



Fakultät für Bauingenieur- und Vermessungswesen
Lehrstuhl für Computergestützte Modellierung und Simulation
Prof. Dr.-Ing. André Borrmann

Einbindung von Evakuierungssimulation in den Bauplanungsprozess am Beispiel der CAD-Pläne der zweiten S-Bahn Stammstrecke München

Heiner Kunkel

Bachelorthesis

für den Bachelor of Science Studiengang Umweltingenieurwesen

Autor: Heiner Kunkel

Matrikelnummer:

1. Betreuer: Prof. Dr.-Ing. André Borrmann

2. Betreuer: Dipl.-Ing. Javier Ramos Jubierre, M.Sc.

3. Betreuer Maximilian Bügler, M. Sc.

Ausgabedatum: 10. Juni 2013

Abgabedatum: 8. November 2013

Zusammenfassung

In dieser Bachelorarbeit geht es um die Einbindung von Evakuierungssimulation in den Planungsprozess von Bauvorhaben. Um dies in einem Beispiel zu testen werden die CAD-Pläne aus der Genehmigungsplanung der Station Marienhof für die neu geplante zweite S-Bahn Stammstrecke in München verwendet. Als CAD (Computer Aided Design) bezeichnet man den Entwurf und die Konstruktion von Produkten durch Computerprogramme. Gearbeitet wird hier mit dem CAD-Programm AutoCAD von Autodesk und dem Fußgängersimulator CrowdControl von Siemens, der bei einem Projekt der Siemens AG entstanden ist. Von zentraler Bedeutung dieser Arbeit ist die Übertragung von für die Simulation wichtigen Objekten aus AutoCAD in den Simulator.

AutoCAD arbeitet mit DWG-Dateien und der Simulator mit XML-Dateien. Diese Datenformate sind von Grund auf unterschiedlich. Um Objekte aus den DWG-Dateien in den Simulator laden zu können, wird ein Programmzusatz (i. F. Plug-In) für AutoCAD entwickelt, das es dem Nutzer ermöglicht, einzelne Objekte auszuwählen und diese in eine XML-Datei zu exportieren. Diese Datei kann dann vom Simulator geöffnet werden. Für das Plug-In wird mit der Entwicklungsumgebung Microsoft Visual Studio gearbeitet, die durch die von Autodesk zur Verfügung gestellte Programmierschnittstelle mit AutoCAD in Verbindung steht. Dabei geht es im Plug-In um eine möglichst benutzerfreundliche und selbsterklärende Bedienung. Das ist die Voraussetzung dafür, dass die Anwendung von den Planern im bestehenden Planungsprozess angenommen wird. Zur Veranschaulichung wird ein Evakuierungsszenario von der Bahnsteigebene der geplanten Station Marienhof in München entwickelt.

Durch die Ergebnisse des Plug-Ins und das durchgeführte Evakuierungsszenario werden das Potential und die Probleme der Einbindung von Evakuierungssimulationen in den Planungsprozess deutlich. Insbesondere hat auch Building Information Modeling eine große Bedeutung für Fußgängersimulationen. Hier wird dreidimensional modelliert und einzelne Bauteile können mit genaueren Informationen versehen werden. Eine verpflichtende Einführung von Evakuierungssimulation für bestimmte Bauprojekte wäre damit noch einfacher durchsetzbar als mit herkömmlicher CAD-Software. Die Simulationen könnten von den Planern durchgeführt werden, die schon jetzt für die Erarbeitung der Fluchtwege zuständig sind. So wäre die Einbindung in den Bauplanungsprozess nur noch Formsache. Vorteil wären verbesserte Fluchtwege und ein grundlegendes Verständnis der Planer für das Verhalten von Personen im Evakuierungsfall. Problemstellen werden hier sehr anschaulich sichtbar. Einige Menschenleben könnten in Zukunft auf diese Art und Weise gerettet werden.

Abstract

This thesis introduces an AutoCAD application that integrates evacuation simulation into the planning process of construction projects. To test the application the CAD drawings of the new planned second subway tunnel in Munich are used. In this work, the CAD program AutoCAD from Autodesk is used together with the pedestrian simulator CrowdControl created by Siemens AG. Central to this work is the transmission of the design elements that are important for the simulation from AutoCAD into the simulator.

AutoCAD works with DWG files and the simulator with XML files. Since data formats are completely different, in order to load objects from DWG files into the simulator, a plug-in for AutoCAD is developed that allows the user to select individual elements and load them into an XML file. This file can be opened by the simulator. The plug-in works with the application programming interface Visual Studio of Microsoft, which is connected to AutoCAD by an Application Programming Interface (API) available from Autodesk. The plug-in should be user-friendly and self-explanatory to be accepted by the planners in the existing planning process. This plug-in loads the simulation elements together with the sources and targets into the pedestrian simulator. Based on this data, an evacuation scenario of the platform level of the station Marienhof is developed.

