Advanced Engineering Informatics 26 (2012) 669–680

Contents lists available at SciVerse ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Generation and use of sparse navigation graphs for microscopic pedestrian simulation models

Angelika Kneidl [a,*], André Borrmann [a], Dirk Hartmann [b]

[a] Computational Modelling and Simulation Group, Technische Universität München, Germany
[b] Siemens AG, Corporate Technology, Germany

ARTICLE INFO

Article history:
Received 30 September 2011
Accepted 26 March 2012
Available online 18 April 2012

Keywords:
Navigation graph
Visibility graph
Microscopic pedestrian simulation
A* algorithm
Cellular automaton

ABSTRACT

For the spatial design of buildings as well as for the layout of large event areas, the crowd behaviour of the future users plays a significant role. The designing engineer has to make sure that potentially critical situations, such as high densities in pedestrian crowds, are avoided in order to guarantee the integrity, safety and comfort of the users. To this end, computational pedestrian dynamics simulations have been developed and are increasingly used in practice. However, most of the available simulation systems rely on rather simple pedestrian navigation models, which reflect human behaviour only in a limited manner. This paper contributes to enhancing pedestrian simulation models by extending a microscopic model by a navigation graph layer serving as a basis for different routing algorithms. The paper presents an advanced method for the automated generation of a spatially embedded graph which is on the one hand as sparse as possible and on the other hand detailed enough to be able to serve as a navigation basis. Three different pedestrian types were modelled: pedestrians with good local knowledge, pedestrians with partly local knowledge and those without any local knowledge. The corresponding algorithms are discussed in detail. To illustrate how this approach improves on simulation results, an example scenario is presented to demonstrate the difference between results with and without using a graph as constructed here. Another example shows the application of the extended simulation in a real-world engineering context. The article concludes with an outlook of further potential application areas for such navigation graphs.

1. Introduction

For the spatial design of buildings as well as for the layout of large event areas, the crowd behaviour of the future users plays a very important role. The designing engineer has to make sure that potentially critical situations, such as high densities of pedestrian crowds, are avoided in order to guarantee the integrity, safety and comfort of the users. In today's engineering practice, rough approximate calculations are used to determine the space required by pedestrian streams. However, these methods are neither able to capture the precise geometric setup of the investigated scenario nor can they consider the complex way-finding and walking behaviour of individual pedestrians. Accordingly, local phenomena are disregarded and potentially critical situations are easily ignored. To overcome these shortcomings, computational pedestrian dynamics simulations have been developed and are increasingly used in practice.

However, most of the available simulation systems either rely on rather simple pedestrian navigation models, which reflect real human behaviour only in a very limited manner, or are computationally expensive. This paper contributes to enhancing computationally cheap pedestrian simulation models by presenting a sophisticated graph-based approach for modelling navigational behaviour of humans. This allows engineers and architects to quickly and effortlessly evaluate different layout options. The implementation of this approach includes an advanced technique for generating sparse navigation graphs from a given spatial layout of the scenario under investigation.

2. Related work

The simulation of pedestrian crowds has been widely examined using a variety of approaches that focus on different details depending on the objective of the simulation [1]. For example, to determine minimum evacuation times for buildings or areas, macroscopic models are typically used. These focus on the overall situations of the simulated scenario and are based on mean values. Examples of such models are network flow models [2], fluid-dynamic models [3] or gas kinetic models [4]. To simulate the individual behaviour of pedestrians on the other hand, microscopic

* Corresponding author.
E-mail address: kneidl@tum.de (A. Kneidl).

1474-0346/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.aei.2012.03.006

models have been developed. These models consider the movements of each individual and focus on the interaction between individuals. Force models (e.g. Social Force Model by Helbing and Molnár [5]) as well as cellular automata [6] or agent-based models [7] belong to this category.

One central aspect of microscopic pedestrian simulation is to simulate the different movement strategies of individuals. Pelechano and Malkawi [8] categorize "virtual human technologies" into different features, such as appearance, function, time, autonomy and individuality.

The focus of this contribution lies on the latter: to differ between individual behaviour as a factor of sex and age, and – the authors' main focus – sense of orientation and familiarity with a location. The aim is to simulate large pedestrian crowds while taking into account different movement behaviours. An important constraint considered for the development of the corresponding algorithms is the requirement of high computational performance which allows for real-time simulations even on standard hardware. This provides the possibility to use the simulator as training facility for preparing and training the security staff of major events. – a feature strongly demanded by security authorities.

