcased that VGI from OpenStreetMap (OSM) can be utilized for different applications in Spatial Data Infrastructures (SDIs) as well as for simple shortest path computations inside buildings. The here presented research now tries to utilize crowdsourced indoor geodata for more complex indoor routing scenarios of multiple users. Essentially, it will be investigated if, and to what extent, the available data can be utilized for performing indoor evacuation simulations with the simulation framework *MATSim*. That is, this paper investigates the suitability of crowdsourced indoor information from OSM (*IndoorOSM*) for evacuation simulations. Additionally, the applicability of *MATSim* for agent-based indoor evacuation simulations is conducted. The paper discusses the automatic generation simulation-related data, and provides experimental results for two different evacuation scenarios. Furthermore, limitations of the *IndoorOSM* data and the *MATSim* framework for indoor evacuation simulations are elaborated and discussed.

Keywords: agent-based simulation; emergency management; evacuation simulation; indoor evacuation; IndoorOSM; MATSim; OpenStreetMap (OSM); rescue planning

1. Introduction

The last couple of years yielded the phenomenon of crowdsourced geodata (also known as Volunteered Geographic Information, VGI). Thereby, geographic data is collaboratively collected by users (both amateurs and professionals) and shared on an online community platform. VGI comprises a special kind of user-generated content (UGC), whereby the spatial component, *i.e.*, the geo-location of a distinct feature, is an integral part of the collected data. It has already been demonstrated that VGI can be used for various kinds of applications and analysis, such as vehicle routing [1,2], emergency routing [3], or traffic simulation studies [4,5]. Furthermore, it has been demonstrated that VGI, especially from OpenStreetMap (OSM), can (potentially) serve as a major dataset [6–8] for urban areas.

With the increasing size and complexity of the interior structure of public buildings, institutions and facilities, such as universities, hotels, or airports, there is also an increasing demand on data—and particularly information—about indoor environments [9,10]. Trying to stimulate this demand and furthermore to use the momentum of the crowd intelligence of VGI, there are efforts to extend OSM to indoor spaces. Originating from research on the requirements of crowdsourced indoor geodata [11], there is a detailed *IndoorOSM* mapping proposal available [12]. It has already been demonstrated that *IndoorOSM* contains very detailed information about indoor environments, which can be utilized for automatically generating standardized 3D city models [13]. Essentially, this means that OSM can be regarded as a rich and powerful (additional) data source for Spatial Data Infrastructures (SDI) (*cf.* [14]), as well as professional applications and analysis for built environments, such as environmental simulations and facility management [15], or emergency response and rescue operations [16]. Furthermore, it has already been showcased that crowdsourced geodata can be utilized for developing and providing indoor routing services with shortest path computation functionality in 2D [17,18] and 3D client applications [19].

In urban built environments and their interior spaces, it seems obvious that, especially during rush-hours, there are a lot of people inside a building at the same time. Once an incident or emergency occurs, people typically start to search for their nearest exit, thus to leave the 'unsecure' building. However, in such emergency situations the collective human behavior is rather unstructured, thus a fast and secure evacuation is not realizable in an easy manner. Additionally, such situations often lead to (mass) panic and crowd stampede which might cause serious injury to humans as people are crushed or trampled. To avoid such harm, it is important to identify potential bottlenecks or blind alleys in the building layout prior to constructing a new building or planning an event in an existing building, not only from the perspective of the designers and architects, but also from the perspective of the legislators. However, "*it is not an easy task to predict evacuation performances for large buildings with complex layouts*" [20]. Since the costs (both monetary and temporally) of practical simulations cannot be easily afforded, evacuation simulations with computers became popular within the last years [20]. Some exemplary scenarios for such simulations are: (1) a planned and structured site clearing of a hospital due to a predicted flood, (2) a fast but safe site clearing of a building due to a hostage crisis on a distinct building floor or (3) a rapid and unplanned evacuation due to a fire. It becomes apparent that there are different requirements for the evacuation as well as different stress-levels of the individual humans. By incorporating different parameters and requirements, it is assumed to create decision support for rescue operations or to mitigate disaster situations [21]. However, it needs to be stated that evacuation simulations always depend on the simulation models, agents and assumptions. That is, simulations are subject to impreciseness which needs to be considered while evaluating the simulation.

For retrieving probable and reasonable simulation results, detailed and fine grained data about the affected building is required. However, it is often difficult to obtain and maintain appropriate data. For instance, information about the different floor layouts, stairways and potential building exits need to be collected from different third parties, such as architects, building owners or public authorities. However, those ‘data providers’ often deliver the data in various (non-standard) formats, typically ranging from low-level pixel images, over vector-based drawings, up to (3D) Computer Aided Design (CAD) plans. Furthermore, the different data sets are usually not explicitly referenced to each other and extensive pre-processing is required. In contrast, crowdsourced (indoor) geodata probably provides a solution for the aforementioned drawbacks, because it can be regarded as a well-defined source for geodata with (potentially) global coverage and unrestricted accessibility.

Therefore, the main objective of this paper is to investigate if and to which extent *IndoorOSM* data can be used for evacuation simulations. Thereby, the available data and information will be discussed, as well as the possibilities for an automated generation of such simulations. Essentially, an automated simulation generation allow a fast application of the here presented approaches to other crowdsourced buildings in *IndoorOSM*. Furthermore, by performing two exemplary evacuation simulation scenarios, limitations and missing data of IndoorOSM will be revealed.

The rest of this paper is organized as follows: Section 2 describes related work, focusing on crowdsourced indoor geodata from OSM, as well as existing agent-based simulation frameworks in general and the *MATSim* framework (which has been used for the here presented proof-of-concept) in particular. Section 3 will focus on the (semi-) automatic generation of all relevant simulation data sources. As a proof-of-concept, Section 4 will depict experimental results of two different evacuation scenarios. Section 5 discusses limitations and problems of the *IndoorOSM* data as well as the *MATSim* framework. The last section serves as conclusion and outlook on future work.

2. Related Work

2.1. Crowdsourced (Indoor) Geodata from OpenStreetMap

Trying to benefit from a crowd-intelligence and utilizing thousands of humans acting as remote sensors [22], more and more Internet communities are aiming at the collection of not only user-generated content (UGC), but spatially-referenced user-generated content. Thereby, volunteers collaboratively collect, generate, enhance and share geodata in a Web 2.0 community platform. Contrary to proprietary data, the available data can be used for individual purposes at no charge. That is, community members benefit from a vast amount of various kinds of geodata, which they can use for their own applications. One of the most popular and most manifold sources for VGI is OSM. Initiated in 2004, OSM grew rapidly regarding the amount of contributors as well as amount of data. It has already been proven that data from OSM can be used for various kinds of applications and analysis, such as routing [1,2], emergency routing [3], or traffic simulation studies [4,5]. Furthermore, it has been demonstrated that (in urban areas) OSM is comparable to commercially collected geodata [6–8].

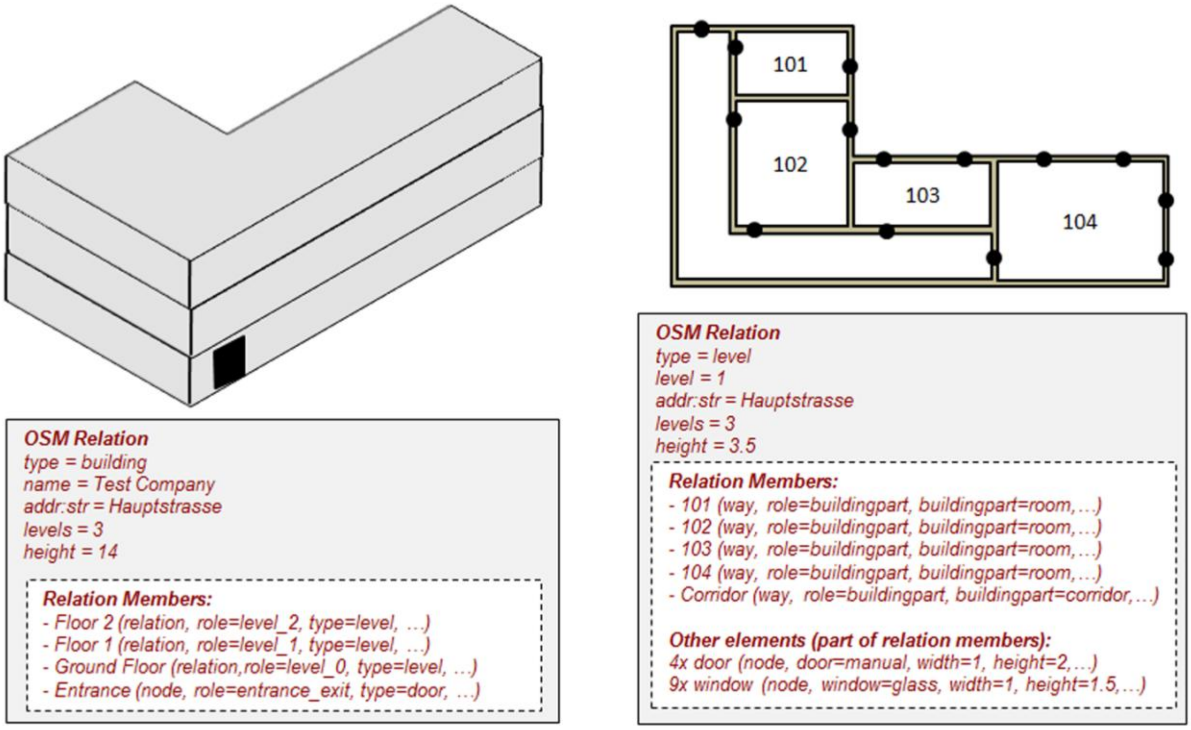
Regarding the data structure, OSM is kept rather simple: the users provide two-dimensional geometries which furthermore can be enriched (*i.e.*, *tagged*) with additional (semantic) information. In general, the user contribute single geo-referenced points (*i.e.*, *nodes*), which can be combined to so called *ways* for representing either lines (*i.e.*, a non-closed *way*) or polygons (*i.e.*, a closed *way*). Additionally, users can utilize *relations* for describing complex relationships between objects, such as turn restrictions or holes in a polygon. The tagging is realized via an open key-value pair methodology. That is, the contributors add an arbitrary key, representing some kind of information or information class (e.g., *building*, *highway etc.*) and additionally refine this information with some value (e.g., *airport*, *residential etc.*). Thereby, the amount of key-value pairs, as well as the keys and values themselves are unlimited. That is, a user can basically provide any kind of information. However, the most commonly used tags, *i.e.*, the community wide accepted map features, are listed on the map features wiki page [23]. In contrast, *Tagwatch* [24] provides an overview of all currently used tags, as well as some corresponding values. Additionally, the *OSM Wiki* [25] provides detailed (user-generated) descriptions for most key-value pairs.