The results of the plug-in and the evacuation scenario show the potential and problems of integrating evacuation simulation into the planning process. In particular, the building information systems have a big influence on pedestrian simulation. It would be very useful to make evacuation simulation mandatory for projects with large crowds. The simulations could be carried out by the planners who are now responsible for the development of the escape routes. The advantage would be improved escape routes and a more fundamental understanding among the planners of crowd behavior in evacuations. In the case study, the problem areas of the evacuation become clearly visible, allowing many human lives to be saved.

Inhaltsverzeichnis

Zusammenfassung	2
Abstract	3
1 Einführung und Motivation	6
1.1 Einführung zweite S-Bahn Stammstrecke in München	7
1.2 Ziel der Arbeit	8
1.3 Aufbau der Arbeit	9
2 Evakuierungssimulation	10
2.1 Phänomene bei Bewegungen großer Menschenmengen	11
Grundbewegungsbilder	11
Phänomen der Selbstorganisation	13
2.2 Modelle der Fußgängersimulation	14
Zelluläre Automaten	15
Soziale-Kraft-Modelle	15
Entscheidungs-Aktivitäts-Modelle (engl.: Activity Choice Models)	15
Geschwindigkeitsbasierte Modelle	15
Kontinuum-Modelle	16
Hybrid-Modelle	16
Verhaltens-Modelle	16
Netzwerk-Modelle	16
2.3 Fußgängersimulator CrowdControl	17
Simulationsmodell	17
Anwendung des Simulators	19
Datenformat XML	20
3 Planungsprozess von Bauvorhaben	22
3.1 Computer Aided Design (CAD) im Planungsprozess	22
CAD-Systeme	23
CAD-Grundfunktionen	24
Struktur eines CAD-Modells	24
3.2 CAD-Pläne der Station Marienhof	27
Datenformat DWG	28
4 Programmzusatz (Plug-In) zur vereinfachten Datenübertragung	29
4.1 Programmzusatz (Plug-In) für AutoCAD	29

Entwicklungsumgebung Microsoft Visual Studio	29
Programmierschnittstelle (API).....	30
Softwareentwicklungskasten (SDK)	30
Assistent (Wizard).....	31
4.2 XML-Serializer.....	31
4.3 Graphical User Interface (GUI).....	32
4.4 Workflow für den Quellcode des Plug-Ins	33
Area	33
SelectWall	35
Source.....	36
Target	36
ConvertToXML.....	38
5 Ergebnisse	41
5.1 Durchführung einer Simulation anhand der übertragenen Daten.....	41
5.2 Probleme und Möglichkeiten.....	46
Simulation	46
Plug-In.....	48
6 Ausblick in die Zukunft.....	50
6.1 Fußgängersimulation für die Planung von Fluchtwegen.....	50
6.2 Building Information Modeling (BIM)	51
Anhang A	53
A.1 Beispiel einer XML-Datei	53
A.2 Vorgang der Installation mit Einführung in das Plug-In.....	54
Anhang B.....	56
B.1 Compact Disc	56
Literaturverzeichnis.....	57

1 Einführung und Motivation

Todesfälle bei Katastrophen durch fehlende Fluchtmöglichkeiten beschäftigen die Menschheit schon lange. Leider waren immer erst schwerwiegende Vorfälle notwendig, um die Fluchtwege zu verbessern. Beispielsweise gab es beim Ringtheaterbrand am 8. Dezember 1881 in Wien 384 Todesopfer. Grund hierfür war ein ausgebrochenes Feuer in Verbindung mit einem fehlerhaft durchdachten Fluchtplan. Die Besucher fanden den Ausgang nicht und nach innen aufgehende Türen erschwerten den Weg nach außen (Österreichische Nationalbibliothek, 1881). Im Jahr 1883 wurde dann eine entsprechende Bauordnung in das niederösterreichische Landesgesetz integriert (Franz Joseph I. & Graf Eduard Taaffe, 1883).

Trotz reichlicher Verbesserungen kommt es auch heute noch zu Problemen. Beim jährlich stattfindenden Haddsch in Mekka sterben immer noch regelmäßig Menschen (1990: 1.426, 1994: 270, 1997: 343, 1998: 107, 2001: 35, 2003: 14, 2004: 244 und 2006: 364 Todesopfer) (Meyer, 2008). In Deutschland sorgte vor Allem das Unglück der Loveparade 2010 in Duisburg für Aufruhr. Auf Grund fehlgeleiteter Besucherströme und Fehlplanungen kamen im Eingangsbereich 21 Besucher ums Leben (Dörries, 2013).

Da die Personenstromsimulation in den letzten Jahren große Fortschritte gemacht hat, werden nun Stimmen laut, diese für Großveranstaltungen und öffentliche Gebäude verpflichtend einzuführen. Ziel ist es, an Hand der in der Planungsphase von Bauprojekten entstehenden Pläne Evakuierungssimulationen durchzuführen. Zwar wäre das heute technisch schon machbar, es fehlen jedoch noch Möglichkeiten, die Simulatoren besser in die bestehenden Zeichenprogramme einzubinden. Um Evakuierungssimulationen in der Praxis zu etablieren, benötigt man daher noch neue Denkansätze. Auf der Suche eines solchen Ansatzes wurde in dieser Arbeit das Projekt der noch in Planung befindlichen zweiten S-Bahn Stammstrecke München genauer betrachtet (Deutsche Bahn, Projekt 2.stammstrecke-muenchen, 2013).