In order to assign individual behaviour to pedestrians, agent-based models are common. These assign different behavioural patterns to each individual, which results in different movement behaviour. Reynolds [9] models the perception of individuals with three different layers, namely a locomotion layer, a steering layer and an action selection layer. Musse and Thalmann [10] developed a human crowd behaviour model, consisting of a random behavioural model, which can be described by a few parameters. In [11], a personality model is mapped into a simulation model. Taking this a step further, Lerner [12] uses tracking from video data to obtain possible movements and trajectories. As these models have to calculate the new position of each pedestrian according to a complex set of rules in every time step, they are very computationally intensive and are capable of simulating only few pedestrians in real time. Another, faster way to assign individual behaviour is to use a navigation graph with different routing algorithms according to the individuals' preference. Since the objective is to simulate a large crowd in a large area in real time, the latter approach has been chosen.

Combining a microscopic layer with such graphs or networks was proposed by Asano et al. [13]. Here, a continuous microscopic model is used as operational model, i.e. to model the microscopic pedestrians' movement, in combination with a tactical model implemented as a network, for pedestrians' route assignment. The network consists of uniform square cells, which are connected by links. Ref. [14] combines an agent-based approach with a macroscopic network. In spite of taking cost functions and optimizing the flow, agents move through this network choosing the next vertex based on different criteria. The authors call this algorithm route choice self organisation (RCSO). However, both approaches do not focus on the derivation of a graph from a given geometry but take either such a network as given or simply divide space into uniform squares, the latter resulting in unrealistic wayfinding behaviour.

In contrast, this paper describes a technique for generating navigation graphs based on navigation points, which precisely reflect human navigational behaviour. At the same time, the graph consists of a minimum number of edges and vertices, enabling a high computational efficiency of the corresponding navigation algorithms.

A variety of alternative techniques have been proposed to create a navigation graph or roadmap from a given topography. Most of these techniques have been developed in the field of Robotics. Ref. [15] gives a good overview of the most common techniques of space decomposition. Ref. [16] describes all kind of planning algorithms, including motion planning algorithms. One technique for deriving a roadmap is to divide the space with Generalized Voronoi Diagrams (GVDs) [17] and to use the resulting lines as graph edges and the intersection points of the lines as graph nodes. The resulting graph consists of edges which are equidistant to each obstacle. A similar approach has been proposed in [18]: Here, agents navigate along combined Voronoi diagrams, which include not only obstacles but other moving agents as well. The intersection of the regions of the first order Voronoi diagram with the second order Voronoi diagram forms the navigation graph. The authors call this graph Multi-agent Navigation Graph (MaNG), which provides maximal clearance for each agent. This kind of graphs is suitable for steering robots, however they do not reflect human navigational cognition and are therefore of only limited applicability for pedestrian simulation.

Approaches which are capable to more accurately model human perception and cognition are based on visibility graphs [15]. A visibility graph consists of vertices defined by sources, destinations and obstacles within a scenario. Two nodes are connected if they are in line-of-sight. To avoid redundant edges, a reduced visibility graph can be constructed by categorizing edges into supporting and separating edges [15]. In [19], such a visibility graph is used to navigate agents through a scenario. Based on this visibility graph, a pre-computed shortest path map is stored. If other moving agents are located on the pre-calculated path, a recalculation has to be performed. Since this recalculation is very computational intensive, the focus of Choset's work lies on the approximation of agents' positions in order to minimize the number of recalculations by excluding agents which are outside the viewable region of the subject under examination. Gloor et al. [20] propose to construct a visibility graph by placing nodes at a certain distance from convex corners. This approach prevents simulated pedestrians from walking too close around a corner, but it also produces many nodes, which are dispensable.

In this paper, we describe a novel navigation graph generation algorithm which is based on the idea of placing nodes at a certain distance from each corner, but discards all superfluous nodes. Furthermore, the resulting graph is not as dense as a common visibility graph because geometrically close edges are omitted.

3. Model setup

An important requirement is that the simulator is able to run in real time, as the simulator is designed as a training tool. To achieve such high performance, a cellular automaton model for space discretization in combination with a conservative force model [21] has been chosen, i.e. a model based on energy potentials that describe the influencing forces on each pedestrian (attracting force of the destination, repellent forces of obstacles as well as the repellent forces of other pedestrians). Combining the cellular automaton with these forces, the navigation of single pedestrians can be modelled efficiently [22]. This combination makes it possible to quickly update pedestrians' positions while taking into account the interactions between them. However, using this approach key aspects of pedestrian movement are neglected, namely the individual navigational behaviour of pedestrians as well as the large-scale orientation of pedestrians. Using only the cellular automaton model, the simulated pedestrians appear as being short-sighted, since only neighbouring cells are considered in each update step.

In order to take into account different degrees of knowledge of efficient routes towards a destination as well as individual navigational behaviour (e.g. keep as close as possible to the direction to the destination, choose routes with less turns, etc.) without losing computational speed, the basic model is extended in our proposal by a navigation graph. Pedestrians move according to the cellular automaton from one graph node to the next. The graph models the large-scale orientation of pedestrians. Thus the pedestrians'