Trying to benefit from the crowd intelligence and to disclose new possibilities, various OSM contributors try to use OSM for collecting indoor information. However, there is currently no community wide accepted standard or schema for mapping indoor information. There are several efforts in the community, varying in both mapping granularity and documentation [26]. To the authors, one of the most advanced proposals is the so called *IndoorOSM* mapping schema, because it provides detailed information about the interior structure of a building. Furthermore, *IndoorOSM* is the only proposal which originated from research on the demands and requirements of collaboratively collected indoor information [11]. It has already been demonstrated that *IndoorOSM* contains very detailed data which is suitable for generating standardized 3D CityGML building models [13] for the application within SDIs. Additionally, the usage of *IndoorOSM* for the development of indoor routing services with basic shortest path computation has already been demonstrate with a 2D client for both desktop computers [18] and mobile devices [17], as well as a real 3D route planning application [19]. However, *IndoorOSM* data has not yet been utilized for the conduction of more complex and advanced indoor route applications with multiple users. Essentially, IndoorOSM has not yet been used as a data source for indoor evacuation simulations. Nevertheless, since *IndoorOSM* does not only contain data about the shape of rooms, but also detailed information about the location and attributes of doors and windows—which is important for emergency situations—the available data of *IndoorOSM* is (potentially) suitable for indoor evacuation simulations.

For mapping indoor information in OSM, *IndoorOSM* utilizes existing OSM data types (*i.e.*, *nodes*, *ways*, *relations*, *tags*). A building is represented as a hierarchically structured object, consisting of several building floors (*i.e.*, *levels*). Each floor furthermore contains a distinct amount of building parts (*buildingpart*), such as rooms, corridors, stairways *etc.* That is, the complete building is represented as a relation in OSM (the main-relation). The different relation-members of this main-relation represent the different floors, whereby each of them is again represented as an OSM-relation. Each individual part of the corresponding floor is then mapped as an OSM element (typically a closed way for representing the geometry of the building part).

Additional (semantic) information, such as room names, level numbers, heights, *etc.*, can be added with key-value pairs to the corresponding OSM features. Information about doors or windows (which are very important for evacuation purposes) can be mapped by adding a *node* to the corresponding building part *way* and tagging it as window or door. Additionally, information about the width or height of a door/window can be tagged respectively. The basic hierarchical idea of *IndoorOSM* is also visualized in Figure 1(a). The composition of a detailed floor plan for an exemplary building level with rooms, corridors, doors and windows is furthermore depicted in Figure 1(b). For more information on the *IndoorOSM* mapping proposal please refer to the underlying research publication [11] as well as to the corresponding OSM wiki page [12].

Figure 1. The basic hierarchy of a complete building (**a**) and an exemplary detailed floor plan with rooms, corridor, doors and windows (**b**) in *IndoorOSM*.



OSM Relation
type = building
name = Test Company
addr:str = Hauptstrasse
levels = 3
height = 14

Relation Members:
- *Floor 2 (relation, role=level_2, type=level, ...)*
- *Floor 1 (relation, role=level_1, type=level, ...)*
- *Ground Floor (relation,role=level_0, type=level, ...)*
- *Entrance (node, role=entrance_exit, type=door, ...)*

OSM Relation
type = level
level = 1
addr:str = Hauptstrasse
levels = 3
height = 3.5

Relation Members:
- *101 (way, role=buildingpart, buildingpart=room,...)*
- *102 (way, role=buildingpart, buildingpart=room,...)*
- *103 (way, role=buildingpart, buildingpart=room,...)*
- *104 (way, role=buildingpart, buildingpart=room,...)*
- *Corridor (way, role=buildingpart, buildingpart=corridor,...)*

Other elements (part of relation members):
4x door (node, door=manual, width=1, height=2,...)
9x window (node, window=glass, width=1, height=1.5,...)

(**a**) (**b**)

2.2. Agent-Based Indoor Evacuation Simulation

Several disastrous emergencies in the past, such as the 9/11 terror attacks or the earthquake in Japan 2011, motivated discussions and research efforts on how to protect and evacuate persons inside a building during an emergency [27]. Furthermore, there are quite a lot of different research efforts on indoor (and outdoor) evacuation simulations. An agent-based simulation of spatial cognition as well as wayfinding in buildings for the scenario of a fire emergency is discussed by Hajibabai *et al*. [28]. Quite similar, an agent-based evacuation model for large public buildings under dynamic fire conditions is also available [20]. A hybrid approach that is a combination of both network simulation as well as free space model has been presented [21]. An assistant system based on indoor networks is described by Yamashita *et al*. [29]. Regarding large scenarios, it has also been showcased that agent-based simulation is suitable for outdoor evacuation simulations in Hong Kong [30]. Besides those, there are furthermore many other evacuation models, such as EGRESS, EXODUS, SIMULEX, EXITT, WAYOUT [31], available, which are all based on a network representation. Essentially, they are suitable for simulating the evacuation, as well as analyzing the evacuation efficiency for arbitrary buildings. A comprehensive overview about pedestrian evacuation simulation can be furthermore found in the book series *Pedestrian and Evacuation Dynamics* [32–37].

Obviously, there are several efforts towards agent-based indoor evacuation simulation. However, current approaches all utilize proprietary data. Essentially, none of the existing approaches utilizes crowdsourced indoor geodata. However, crowdsourced indoor geodata from OSM has the benefit that it is a non-proprietary internationally growing dataset. That is, developed applications and approaches can be used for any arbitrary building which is available in OSM. Therefore, within the remainder of this paper, possibilities for an efficient and automated provision of indoor evacuation simulations for arbitrary buildings based on crowdsourced indoor geodata will be investigated and discussed.

2.3. Multi-Agent Transport Simulation (MATSim)

As described in the previous section, there are quite lot of different simulation frameworks—all with advantages (e.g., iterative learning, consideration of different agents, modular extensibility *etc.*) and disadvantages (e.g., neglecting the third dimension, restricted to vehicle simulation *etc.*). However, when comparing a framework with the utilization of large multidimensional probability arrays, computational savings in favor of the framework should appear. Besides that, a large range of output options as well as explicit modeling techniques for individuals' decision making processes are desirable [38]. Especially the last point is very essential, because evacuation scenarios strongly depend on the decisions of the individuals. For the here conducted research it has been decided to use the so called *Multi-Agent Transport Simulation* (*MATSim*) framework, because it fulfills all the before mentioned requirements [14]. *MATSim* incorporates a multi-agent micro-simulation, describing that the behavior of each simulated person (*i.e.*, an agent) can be defined by individual parameters, such as age, profession, traveling plan *etc.* The key features of *MATSim* are a fast agent-based traffic simulation, support of large multi-level scenarios, sophisticated interactive visualizer, versatile analyses, modular approach, and active open source development [39].

MATSim is developed in Java, thus utilizable on many operation platforms. All required input files, such as the network, the population or the evacuation area (will be described later), are based on XML schemas. *MATSim* has already been used for a variety of different (evacuation) simulations, ranging from large-scale vehicle traffic simulations [40], over city evacuation scenarios [41–43], up to pedestrian evacuation [44]. However, up to the authors' knowledge, *MATSim* has not yet been used for simulating mass evacuation in a multi-level building. It has been proven that *MATSim* simulations produce more realistic results—especially from a temporal point of view—than other simulation frameworks [45]. It is based on a parallel queue model with a capacity constraint and a storage constraint [46]. The former constraint avoids that more agents leave a link in the network within a certain time than the flow capacity of this link. The latter constraint describes that a link can only contain a certain amount of agents at one point of time. That is, as soon as a link is full, a queue spill-back occurs and the amount of incoming agents is reduced [46]. More information about *MATSim* is available on the corresponding project webpage [39].