1.1 Einführung zweite S-Bahn Stammstrecke in München

Zur Leistungserhöhung der bestehenden Stammstrecke, sollen parallel dazu zwei neue Gleise in Hin- und Rückrichtung mit je zehn Kilometer Länge gebaut werden. Die Strecke beinhaltet zwei insgesamt sieben Kilometer lange Tunnel und drei unterirdische Stationen am Hauptbahnhof, Marienhof und Ostbahnhof (Hauff, 2013).



Abbildung 1.1: Übersicht mit der bestehenden Stammstrecke in grün und der geplanten zweiten Stammstrecke in rot (Hauff, 2013)

Weil die Strecke mehrere U-Bahnen kreuzt, befinden sich die Stationen in rund 40 Meter Tiefe. Der Fußgängerverkehr soll mit Aufzügen und Rolltreppen abgewickelt werden. Zusätzlich gibt es noch weitere Fluchttreppen. Die bestehende U- und S-Bahn wird über unterirdische Übergänge angebunden. Es handelt sich also um ein komplex verflochtenes Netz aus Fußwegen, das im Ernstfall schwierig zu evakuieren ist. Daher ist es ein sehr interessantes Beispiel, um eine Fußgängersimulation zu testen (Deutsche Bahn, www.2.stammstrecke-muenchen.de, 2013).

Um den Umfang etwas zu reduzieren, beschränkt sich diese Arbeit auf die Station Marienhof. Sie liegt in Ost-Westrichtung mittig unter dem Marienhof. Mit 210 Meter Länge unterquert sie die U-Bahnlinien U3 und U6. Die Station besteht aus den Ebenen Bahnsteigebene, Verteilerebene, Treppengeschoss, Sperrengeschoss und Oberfläche. Anfallende Fahrgastströme werden größten teils aus der Erschließung des Innenstadtzentrums und dem Umstieg in die U-Bahnlinien U3 und U6 bestehen (Deutsche Bahn, www.2.stammstrecke-muenchen.de, 2013).

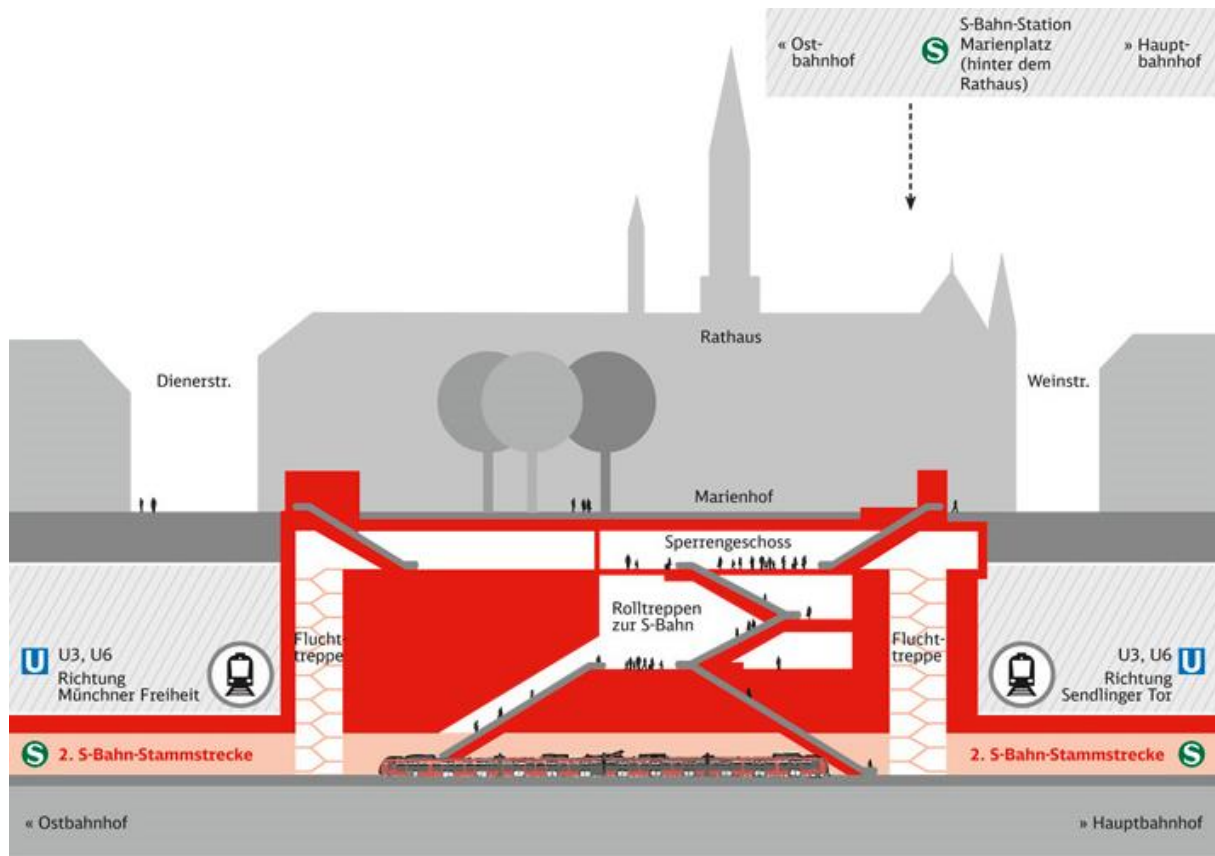


Abbildung 1.2: Längsschnitt der geplanten Station Marienhof (Deutsche Bahn, www.2.stammstrecke-muenchen.de, 2013)

1.2 Ziel der Arbeit

Als Ausgangsbasis für die Simulation dienen die Grundrisse des Planungsbüros Obermeyer, das hier freundlicherweise die Pläne zur Verfügung gestellt hat. Die Pläne liegen im DWG-Format vor und können mit digitalen Zeichenprogrammen geöffnet werden. Die Entscheidung fiel hier auf das oft verwendete und bewährte Programm AutoCAD von Autodesk. Als Simulator dient der aus dem Projekt CrowdControl der Firma Siemens entstandene Fußgängersimulator. Der Simulator arbeitet mit XML-Dateien. Hier stellte sich als größtes Problem heraus, dass sich die DWG-Dateien nicht in den Fußgängersimulator laden lassen. Zwar kann man die Zeichenpläne über eine DXF-Datei in den Simulator laden, man kann so jedoch nicht zwischen für die Simulation notwendigen und nicht notwendigen Linien unterscheiden. Dies führt zu Unübersichtlichkeit durch zu viele Linien im Simulator. Dort kann man überflüssige Linien dann nur mit sehr viel Zeitaufwand löschen. Außerdem fällt es schwer, nur den für die Simulation notwendigen Bereich der Pläne anzuzeigen. Die Idee ist es, einen Programmzusatz (i. F. Plug-In) für AutoCAD zu entwickeln, der es dem Nutzer ermöglicht nur die für die Simulation notwendigen Linien in eine XML-Datei zu exportieren. Linien stellen im Simulator Wände dar. Das sind Widerstände, die von den Fußgängern nicht überschritten werden können. Besonders wichtig bei der Entwicklung des Plug-Ins ist die Benutzerfreundlichkeit. Durch Hilfestellungen soll der Nutzer möglichst intuitiv durch das Programm geleitet werden. Die entstehende XML-Datei kann anschließend mit CrowdControl geöffnet und mit entsprechendem Input gespeist werden. Nur durch eine möglichst einfache

Bedienung kann Fußgängersimulation in den Prozess der Bauplanung nachhaltig integriert werden. Ziel dieser Arbeit ist es zu prüfen, wie man Evakuierungssimulation bestmöglich in den bestehenden Planungsprozess einbinden kann und wo die Möglichkeiten und Probleme der Simulation liegen.

1.3 Aufbau der Arbeit

In Kapitel 2 gibt es eine allgemeine Einführung in das Thema Evakuierungssimulation. Hier werden die Phänomene bei der Bewegung großer Menschenmengen, Modelle der Fußgängersimulation und das Projekt CrowdControl genauer betrachtet. Bei der Bewegung der Menschenmengen geht es um Grundbewegungsbilder und Phänomene der Selbstorganisation. Die Modelle bauen auf unterschiedlichen Ansätzen und Möglichkeiten zur Umsetzung eines Fußgängersimulators auf. Anwendung und Datenformat des im Projekt CrowdControl entstandenen Fußgängersimulators werden hier genau erklärt.

Kapitel 3 beschäftigt sich mit dem Planungsprozess von Bauvorhaben. Dabei geht es eingehend um digitales Zeichnen und Modellieren im Planungsprozess und die Zeichenpläne der Station Marienhof. Die verschiedenen Zeichenprogramme, deren Grundfunktionen und die Struktur der Pläne werden hier genauer dargestellt.

Danach, in Kapitel 4, geht es um die Übertragung von Objekten aus den Zeichenplänen in den Fußgängersimulator. Hier wird erklärt, was ein AutoCAD-Plug-In, ein XML-Serializer und eine GUI-Anwendungen ist und wie man ein Plug-In für die Datenübertragung entwickelt. Dabei wird genauer auf die Entwicklungsumgebung, die Programmierschnittstelle, den Softwareentwicklungskasten und den Assistenten eingegangen. Im Punkt Workflow werden die nötigen Funktionen für eine Datenübertragung genauer erklärt.