To conclude, it can be said that *MATSim* seems to be the best choice for the aim of the here presented research. On the one hand, it has already been demonstrated that *MATSim* can be used with OSM data [5] and on the other hand it provides many possibilities for adapting the simulations to individual requirements. Therefore, *MATSim* has been selected as the evacuation simulation framework for the here conducted research. That is, after introducing *IndoorOSM* in the next subsection, the remainder of this paper will discuss the possibilities of using crowdsourced indoor geodata from OSM for indoor evacuation simulations with *MATSim*.

3. Evacuation Simulations with *IndoorOSM*

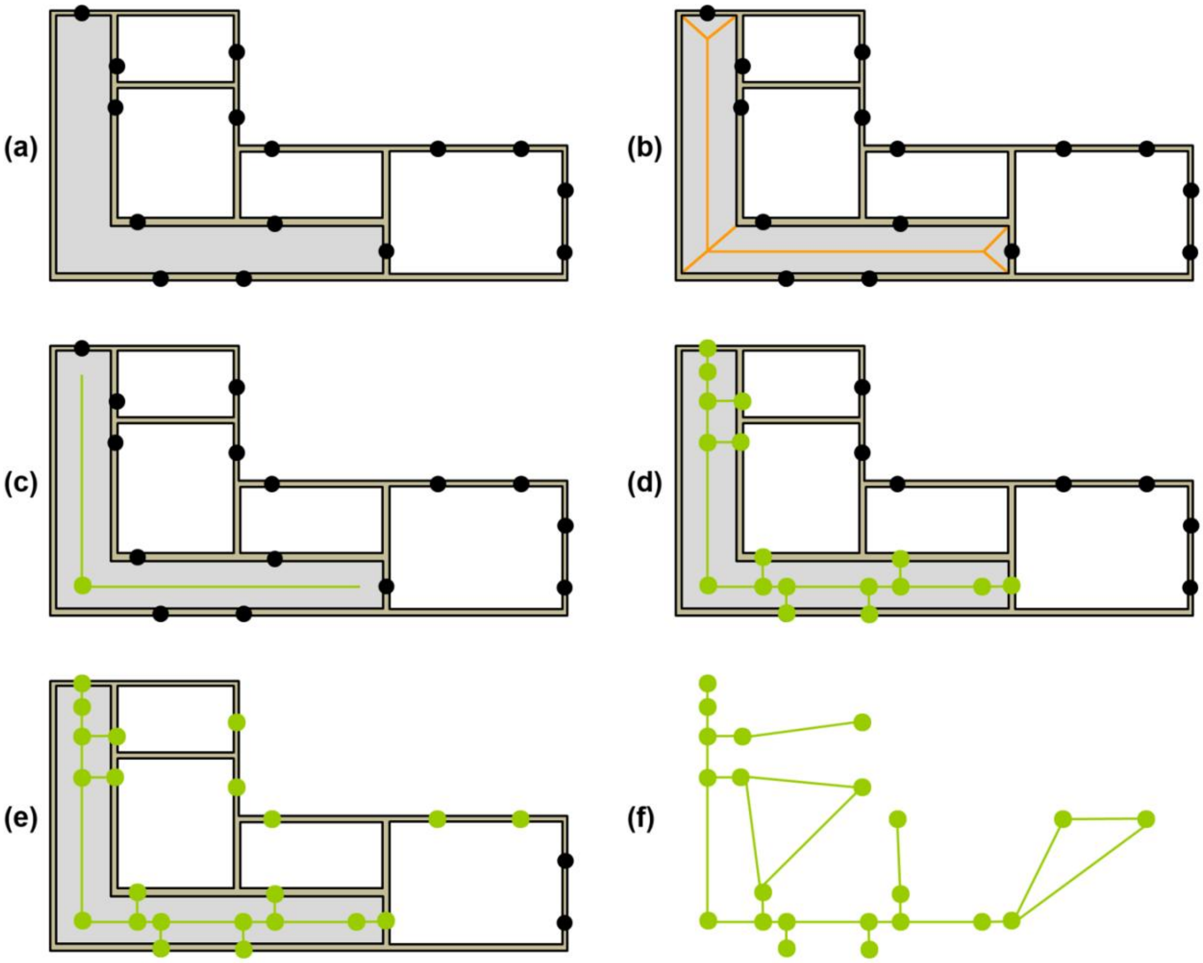
This section focusses on the possibilities of automatically deploying indoor evacuation simulations with *IndoorOSM* data. In particular, it will be discussed what kind of evacuation related information is available and how this can be used for the simulation. As described in the previous section, the framework *MATSim* has been selected for the here conducted research. Furthermore, the different required input files and parameters as well as necessary assumptions are elaborated.

3.1. Generating the Network

When computing (shortest) paths inside buildings, the indoor environment is typically represented with a routing graph which is capable for applying a shortest path algorithm (e.g., Dijkstra [47] or Hart [48]) to it. Thereby, nodes in the graph typically represent decision-points and links represent connections between different decision-points. In outdoor environments, a decision-point typically comprises a street crossing. However, since the work within this paper focusses on indoor traffic simulations, a decision-point represents indoor transitions, such as doors or intermediate turning points in a corridor. In principle, there are different graph models available, such as [49–52], which mainly vary in granularity and formalism. It has been demonstrated that some of those can be automatically extracted from official data sources, but the extraction of such graphs from VGI has not yet been discussed. In contrast, the so called *Weighted Indoor Routing Graph (WIRG)* [53]—representing a formally defined and reasoned graph model for length-optimal indoor routing—can be automatically generated by purely using crowdsourced indoor geodata from *IndoorOSM* [19]. That is, it is feasible to generate an indoor routing graph from *IndoorOSM*. However, it needs yet to be proven, that *IndoorOSM* data is also suitable for generating a graph which can be used for complex indoor evacuation simulations. In particular, such a graph should not only contain information about doors and rooms, but also about windows, because those are important for evacuation purposes. However, the selection of contemplable windows is not always an easy task, because it might require local knowledge about the building structure as well as the surrounding terrain, to define which windows can serve as emergency exits and which ones not. This problem will also be discussed in Section 5.1.

Figure 2(a–f) visualizes the general principle of the generation of a detailed indoor routing graph step-by-step. Thereby, not only doors are considered and integrated in the graph, but also the different windows of the rooms. All the required information is available in *IndoorOSM* and the graph can be automatically generated by purely using this data.

Figure 2. Stepwise generation of an indoor routing graph (according to the Weighted Indoor Routing Graph (WIRG) definition [53]) with a selection of contemplable windows for emergency evacuation simulations.



As an initial situation (Figure 2(a)), all building part footprints, *i.e.*, the polygonal shapes of all rooms or corridors of a distinct building floor, are mapped according to the *IndoorOSM* mapping proposal (*cf.* Section 2.1). For each corridor (those are explicitly tagged in *IndoorOSM* as *buildingpart=corridor*) the centerline is computed by generating the skeleton [54] of the underlying corridor polygon (orange line in Figure 2(b)) and pruning all lines of the skeleton which are connected to the outline of the corresponding corridor. The remainder constitutes the centerline of the corridor (green line and nodes in Figure 2(c)), which on the one hand have been proven to represent the geometry of the corridor very detailed and on the other hand also represents human behavior when walking along a corridor [53]. Thereafter, all doors (OSM key *door*) which are related to the corridor (those are either part of the corridor geometry or part of an adjacent building part) are added to the routing graph and vertically connected with the corresponding corridor (Figure 2(d)). Afterwards, contemplable windows, *i.e.*, those which are suitable for emergency evacuation (*cf.* also the discussion in Section 5.1) are added as nodes to the graph. Finally—as stated in the *WIRG* definition [53]—all nodes (*i.e.*, doors and windows) of a single room are connected pairwise with each other via an edge in the graph, resulting in the final routing graph (Figure 2(f)). Fore multi-level buildings, the steps (a) to (f) are repeated accordingly for every building floor. Furthermore, vertical connections, such as stairways or escalators, are included by evaluating the available data of IndoorOSM (e.g., the keys *buildingpart=verticalpassage*, *connector:ids etc.*, *cf.* Section 2.1).

In *MATSim*, the network inside the building is represented with a directed graph, *i.e.*, the graph contains two directed links rather than one non-directed link. The nodes are represented via a unique ID, as well as a two-dimensional coordinates (*i.e.*, x and y) from a bird's perspective. The height (*i.e.*, the z-value) however is not explicitly required. The different links are represented via a unique ID, information about the start (*from*) and end (*to*), the *length* of the link, the *freespeed* (*i.e.*, the maximum traveling speed) and the *capacity* (*i.e.*, the flow capacity in terms of how many agents can pass this link in a given time). Furthermore, all links contain information about the amount of permanent lanes (*permlanes*). Regarding outdoor simulations, this parameter is basically used for describing the number of car lanes for a distinct road. Considering indoor spaces (and especially corridors), this parameter represents the number of agents which are able to stand next to each other in a given space (e.g., corridor). *Weidmann* [55] defined the average agent's width as 0.71 m, so for example a corridor with a width of 2.13 m is represented as a link with three *permlanes*. In the *MATSim* specification, the value of *permlanes* is a double, thus values like *permlanes=1.46* are also possible. Important to mention is that the value of *capacity* is related to the value of the parameter *capperiod*. This parameter is globally defined for the whole simulation scenario and describes the time period for which the different *capacity* values are valid. That is, when *capperiod* is set to 00:01:00 (*i.e.*, one hour), the value of *capacity* is interpreted as flow capacity for one hour. As already described in Section 2.2, *MATSim* applies a cell-based queue model. Therefore, the cell-size needs to be globally defined for the simulation via the two parameters *effectivecellsize* and *effectivelanewidth*. For pedestrian simulation, reasonable values are 0.26 m and 0.71 m [55]. However, due to the static definition of these values, this causes some impreciseness. This issue will be further discussed in Section 5.2.