In Kapitel 5 wird eine Simulation anhand der Pläne der Station Marienhof und dem dafür entwickelten Plug-In durchgeführt. Hier wird genau auf die dabei entstandenen Probleme und die daraus entstehenden Möglichkeiten für die Bauplanung eingegangen. Auch die Einbindung von Building Information Modeling (BIM) in den Planungsprozess wird an dieser Stelle in Bezug auf Evakuierungssimulation genauer begutachtet. Dabei handelt es sich um dreidimensionale Modellierung, wobei besonders viel Wert auf bauteilspezifische Informationen gelegt wird (Rank, Teil II: Ingenieur Anwendungen, 2010).

2 Evakuierungssimulation

Die Beurteilung der Sicherheit von Veranstaltungen mit großen Menschenmengen hat sich in der Vergangenheit als äußerst schwierig erwiesen. Die äußeren Gegebenheiten und das Verhalten der Fußgänger können sehr unterschiedlich sein. Während sich die Besucher eines Festivals frei zwischen den Bühnen auf großen freien Flächen bewegen können, gehen die Beteiligten eines Umzugs bis zum Ziel näherungsweise auf einer Linie. Bei beiden Situationen können bei hoher Personendichte gefährliche Situationen entstehen (Duives, Daamen, & Hoogendoorn, 2013).

Es existieren viele verschiedene Modelle für Fußgängersimulationen. Einige davon sind jedoch für Situationen mit geringer Personendichte entwickelt worden. Ein Beispiel hierfür wären die Bewegungen und das Verhalten von Besuchern einer Shopping Mall. Hier stehen eher Marketinginteressen im Vordergrund und weniger die Sicherheit der einzelnen Einkäufer. Die Anwendbarkeit der Systeme für Evakuierungssimulationen und die Frage, welche Faktoren von dem Simulator miteinbezogen werden, müssen daher eingehend geprüft werden. Die Ergebnisse unterschiedlicher Simulatoren können sich dadurch stark voneinander unterscheiden. (Duives, Daamen, & Hoogendoorn, 2013)

Mögliche Faktoren ergeben sich beispielsweise durch die Fragen:

- Befinden sich andere Fußgänger in der Nähe?
- Stehen sie unter Druck?
- Haben sie ein bestimmtes Ziel?
- Herrscht eine freundliche Grundstimmung?
- Handelt es sich um eine bestimmte Altersgruppe?
- Ein bestimmtes Geschlecht?
- Eine heterogene Menschenmenge?
- Fühlen sich einzelne Fußgänger als Individuen oder als Teil einer Gruppe?
- Tragen sie Gepäck mit sich?
- Sind sie mit der Infrastruktur vertraut?

Bei der Modellierung wirken sich diese Fragen auf die Geschwindigkeit, den Weg, den Platzbedarf oder das operative Verhalten der Fußgänger aus. Die Geschwindigkeit hängt von der Behinderung durch andere Fußgänger, der Altersgruppe oder dem Gepäck ab. Der Weg ergibt sich durch das Ziel und mögliche Zieländerungen. Er kann stark variieren, wenn die Umgebung nicht vertraut ist, da hier eine optimale Routenwahl nicht möglich ist. Der Platzbedarf einzelner Personen steigt zum Beispiel bei Flughäfen, weil durch das Gepäck zusätzliche Fläche nötig ist. Das operative Verhalten von Mitgliedern einer Gruppe hält sich stärker an das vorgegebene Ziel, als ein einzelner Fußgänger, der möglicherweise auf dem Weg noch andere mögliche Ziele entdeckt und diese besucht. Neben den bisher genannten Phänomenen gibt es noch Verhaltensweisen, die nur bei Bewegungen besonders großer Menschenmengen auftreten. Diese werden im nächsten Unterpunkt genauer untersucht.

2.1 Phänomene bei Bewegungen großer Menschenmengen

Bei Simulationsmodellen für Menschenmengen ist es wichtig, dass sie nicht nur das Verhalten jeder einzelnen Person in der Mikrosichtweise darstellen, sondern auch das der gesamten Menge in der Makrosichtweise. Hieraus ergeben sich oft besonders prekäre Situationen für die einzelnen Individuen, die sonst nicht berücksichtigt würden. Außerdem ist bei großen Menschenmengen häufig eine gewisse Selbstorganisation zu beobachten. Dazu werden im Folgenden einige Grundbewegungsbilder erklärt

Grundbewegungsbilder

Man unterscheidet hier zwischen Flüssen in eine Richtung und Flüssen in mehrere Richtungen. Bei Flüssen in eine Richtung kommt es eher zu Interaktionen in gleicher Richtung und bei Flüssen in zwei Richtungen zu Kollisionen in entgegengesetzter Richtung. Flüsse in eine Richtung können entweder gerade sein, um eine Ecke, zu einem Eingang oder zu einem Ausgang führen.