The *length* of the links is the distance between the two involved nodes in meters which can be easily populated by computing the Euclidean distance between the two nodes. The parameter *freespeed* defines the upper boundary for the maximum travel speed of an agent, thus the traveling speed of an agent in a free space. There are different research efforts and investigations on travel speeds of pedestrians in indoor spaces, typically also depending on the agents themselves as well as the investigated scenario. *Weidmann* [55] evaluated 52 different investigations on the traveling speed of pedestrians. He discovered a range between 0.97 m/s and 1.65 m/s, whereby most of the values are between 1.25 and 1.45 m/s. The general average traveling speed for pedestrians can be assumed to be 1.34 m/s [55]. Depending on the individual situation of the agent, these values can vary, so for example a 70 years old person is likely to achieve approximately 72% of this average speed [55]. Regarding movements on stairs, pedestrians are approximately 50% slower [55]. The amount of permanent lines also needs to be provided for each link. Thereby, the average agent's width of 0.71 m [55] can be utilized. For an automated population of this value, there are basically three possible approaches, as described in the equations below. Thereby, the variable *width* represents a list of all corridor widths in increasing order and *length* represents the length of the corridor.

$$permlanes = \frac{\min widths}{0.71} \quad (1)$$

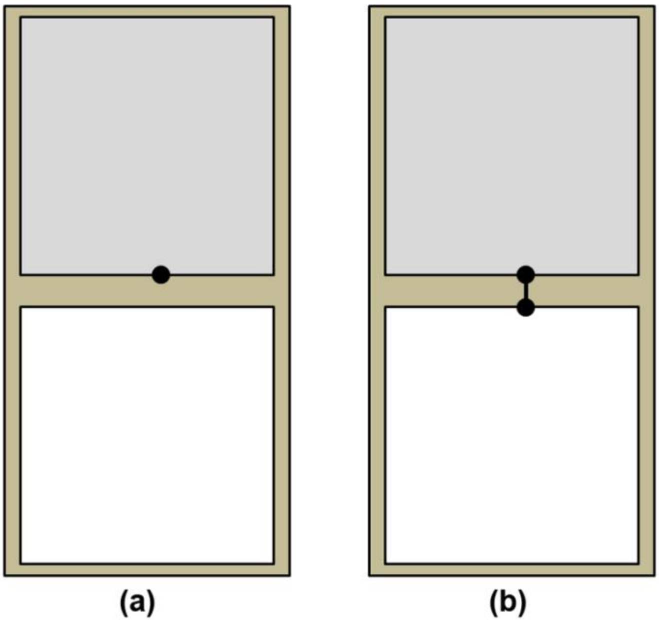
$$permlanes = \frac{\sum_{i=1}^{n} widths_i}{n * 0.71} \quad (2)$$

$$permlanes = \frac{\sum widths}{length * 0.71} \quad (3)$$

Equation (1) does only incorporate the minimum width, thus represents the most pessimistic parameter (worst case). In contrast, computation Equation (3) is the most optimistic approach (best case). Usually a worst case and a best case approach are used for demonstrating the bandwidth of the expected evacuation performance to the decision makers. The individual *capacity* of a link can be defined by considering the agent's average width, the traveling speed and the corridor width.

Since doors in indoor environments reduce the flow capacity of the mass (less people can simultaneously pass a one meter wide door than a three meter wide gateway), the incorporating of the size of a door or window (especially its width) is important. This information is explicitly mapped in OSM with the key *width* (as well as *height* or *breast*) and can therefore be used when generating routing graphs. As described in Section 2.1, in *IndoorOSM* a door or window is added to one of the two involved building part geometries (Figure 3(a)). Therefore, the single node (door or window) needs to be projected to the other involved geometry, resulting in an additional node in the graph (Figure 3(b)). Those two nodes are then furthermore connected via a (very short) edge, whereby the width is utilized for calculating the different link parameters, such as *permlanes* or *capacity*.

Figure 3. An OSM node (black circle) representing a door between to rooms (**a**) and the additional node (projected to the adjacent polygon) with an additional edge connecting them (**b**).



3.2. Generating the Synthetic Population

MATSim requires information about the different agents. This information is called population and is described in the *population.xml* file. Basically, a population consists of a (non-limited) amount of persons. Each person element contains a unique *id* (mandatory), as well as some other (optional) parameters, such as the agent's *sex* or *age*. Furthermore, each person has a *plan*, thus a sequence of at least one planed activity (*act*). Thereby, an activity represents the physical location (defined via the network link *id*) of the corresponding agent for some distinct time or interval. The movement of an agent within the network over time can be represented by adding several activities to an agent.

Currently, *IndoorOSM* does not consider population information for indoor spaces. This issue will be further discussed in Section 5.1. That is, when performing indoor evacuation simulations, there are basically two possibilities: (1) a manual creation of the population with real-world figures or (2) an automated creation with a random distribution of the agents to the different rooms (probably based on estimations according to room size or function). Depending on the amount of agents and the distinct scenario, both possibilities have their advantages and disadvantages. However, more realistic results can be achieved with manually added real world figures (if available).

3.3. Defining the Evacuation Area

Basically, the two input files mentioned beforehand would be enough for a *MATSim* traffic simulation. However, as the intention of this paper is the conduction of evacuation simulations, there is a third input file required: the *evacuationarea.xml* file. The sake of this input file is (as the name might indicate) to define which area of the scenario (the network) needs to be evacuated. In other words, the *evacuationarea.xml* file describes which parts of the network are safe and which are endangered (by some threat). The file structure itself is rather simple, as it only contains the *id*s of the endangered links in the network. In addition, it is possible to define a deadline for the individual links, *i.e.*, the point of time at which the link is accessible at the latest.

Basically, this file can be easily generated automatically from *IndoorOSM* data. Typically, an indoor evacuation aims at evacuating all inhabitants to the outside of the building. That is, the evacuation area consists of all links of the network which are inside the building. All other links (*i.e.*, the outdoor features of OSM) are not part of the evacuation area (at least for a basic indoor evacuation simulation). For incorporating some kind of safety area around the building (e.g., in the scenario of a fire), it is also possible to add all features (essentially streets or paths) to the evacuation area, which are within a distinct distance around the building.

4. Demonstration and Experimental Results

As a demonstration and proof-of-concept, two exemplary evacuation simulations have been performed. By doing this, several limitations of both the *IndoorOSM* data and the *MATSim* framework became apparent. Those will be discussed in Section 5. The two simulation scenarios are as follows: (1) a planned site clearing, *i.e.*, a structured, organized and safe evacuation as it is for example required in the case of a forecasted flood, and (2) an unpredicted evacuation simulation for a sudden disastrous incident. The main difference between those two scenarios is that in the first scenario all persons are safely evacuated trough the main building exits. In contrast, in the latter scenario all possible (emergency) exits are considered. Essentially, also windows, garage doors *etc*., can serve as an exit for the affected population in Scenario 2. For simplification purposes, in both scenarios an average travelling speed of 1.0 m/s has been defined. This seems to be an adequate (simplified) average value which incorporates different agent's conditions, different movements (*i.e.*, plane *vs.* stairs), and different stress levels. For simplification, a distinction between different kinds of agents has not been performed. That is, all agents have the same average travelling speed, as well as the same sex or age. As another simplification, in both scenarios it is assumed that all agents start their movement instantly at the same time. In particular, no individual reaction time has been included. In reality this behavior is rather unlikely, because people have different reaction times as well as behavior, e.g., some will instantly escape the building, others will get their belongings and then leave the building, and probably some will not even notice the emergency alert. It needs to be mentioned that all these simplifications are made because of the focus of this paper. The main aim of this paper is to demonstrate and discuss the utilization of *IndoorOSM* for evacuation simulations, rather than performing highly realistic simulations for a distinct building and discussing those afterwards.

As a test case, the building of the GIScience research group of the University of Heidelberg has been utilized. It is an averaged sized university building consisting of one basement and three floors above the ground. Figure 4(a) depicts a 3D model of the front side of the building with the main entrance, whereas Figure 4(b) shows the building back side with the basement windows and garage doors (they can serve as an emergency exit). Inside the building, the different floors are connected via two staircases with each other. The building contains one big lecture room with a capacity of 120 persons, one smaller lecture room for 30 persons and two computer pools with capacities of 25 and 16 persons. Furthermore, there are several offices which are occupied by one to three persons. For both evacuation simulation scenarios it is assumed that the building is fully occupied, which leads to a total amount of 313 inhabitants in the building. Table 1 contains the distribution of those to the different floors. For both scenarios, a *MATSim* evacuation simulation has been performed. All required input files have been automatically generated from *IndoorOSM* data.

Figure 4. A 3D model of the use case building: the front side with the main entrance (**a**) and the back side with the garage doors and basement windows (**b**).

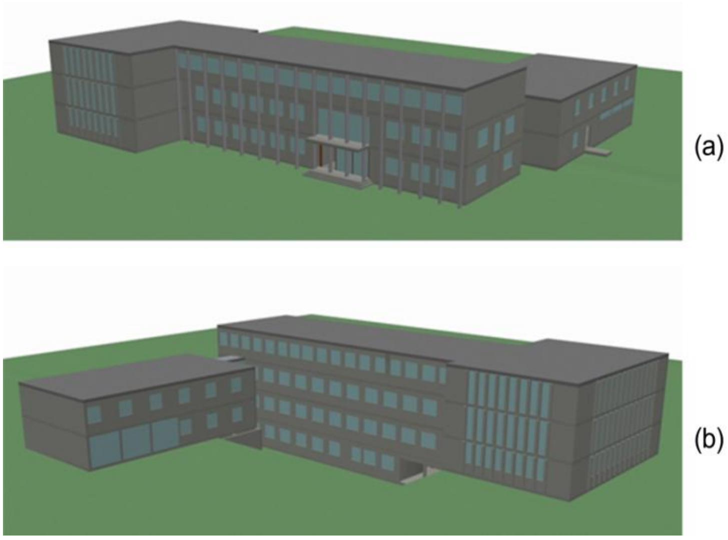


Table 1. Aggregated population distribution for the different building floors.

Building Floor	**Population**
Basement (Floor −1)	*24*
Ground (Floor 0)	*178*
Floor 1	*74*
Floor 2	*37*
Total	**313**

4.1. Scenario 1: Planned Site Clearing

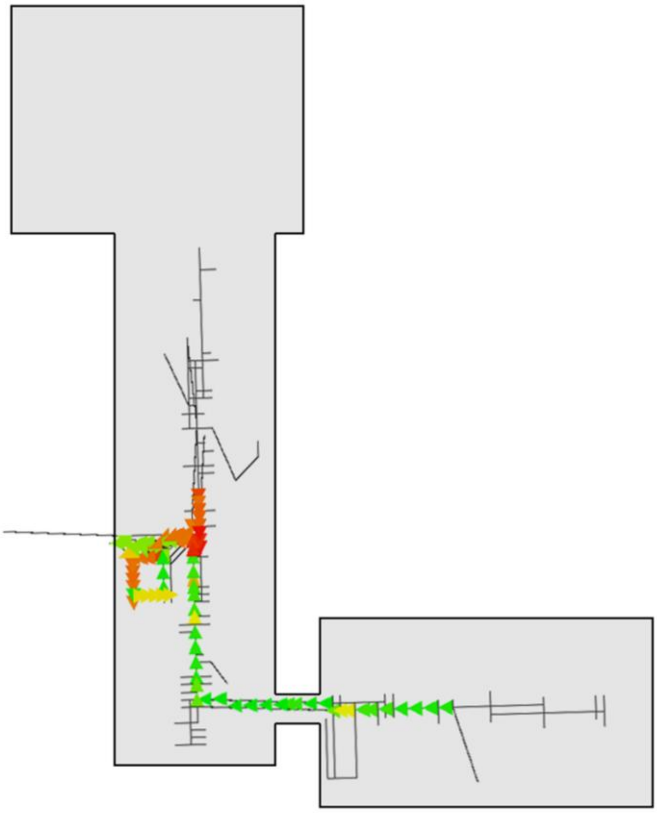
The first evacuation scenario represents a planned site clearing. Since all agents should leave the building through the main entrance, the *evacuationarea.xml* file contains all links except the link from the main entrance to the outside of the building. A total amount of 100 simulations has been performed, because *MATSim* agents can learn from previous simulations, which influences their behavior in future simulation iterations. Regarding all iterations, the average travel distance is 39.55 m for the executed plan, 39.62 m for the worst plan (*i.e.*, the longest overall evacuation time for the complete building), 39.52 m for the average plan and 39.46 m for the best plan (*i.e.*, the shortest overall evacuation time). By investigating the different iterations in more detail, the learning mechanisms of *MATSim* become apparent. One example is that in the very first iterations all students from the big lecture room (which is in the ground floor) use the direct path to the exit of the building. This results in a huge traffic jam in the corridors. In contrast, in the very last iterations, some agents avoid those traffic jams by directly walking to the first floor (via the staircase next to the lecture room), traversing the corridor and then returning to the ground floor via the other staircase (which is directly adjacent to the exit). However, although this learning leads to shorter evacuation times (*i.e.*, the time required to reach a safe place) for some individual agents, the overall scenario does hardly change (*cf.* Table 2).

Table 2. Evacuation time statistics: Scenario 1.

Figure	**Iteration 1**	**Iteration 50**	**Iteration 100**	**Average**
Evacuation time for first agent	*7.1 s*	*7.1 s*	*7.1 s*	*7.1 s*
Evacuation time for last agent (i.e., complete building)	*166.2 s*	*166.0 s*	*165.8*	*166.0 s*
Average trip duration for one agent	*80.53 s*	*80.82 s*	*80.86*	*80.85 s*

Figure 5 depicts the evacuation simulation scenario in an early stage (after 40 s). Thereby the underlying network as well as the building outline is visualized. The different colors of the agents indicate their current situation. Green indicates a free movement at maximum speed, orange indicates a slowdown in the traffic flow and red indicates a traffic jam in the corresponding building part. In Figure 5 it can be seen that the staircase next to the building exit is a bottleneck because the agents in this area have to reduce their traveling speed due to a traffic jam.

Figure 5. Visualization of the evacuation simulation Scenario 1 in its early progress after 40 s.



4.2. Scenario 2: Unpredicted Evacuation

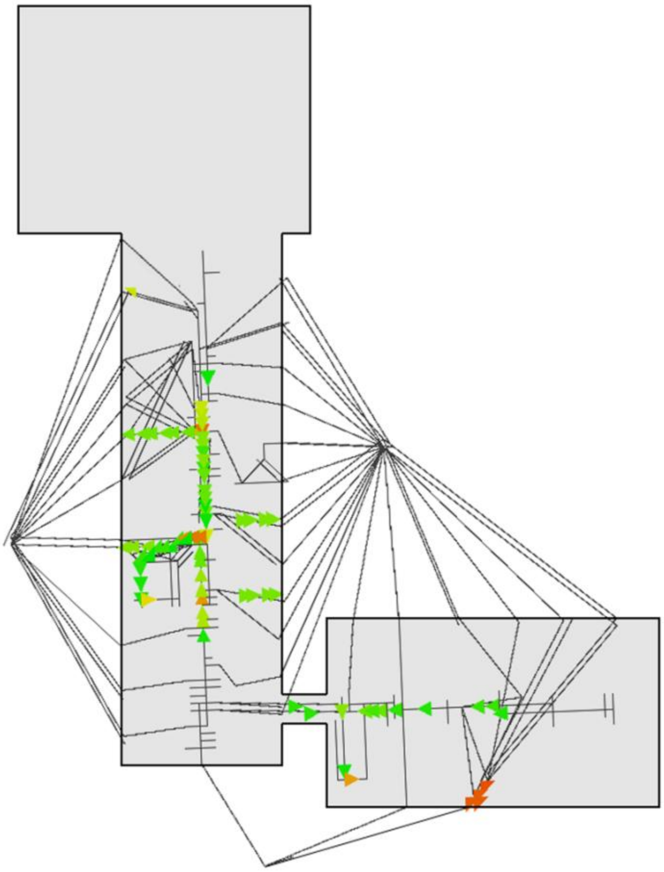
The second evacuation scenario represents an unpredicted evacuation. Those are typically required in the case of an unforeseeable incident, such as a fire or an earthquake. In this scenario, all agents instantly try to leave the building as fast as possible. For simplification purposes, it is again assumed that all agents start their movement simultaneously at the beginning of the simulation. Contrary to the previous simulation, Scenario 2 incorporates all possible exits. Essentially, all windows of the basement and ground floor and all (garage) doors are considered as emergency exits. Therefore, the network also contains links between rooms and the corresponding windows. Furthermore, all windows are connected via links to the safe outdoor environment. The *evacuationarea.xml* file contains only the links which are inside the building.

Regarding all 100 iterations, the average travel distance is 22.86 m for the executed plan, 23.06 m for the worst plan, 22.85 m for the average plan and 22.74 m for the best plan. Similar to Scenario 1, Table 3 provides evacuation time statistics for evacuation simulation Scenario 2. Again, the evacuation time is defined as the time between the start of the simulation and the time till when the corresponding agent reaches a safe place. The evacuation time for the first agent of 2.1 s results from the assumption of an immediate reaction of the individual agents. Some agents in the basement as well as in the ground floor can use the windows as emergency exits. This leads to a reduced average evacuation time and traveling distance (compared to Scenario 1). Figure 6 depicts the evacuation simulation Scenario 2 after a period of 10 s; therein, the main difference is the network which also incorporates the appropriate windows as emergency exits.