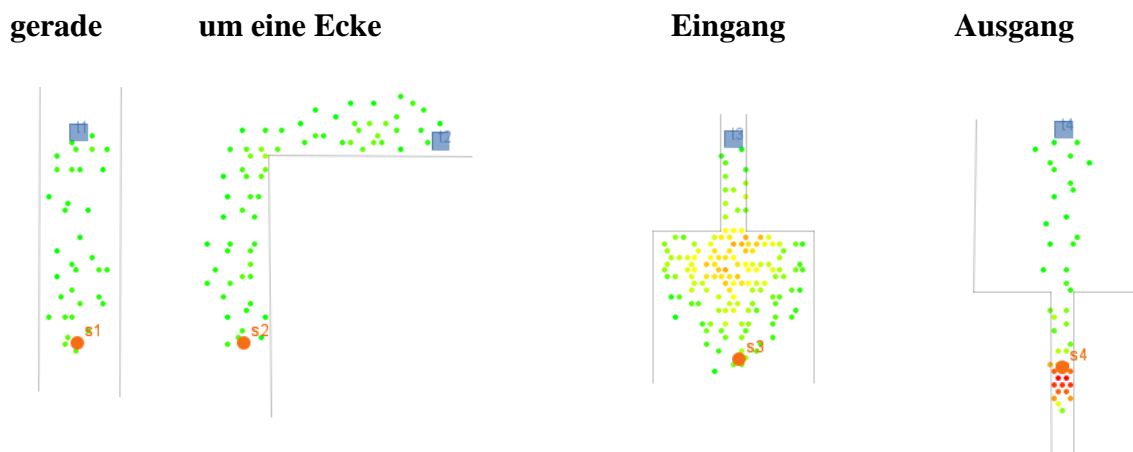


Abbildung 2.1: Screenshot für Beispiele mit einem Strom aus dem Fußgängersimulator CrowdControl. Rote Kreise stellen Quellen und die blauen Quadrate Ziele dar. Die kleinen Punkte sind Personen.

Hier werden die einzelnen Personen nicht, wie in der Hydromechanik, bei der Engstelle beschleunigt, sondern gehen im gleichen Tempo weiter bzw. verhaken sich vor der Engstelle. So kann es leicht zum Stau kommen. Besondere Gefahr gilt hier, wenn von hinten eine große Gruppe Richtung Engstelle drückt, wie es bei Ausgängen häufig der Fall ist.

Bewegungen in mehrere Richtungen (≥ 2 Richtungen) können entweder gegensätzlich parallel verlaufen oder sich unter einem bestimmten Winkel kreuzen.

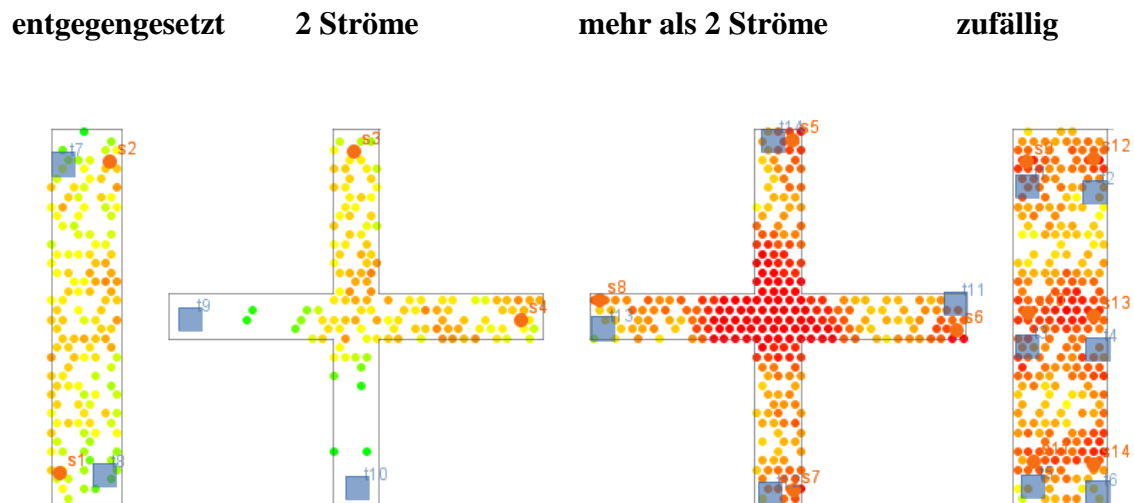


Abbildung 2.2: Screenshot für Beispiele mit mehreren Strömen aus dem Fußgängersimulator CrowdControl. Rote Kreise stellen Quellen und die blauen Quadrate Senken dar. Die kleinen Punkte sind Personen.