Table 3. Evacuation time statistics: Scenario 2.

Figure	Iteration 1	Iteration 50	Iteration 100	Average
Evacuation time for first agent	*2.1 s*	*2.1 s*	*2.1 s*	*2.1 s*
Evacuation time for last agent (i.e., complete building)	*106.1 s*	*106.0 s*	*105.8 s*	*105.9 s*
Average trip duration	*30.04 s*	*30.03 s*	*29.99 s*	*30.03 s*

Figure 6. Visualization of the evacuation simulation Scenario 2 in its early progress after 10 s.



5. Discussion

While investigating the possibilities for an automatic generating of indoor evacuation simulations and performing the two exemplary use-case simulation scenarios, different limitations and issues regarding both the available data in *IndoorOSM* and the *MATSim* simulation framework became apparent. Those will be discussed in the two following sub-sections. Section 5.1 focusses on the suitability of crowdsourced indoor geodata for evacuation simulations. Thereafter, Section 5.2 discusses several limitations of the *MATSim* simulation framework for indoor evacuation simulations.

5.1. Limitations of the IndoorOSM Data

Crowdsourced indoor geodata from OSM contains very detailed information about the interior structure of a building. In the previous section it has been furthermore demonstrated that it is possible to use this data for different evacuation simulation scenarios. Nevertheless, while conducting the here presented research, several limitations and constraints became apparent. In general, *IndoorOSM* contains detailed information about the geometry and topology of each individual building floor. That is, detailed floor plans, such as evacuation plans, as well as routing graphs which are suitable for individual indoor routing as well as indoor mass evacuation simulations, can be generated based upon this data source. For investigating the evacuation performance of a future building, *i.e.*, a scenario for which no real world figures are available, this approach leads to satisfying simulation results which provide a first indicator on the safety of the building. Nevertheless, for more realistic results and especially for simulating the performance of existing buildings for which real world data exists, *IndoorOSM* lacks various kinds of information.

As already discussed in Section 3.2, real population figures are currently not available in *IndoorOSM*. That is, *IndoorOSM* is suitable for generating the underlying network, but a (either real or synthetic) population cannot be generated by only using *IndoorOSM* data. Due to the open key-value pair methodology of OSM, one could argue that an appropriate key (e.g., *population=2*, *etc.*) could be easily defined and utilized for this effort; however it is questionable if such a key is suitable for describing (dynamic) population figures, because population numbers typically vary with changing date, time, or function of the room. For example a lecture room with tables can accommodate less students than an emptied out lecture room (e.g., in the case of a ceremony). Another example is that there are more humans in an indoor swimming pool in the winter season than in the summer season. That is, when introducing such a key in OSM, it still is not clear for which time of day or season it is valid. Furthermore, the population might change over time, for example due to people moving from one office to another, and it is questionable if such (fast) changing information is maintained properly in OSM (in contrast to the geometry of a building which is rather static). Nevertheless, a potential key *population* might provide an indicator on the maximum capacity of a room, such as an office is suitable for accommodating four permanent employees or a lecture room can accommodate 100 students. As a conclusion, such maximum capacity information could then be used for worst-case simulations which furthermore provide an insight on the worst-case evacuation performance of a (future) building. Nevertheless, for gathering more realistic and probable results—especially for existing buildings—the additional incorporation of other available data sources, such as linked geo data [56], real-time data (so called *Live Cities*) [57], or facility management systems, will probably lead to more realistic simulation results.

Closely related to this issue, *IndoorOSM* (and probably also other data sources) currently lacks more precise information about the individual inhabitants. That is, even if OSM contributors provide population information, more details on the individual persons, such as sex, age, health situation *etc.*, are not available. That is, while performing the evacuation simulation, one still have to assign random values and parameters to the individual agents, manually assign real world figures, or (if only rough results are required) perform a simulation with average values and consider all agents as being physically equal. *IndoorOSM* currently cannot—and very likely will never be able to—provide any precise details on the different inhabitants of a building. Drawing conclusions on the (average) condition of the inhabitants of a building according to the building function is probably the only solution when only using OSM data. For example if a building is tagged with *amenity=university*, one could argue that most of the inhabitants are students, thus healthy and vital, whereas in contrast inhabitants of a building tagged with *amenity=hospital* are probably not that vital and limited regarding their movement and traveling speed.

Depending on the layout of a building and the corresponding evacuation scenario, windows can also be considered as possible emergency exits. However, the selection of windows which can serve as emergency exits is not an easy task. Although information about the location (x, y and z) as well as the height and width of a window is basically available in *IndoorOSM*, selecting appropriate emergency windows is not always easily feasible because both the surrounding terrain as well as the building layout need to be included in the selection process. To some extent, it can be assumed that all windows of the ground floor could potentially serve as emergency window; however it might be quite hard to define which floor the ground floor is, as for example in the case of a hillside building. It might also be possible that a window in the second floor can—due to a sloped terrain—still be used as an emergency exit. Also, if a building is situated directly next to a lake, it might be possible to jump from the roof into the lake. In contrast, in a building which is located next to a canyon, the windows which are faced towards the canyon cannot serve as an emergency exit—even on the ground level. Also, for some building structures it is possible to emerge a building by climbing out of the window on the second floor to the roof of the first floor and then climbing down to the ground. However, *IndoorOSM* currently lacks additional information describing the individual emergency suitability of a window, although this kind of information probably improves the simulation results. Nevertheless, by introducing a new tag, e.g., *emergency:exit=yes*, which is added to the corresponding window node, OSM contributors can easily and clearly define which windows can serve as an emergency exit and which ones not. That is, in principle this kind of information can be added to OSM and then be utilized while selecting appropriate windows for one's own individual evacuation simulation.

Another important aspect which has been hardly considered yet in *IndoorOSM* is the incorporation of barriers and obstacles inside buildings. These can be either static objects, such as furniture or struts, but also dynamic objects, such as parts of the ceiling falling down. Both kinds influence the mass behavior inside the building and therefore also the execution and performance of the evacuation. The former kind of obstacle is not yet available in *IndoorOSM*. Nevertheless, contributors can provide a closed way representing the outline of the obstacle and tag it with an appropriate tag, e.g., *obstacle=table* or *obstacle=cupboard*. That is, a contributor who really cares about detailed indoor environments can basically provide information about obstacles, but such information is not yet available in the current OSM dataset. In contrast, information about dynamic and moving objects cannot be provided, because every object within OSM can only be represented for one single distinct point of time and dynamics cannot be expressed with the current OSM data model. That is, such (either live or synthetic) data needs to be generated manually, for example by calculating and modeling the diffusion of dense smoke, or by integrating live sensors, for example by using an Open Geospatial (OGC) Sensor Observation Service (SOS) [58]. The latter approach is similar to the before mentioned idea of integrating real-time data from so called *Live Cities* (*cf*. above).

Closely related to this issue, *IndoorOSM* does also only provide limited information about the height of the individual building parts (e.g., rooms or corridors). Although *IndoorOSM* proposes the key *height* for providing information about the height of a room, it is not always clear for which part of a room this height is valid. Essentially, a ceiling with an inclined surface (which cannot be represented or described in *IndoorOSM*) might have varying heights, or drooping objects, such as lamps, probably reduce the effective height of a room. Furthermore, the effective height might be reduced due to some furniture or obstacle (*cf*. above). Nevertheless, depending on the evacuation scenario, such information needs to be integrated, because potential evacuation routes might be affected by such circumstances, as for example a wheelchair driver cannot pass obstacles lying on the floor.

As a conclusion of this section, it seems apparent that crowdsourced indoor geodata from OSM contains detailed information about the static features of a building, but currently lacks both dynamic aspects as well as population information. Furthermore, it can be stated that such information probably will be never integrated in OSM That is, evacuation simulations which are purely based on OSM data and especially do not integrate additional (synthetic or real) data, will always result in coarse simulations. However, crowdsourced indoor geodata seems to be a rich additional data source which provides detailed information about the geometry and topology of a building. That is, *IndoorOSM* can be basically used for generating any kind of static component of an indoor evacuation simulation.

5.2. Limitations of the MATSim Simulation Framework

It has been proven that *MATSim* is suitable for not only outdoor simulations, but also indoor evacuation simulations. However, it needs to be mentioned that there are some limitations of *MATSim* and essentially its queue model for the conduction of indoor simulations. Since the network within *MATSim* is directed and corridors in indoor environments are typically not limited to a distinct traveling direction (except for some special cases, such as security controls or escalators), the network of *MATSim* typically contains two directed links with the same amount of *permlanes* and the same *capacity*. However, the amount of *permlanes* and the *capacity* of one link strongly depend on the current traffic situation on the opposite link. For example, the capacity of a corridor where agents only travel in one direction is different to the capacity in the case of agents traveling in both directions. For an appropriate representation of this effect, a dynamic adaptation of both *capacity* and *permlanes* for the affected links would be required. However, this is (at the moment) not feasible with *MATSim*. That is, for the simulation of normal traffic flows in a building, *MATSim* is not suitable. Nevertheless, for simulation evacuation scenarios in which all agents move to the outside of the building, *MATSim* is an appropriate simulation framework. Also, the agents of *MATSim* are not clever. Since they are tied to the simulation network, their movement is rather linear. Furthermore, they are restricted to the direction of the network, thus they cannot change the travel direction while traversing a link. This behavior perfectly suits to car traffic simulation, but brings the before mentioned limitations when simulation pedestrian behavior. That is, a real two-dimensional movement space for the agent would probably lead to different (better) simulation results.