Gegensätzlich parallele Bewegungen gibt es häufig bei Festivals, wenn Besucher nach einem Konzert von Bühne A zu Bühne B gehen und umgekehrt. Beim Kreuzen hängt das Verhalten vom Platzangebot, der Anzahl an Strömen und der Stärke der einzelnen Ströme ab. Hier können sehr unterschiedliche Bewegungsbilder entstehen. Daher unterscheidet man nochmals zwischen zufälligem Kreuzen und Kreuzen an einem Brennpunkt (Duives, Daamen, & Hoogendoorn, 2013). Zufälliges Kreuzen wäre zum Beispiel in einer Fußgängerzone, wo viele Leute durcheinander die Straße kreuzen, um zum nächsten Geschäft zu gelangen. Kreuzen an einem Brennpunkt wäre eine Situation, in der eine Gruppe von Ost nach West und eine andere von Nord nach Süd geht, und die beiden Gruppen sich in der Mitte, im Brennpunkt, treffen. Mögliche Verhaltensweisen wären hier das Abschneiden der schwächeren Gruppe, bis diese zu einer größeren anschwillt oder das Durchmischen der beiden Gruppen im Brennpunkt. Das Kreuzen in einem Brennpunkt unterteilt man weiter in einen Brennpunkt mit zwei und mit mehr als zwei Strömen (Duives, Daamen, & Hoogendoorn, 2013).

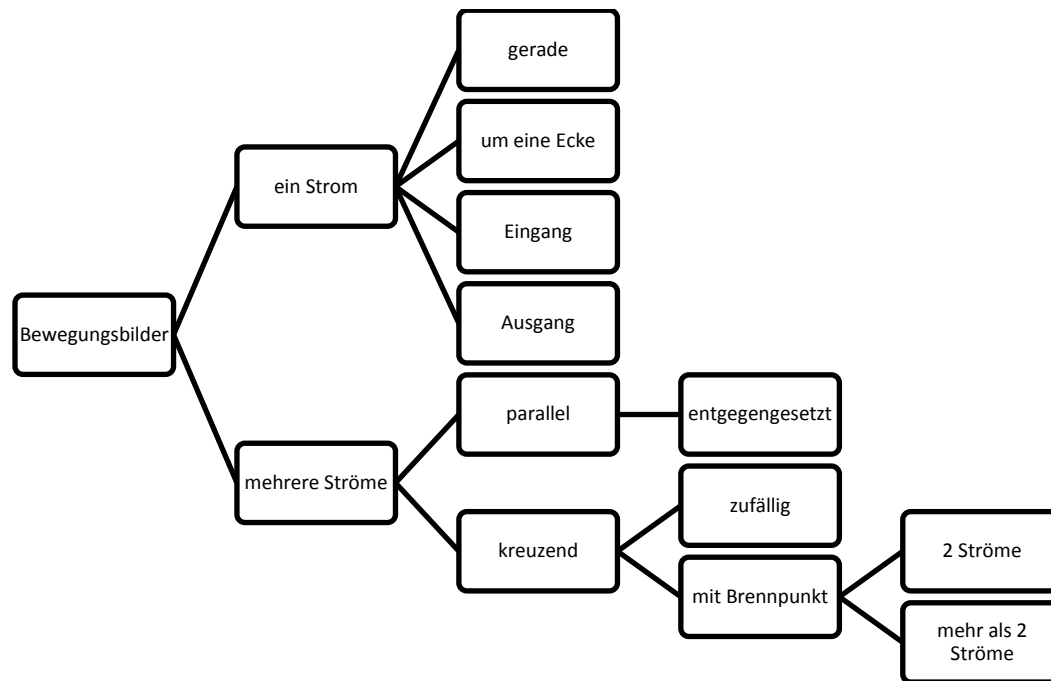


Abbildung 2.3: Grundbewegungsbilder von Menschenmengen (Duives, Daamen, & Hoogendoorn, 2013)

Neben den möglichen Bewegungsbildern hängt das Verhalten einer Gruppe zusätzlich von der Selbstorganisation der Menge ab. Dieses Thema soll nun genauer erklärt werden.

Phänomen der Selbstorganisation

Im Moment kennt man sechs verschiedene Formen der Selbstorganisation (Duives, Daamen, & Hoogendoorn, 2013):

- Entstehung (engl. emergence)
- Spurbildung
- Stop & Go Wellen
- Herdenverhalten (engl. herding)
- Reißverschluss effekt
- Faster-is-Lower

Von Entstehung spricht man beispielsweise beim Ausbrechen einer Panik. Es ist eine spontane Abweichung von einer linearen Bewegung, die meist auf ein zu klein dimensioniertes Platzangebot und zu wenige Fluchtwege zurückzuführen ist (Meyer, 2008).

Spurbildung ist, wenn sich dynamische „Fahrspuren“ bilden, in denen sich die Personen nur in eine Richtung bewegen. Sie tritt häufig bei parallel entgegengesetzten Strömen auf. Auf diese Weise werden starke Wechselwirkungen mit den entgegenkommenden Fußgängern reduziert, was bequemer ist und höhere Gehgeschwindigkeiten erlaubt (Meyer, 2008).

Stop & Go Wellen breiten sich in Längsrichtung aus und zeichnen sich durch periodische Dichteänderungen in Raum und Zeit aus. Sie sind vergleichbar mit einer Staubildung im Straßenverkehr bei hoher Verkehrsbelastung. Ab sieben Personen pro Quadratmeter Verhält sich die Menge wie ein Fluid, wodurch es zu so genannten Schockwellen kommen kann, bei

denen die Beteiligten den Kontakt zum Boden verlieren und bis zu drei Meter und mehr umhergedrückt werden (Meyer, 2008; Duives, Daamen, & Hoogendoorn, 2013).