Another kind of dynamic data which cannot be represented in *MATSim* is the consideration of different movement types of an agent, such as sauntering, walking or running. As discussed in the previous sections, the covered space of a single agent is globally defined for all agents. However, it can be argued that these values vary with the actual movement type of the agent, e.g., a running person probably requires more space (especially in the direction of the movement) than a sauntering person. It can be assumed that different agents (depending on their health condition as well as stress level) prefer different movement types. That is, a dynamic adaptation of the covered space for arbitrary agents would probably lead to more realistic results.

6. Conclusion and Future Work

This paper proposed the utilization of crowdsourced indoor geodata from OSM (*IndoorOSM*) for indoor evacuation simulation. It can be concluded that it is basically feasible to perform multi-agent evacuation simulations with *MATSim* based on *IndoorOSM* data. That is, not only simple route planning applications can be created with crowdsourced indoor geodata, but also more complex and advanced applications, such as evacuation simulations, can benefit from *IndoorOSM*. However, as discussed in the previous section, there are some limitations and restrictions regarding both the *IndoorOSM* data and the *MATSim* simulation framework. Essentially, *IndoorOSM* is only suitable for the generation of static scenario components, such as the geometry and topology of a building interior. Any kind of dynamic factor, such as moving objects or gas emission, cannot be mapped in *IndoorOSM*. That is, such details cannot be populated or simulated with crowdsourced indoor data from OSM. Furthermore, detailed information about the real population of a building can hardly be provided via *IndoorOSM*. Due to the queue model, *MATSim* is not able to simulate real 2D movement. Furthermore, *MATSim* cannot simulate different movement spaces for different kinds of movements. Regarding the conducted experimental simulation, it needs to be stated that various simplifications have been made which are suitable for the purpose of the paper, but influence the simulation results.

Since crowdsourced data is typically created by non-professional users and due to the open-access paradigm of OSM, the available data is always subject to errors as well as vandalism—this also comprises one of the major counter-arguments for third parties when evaluating the potential usage of OSM data. Nevertheless, for the road network it has already been proven that the crowd is able to collect fine-grained data which is comparable to commercially collected data [8]. Quality concerns are even more crucial for detailed indoor spaces, because indoor environments are typically more fine-grained than outdoor spaces, especially when conducting complex simulations, as for example discussed in this paper. However, due to its novelty, there is not yet enough data available for conducting comprehensive quality analysis or comparing crowdsourced indoor data to commercially collected data. Nevertheless, those will be required as soon as more data is available, for supporting the importance of crowdsourced indoor data. For future work it will be interesting to see if *IndoorOSM* data can be used for indoor evacuation simulations which are not based on a graph, but on polygonal structures (*i.e.*, real two-dimensional movement of the agents). A combination of both indoor and outdoor evacuation simulation based on OSM data is a desirable, but challenging task. As discussed, *IndoorOSM* has some limitations and probably not every building structure can be mapped in OSM. To overcome this issue, researchers intend crowdsourcing virtual 3D cities and especially 3D building models [59], which can potentially be utilized for indoor evacuation simulations. However, since this *OpenBuildingModels* [59] approach currently is still in its infancy and there are not yet so many buildings available, this has yet to be brought into fruition.

Acknowledgments

The authors of this paper would like to express their thankfulness to the anonymous reviewers. By providing their valuable comments on this paper they contributed towards the improvement of our work. Additionally we would like to thank Gregor Lämmel for his support and help towards understanding the *MATSim* framework specification and its individual components. Furthermore, the authors would also like to thank their intern Daniel Söder for creating the 3D model of the use case building. This research has been partially funded by the Klaus-Tschira Foundation (KTS) Heidelberg. The visualization of the simulation results has been realized with the application Senozon Via.

References

1. Schmitz, S.; Zipf, A.; Neis, P. Proposal to Define Common Resources for OpenGIS Location Services. In *Proceedings of 5th International Symposium on LBS & TeleCartography*, Salzburg, Austria, 26–28 November 2008.
2. Neis, P.; Zipf, A. OpenRouteService.org Is Three Times "Open": Combining OpenSource, OpenLS and OpenStreetMaps. In *Proceedings of the GISRUK 2008 Conference*, Manchester, UK, 2–4 April 2008.
3. Neis, P.; Singler, P.; Zipf, A. Collaborative Mapping and Emergency Routing for Disaster Logistics—Case Studies from the Haiti Earthquake and the UN Portal for Afrika. In *Proceedings of Geospatial Crossroads @ GI_Forum '10*, Salzburg, Austria, 6–9 July 2010.
4. Dallmeyer, J.; Lattner, A.D.; Timm, I.J. From GIS to Mixed Traffic Simulation in Urban Scenarios. In *Proceedings of 4th International ICST Conference on Simulation Tools and Techniques*, Barcelona, Spain, 21–25 March 2011; pp. 134–143.
5. Zilske, M.; Neumann, A.; Nagel, K. OpenStreetMap For Traffic Simulation. In *Proceedings of State of the Map Europe 2011 (SOTM-EU)*, Vienna, Austria, 15–17 July 2011; pp. 126–134.
6. Zielstra, D.; Zipf, A. A Comparative Study of Proprietary Geodata and Volunteered Geographic Information for Germany. In *Proceedings of 13th AGILE International Conference on Geographic Information Science*, Guimarães, Portugal, 10–14 May 2010; pp. 1–15.
7. Haklay, M. How good is volunteered geographical information? A comparative study of OpenStreetMap and Ordnance Survey datasets. *Environ. Plan. B* **2010**, *37*, 682–703.
8. Neis, P.; Zielstra, D.; Zipf, A. The street network evolution of crowdsourced maps: OpenStreetMap in Germany 2007–2011. *Future Internet* **2012**, *4*, 1–21.
9. Winter, S. Indoor spatial information. *Int. J. 3-D Inf. Model.* **2012**, *1*, 25–42.
10. Jensen, C.; Li, K.-J.; Winter, S. The other 87%: A report on The Second International Workshop on Indoor Spatial Awareness (San Jose, California–November 2, 2010). *ACM Sigspatial Newsletter* **2011**, *3*, 10–12.
11. Goetz, M.; Zipf, A. Extending OpenStreetMap to Indoor Environments: Bringing Volunteered Geographic Information to the Next Level. In *Proceedings of Urban and Regional Data Management: UDMS Annual 2011*, Delft, Netherlands, 28–30 September 2011; pp. 47–58.
12. OSM IndoorOSM. Available online: http://wiki.openstreetmap.org/wiki/IndoorOSM (accessed on 10 May 2012).
13. Goetz, M. Towards generating highly detailed 3D CityGML models from OpenStreetMap. *Int. J. Geog. Inf. Sci.* **2012**, accepted.

14. Zipf, A.; Basanow, J.; Neis, P.; Neubauer, S.; Schilling, A. Towards 3D Spatial Data Infrastructures (3D-SDI) based on Open Standards—Experiences, Results and Future Issues. In *Advances in 3D Geoinformation Systems, Lecture Notes in Geoinformation and Cartography*; van Oosterom, P., Zlatanova, S., Penninga, F., Fendel, E.M., Eds.; Springer: Heidelberg, Germany, 2007; pp. 65–86.
15. Kolbe, T.H. Representing and Exchanging 3D City Models with CityGML. In *3D GEO-Information Sciences LNG&C Part I*; Lee, J., Zlatanova, S., Eds.; Springer: Heidelberg, Germany, 2009; pp. 15–31.
16. Kolbe, T.H.; Gröger, G.; Plümer, L. CityGML: 3D City Models and their Potential for Emergency Response. In *Geospatial Information Technology for Emergency Response*; Zlatanova, S., Li, J., Eds.; Taylor & Francis: London, UK, 2008; pp. 257–274.
17. Goetz, M.; Zipf, A. Indoor Route Planning with Volunteered Geographic Information on a (Mobile) Web-based Platform. In *Proceedings of 9th Symposium on Location Based Services*, Munich, Germany, 16–18 October 2012; p. 16 (accepted).
18. Goetz, M. IndoorOSM: Mapping the Indoor World. Available online: http://indoorosm.uni-hd.de/ (accessed on 10 May 2012).
19. Goetz, M. Using Crowdsourced Indoor Geodata for the Creation of a Three-dimensional Indoor Routing Web Application. *Future Internet* **2012**, *4*, 575–591.
20. Shi, J.; Ren, A.; Chen, C. Agent-based evacuation model of large public buildings under fire conditions. *Automat. Constr.* **2008**, *18*, 338–347.
21. Okaya, M.; Yotsukura, S.; Sato, K.; Takahashi, T. Agent Evacuation Simulation Using a Hybrid Network and Free Space Models. In *Proceedings of 12th International Conference on Principles of Practice in Multi-Agent Systems*, Nagoya, Japan, 14–16 December 2009; pp. 563–570.
22. Goodchild, M.F. Citizens as voluntary sensors: Spatial data infrastructure in the world of Web 2.0. *Int. J. Spat. Data Infrastruct. Res.* **2007**, *2*, 24–32.
23. OSM Map Features. Available online: http://wiki.openstreetmap.org/wiki/Map_Features (accessed on 23 July 2012).
24. Tagwatch Tagwatch Planet-Latest. Available online: http://tagwatch.stoecker.eu/Planet-latest/En/tags.html (accessed on 10 May 2012).
25. OSM OpenStreetMap Wiki. Available online: http://wiki.openstreetmap.org/ (accessed on 10 May 2012).
26. OSM Indoor Mapping: OpenStreetMap Wiki. Available online: http://wiki.openstreetmap.org/wiki/Indoor (accessed on 10 May 2012).
27. Peacock, R.D.; Kuligowski, E.D. Workshop on building occupant movement during fire emergencies. *NIST Special Publication* **2005**, *1032*, 105.
28. Hajibabai, L.; Delavar, M.R.; Malek, M.R.; Frank, A.U. Agent-Based Simulation of Spatial Cognition and Wayfinding in Building Fire Emergency Evacuation. In *Geomatics Solutions for Disaster Management Lecture Notes in Geoinformation and Cartography*; Li, J., Zlatanova, S., Fabbri, A.G., Eds.; Springer: Berlin, Germany, 2007; pp. 255–270.
29. Yamashita, T.; Soeda, S.; Noda, I. Evacuation Planning Assist System with Network Model-Based Pedestrian Simulator. In *Proceedings of 12th International Conference on Principles of Practice in Multi-Agent Systems*, Nagoya, Japan, 14–16 December 2009; pp. 649–656.

30. Wu, L.; Lin, H. A personalized spatial cognitive road network for agent-based modeling of pedestrian evacuation simulation: A case study in Hong Kong. *Ann. GIS* **2012**, *18*, 109–119.
31. Gwynne, S.; Galea, E.R.; Owen, M.; Lawrence, P.J.; Filippidis, L. A review of the methodologies used in the computer simulation of evacuation from the built environment. *Build. Environ.* **1998**, *34*, 741–749.
32. Schreckenberg, M. *Pedestrian and Evacuation Dynamics*; Springer: Berlin, Germany, 2001; p. 452.
33. Schreckenberg, M.; Sharma, S.D. *Pedestrian and Evacuation Dynamics*; Springer: Berlin, Germany, 2002; p. 452.
34. Galea, E.R. *Pedestrian and Evacuation Dynamics*; CMS Press: Greenwich, UK, 2003; p. 411.
35. Waldau, N.; Gattermann, P.; Knoflacher, H.; Schreckenberg, M. *Pedestrian and Evacuation Dynamics*; Springer: Berlin, Germany, 2006; p. 495.
36. Klingsch, W.W.F.; Rogsch, C.; Schadschneider, A.; Schreckenberg, M. *Pedestrian and Evacuation Dynamics 2008*; Springer: Berlin, Germany, 2010; p. 847.
37. Peacock, R.D.; Kuligowski, E.D. *Pedestrian and Evacuation Dynamics*; Springer: Berlin, Germany, 2011; p. 910.
38. Balmer, M.; Rieser, M.; Meister, K.; Charypar, D.; Lefebvre, N.; Nagel, K. MATSim-T: Architecture and Simulation Times. In *Multi-Agent Systems for Traffic and Transportation Engineering*; Bazzan, A.L.C., Klugl, F., Eds.; Idea Group Reference: Hershey, PA, USA, 2009; pp. 57–78.
39. MATSim MATSim: Multi-Agent Transport Simulation Toolkit. Available online: http://www.matsim.org/ (accessed on 12 May 2012)
40. Meister, K.; Balmer, M.; Ciari, F.; Horni, A.; Rieser, M.; Waraich, R.A.; Axhausen, K.W. Large-Scale Agent-Based Travel Demand Optimization Applied to Switzerland, Including Mode Choice. In *Proceedings of 12th World Conference on Transportation Research*, Lisbon, Portugal, 11–15 July 2010; p. 30.
41. Bekhor, S.; Dobler, C.; Axhausen, K.W. Integration of Activity-Based with Agent-Based Models: an Example from the Tel Aviv Model and MATSim. In *Proceedings of 90th Annual Meeting of the Transportation Research Board*, Washington, DC, USA, 23–27 January 2011; p. 16.
42. Bakillah, M.; Klüpfel, H.; Lämmel, G.; Zipf, A. Agent Based Simulation for Disaster Preparedness. In *Proceedings of 6th International Conference on Pedestrian and Evacuation Dynamics (PED 2012),* Zurich, Switzerland, 6–8 June 2012.
43. Bakillah, M.; Klüpfel, H.; Lämmel, G.; Zipf, A. Toward a Generic Data Model for Multi-Agents Evacuation Simulation in Disaster Management Context. In *Proceedings of 2nd International Conference on Evacuation Modeling and Management*, Chicago, IL, USA, 13–15 August 2012.
44. Lämmel, G.; Klüpfel, H.; Nagel, K. The MATSim Network Flow Model for Traffic Simulation Adapted to Large-Scale Emergency Egress and an Application to the Evacuation of the Indonesian City of Padang in Case of a Tsunami Warning. In *Pedestrian Behavior: Models, Data Collection and Applications*; Timmermans, H., Ed.; Emerald Group Publishing Limited: Bingley, UK, 2009; pp. 245–264.
45. Gao, W.; Balmer, M.; Miller, E.J. Comparisons between MATSim and EMME/2 on the Greater Toronto and Hamilton Area Network. *Transp. Res. Record* **2010**, *2197*, 118–128.

46. Cetin, N.; Nagel, K. Parallel Queue Model Approach to Traffic Microsimulations. In *Proceedings of 82th Annual Meeting of the Transportation Research Board,* Washington, DC, USA, 12–16 January 2003; p. 10.
47. Dijkstra, E.W. A note on two problems in connexion with graphs *Numer. Math.* **1959**, *1*, 267–271.
48. Hart, P.E.; Nilsson, N.J.; Raphael, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE T. Syst. Sci. Cyb.* **1968**, *4*, 100–107.
49. Lorenz, B.; Ohlbach, H.J.; Stoffel, E.P. A Hybrid Spatial Model for Representing Indoor Environments. In *Proceedings of 6th International Symposium on Web and Wireless Geographical Information Systems (W2GIS 2006),* Hong Kong, China, 4–5 December 2006; pp. 102–112.
50. Zlatanova, S. SII for Emergency Response: The 3D Challenges. In *Proceedings of XXI ISPRS Congress,* , Beijing, China, 3–11 July 2008; Part B4-TYC IV, pp. 1631–1637.
51. Lee, J. A spatial access oriented implementation of a topological data model for 3D urban entities. *GeoInformatica* **2004**, *8*, 235–262.
52. Yuan, W.; Schneider, M. iNav: An Indoor Navigation Model Supporting Length-Dependent Optimal Routing. In *Proceedings of 13th AGILE International Conference on Geographic Information Science*, Guimarães, Portugal, 10–14 May 2010.
53. Goetz, M.; Zipf, A. Formal definition of a user-adaptive and length-optimal routing graph for complex indoor environments. *Geo-Spat. Inf. Sci.* **2011**, *14*, 119–128.
54. Felkel, P.; Obdrmalek, S. Straight Skeleton Implementation. In *Proceedings of 14th Spring Conference on Computer Graphics*, Budmerice, Slovakia, 1998; pp 210–218.
55. Weidmann, U. *Transporttechnik der Fussgänger, Transporttechnische Eigenschaften des Fussgängerverkehrs (Literturauswertung)*; Institut für Verkehrsplanung, Transporttechnik, Strassen- und Eisenbahnbau (IVG) ETH Zurich: Zurich, Switzerland 1993.
56. Auer, S.; Lehmann, J.; Hellmann, S. LinkedGeoData: Adding a Spatial Dimension to the Web of Dat. In *Proceedings of 8th International Semantic Web Conference (ISWC 2009),* Washington, DC, USA, 25–29 October 2009; pp. 731–746.
57. Resch, B.; Zipf, A.; Breuss-Schneeweis, P.; Beinat, E.; Boher, M. Live Cities and Urban Services: A Multi-dimensional Stress Field between Technology, Innovation and Society. In *Proceedings of 4th International Conference on Advanced Geographic Information Systems, Applications, and Services (GEOProcessing 2012),* Valencia, Spain, 30 January–4 February 2012; pp. 28–34.
58. Botts, M.; Percivall, G.; Reed, C.; Davidson, J. OGC® Sensor Web Enablement: Overview and High Level Architecture. In *Proceedings of Geosensor Networks: Second International Conference GSN*, Boston, MA, USA, 1–3 October 2006.
59. Uden, M.; Zipf, A. OpenBuildingModels: Towards a Platform for Crowdsourcing Virtual 3D Cities. In *Proceedings of 7th 3D GeoInfo*, Quebec City, QC, Canada, 16–17 May 2012; p. 17.