Herdenbildung nennt man den Effekt, wenn Fußgänger in unklaren Situationen dazu neigen einander zu folgen, anstatt den für sie optimalsten Weg zu wählen (Duives, Daamen, & Hoogendoorn, 2013). Dieses Verhalten kennt man zum Beispiel bei der Ankunft am Flughafen auf dem Weg zum Gepäckband. Erst warten alle bis der erste losgeht und dann folgen ihm alle. Dieses Verhalten tritt leider auch häufig bei Evakuierungssituationen auf und kann sehr gefährlich werden.

Besonders beim Reißverschlusseffekt ist, dass normalerweise jeder Fußgänger ein Territorium um sich herum hat, in das er keine anderen Personen lässt. In speziellen Fällen lässt er jedoch Personen in diesen Bereich, wenn diese nicht direkt vor ihm, sondern diagonal vor ihm sind. Bei Engstellen kann das Territorium jedes Individuums also als kleiner angenommen werden als auf freier Fläche (Duives, Daamen, & Hoogendoorn, 2013; Meyer, 2008).

Das letzte Phänomen der Selbstorganisation ist der Faster-is-Lower Effekt. Er bedeutet, dass Personen vor einer Engstelle weiter nach vorne drücken und so den „Flaschenhals“ verstopfen. Es kommt zu Koordinationsproblemen, da aufgrund der hohen Dichte nur noch wenige Lücken für viele Personen vorhanden sind. Bei fortschreitender Dichte wird die Menschenmenge zusätzlich durch körperliche Interaktionen und Reibung verlangsamt (Duives, Daamen, & Hoogendoorn, 2013).

Während die Spurbildung und der Reißverschlusseffekt von außen betrachtet logisch erscheinen, wirken die Entstehung einer Panik, Stop & Go Wellen, Herdenverhalten und Faster-is-Lower eher irrational. Es ist jedoch sehr wichtig, all diese Effekte der Selbstorganisation in einem Fußgängersimulator zu berücksichtigen. Dazu wurden verschiedene Modelle entwickelt. Bisher gibt es noch kein Modell, das alle Eigenschaften für jede Situation ausreichend abbildet. Jedes Modell muss daher speziell für die notwendigen Anforderungen eines Projekts ausgewählt werden. Einen kleinen Einblick gibt es im nächsten Unterpunkt.

2.2 Modelle der Fußgängersimulation

Da es kein Modell gibt, das alle Grundbewegungsbilder und Phänomene der Selbstorganisation gleich gut abbildet, muss auf unterschiedliche Modelle zurückgegriffen werden. Im Moment spricht man in der Literatur von Zellulären Automaten, Soziale-Kraft-Modellen, Entscheidungs-Aktivitäts-Modellen, geschwindigkeitsbasierten Modellen, Kontinuitäts-Modellen, Hybrid-Modellen, Verhaltens-Modellen und Netzwerk-Modellen. Jedes dieser Modelle hat Vor- und Nachteile bezüglich der folgenden Faktoren (Duives, Daamen, & Hoogendoorn, 2013):

- Externer Einfluss auf das Bewegungsverhalten
- Globale Routenwahl
- Adaptive Routenwahl

- Kollisionsvermeidung
- Druck
- Gruppen in der Menschenmenge
- Verschiedene Personentypen
- Rechenaufwand
- Unbekannte Infrastruktur/Veranstaltung

Im nächsten Abschnitt werden die einzelnen Modelle kurz erklärt.

Zelluläre Automaten

Hier handelt es sich um mikroskopische Modelle, die diskret in Zeit, Raum und Bewegungszustand sind. Das bedeutet, dass zum Beispiel pro Sekunde nur zehn Zustände ermittelt werden und die Fläche durch ein Gitter in 40 x 40 cm große Teilflächen aufgeteilt wird. Das entspricht in etwa der geringsten Fläche, die ein Fußgänger für sich beansprucht. Jeder Teilfläche kann höchstens eine Person zugeordnet werden. Der Übergang eines Fußgängers zu einer Nachbarzelle hängt von dynamischen Entscheidungsregeln ab, die nach jedem Zustand wieder neu berechnet werden. Diese berücksichtigen das Ziel, die Interaktion mit anderen Fußgängern und die Infrastruktur (Meyer, 2008).

Soziale-Kraft-Modelle

Hierbei handelt es sich ebenfalls um mikroskopische Modelle, bei denen jedoch die Wechselwirkungen zwischen den Fußgängern auf dem Prinzip der sozialen Kraft beruhen. Die Idee kommt aus der Dynamik und teilt jedem Fußgänger einen Kraftvektor zu. Bei einer Interaktion kommt es zu Änderungen von Bewegungsrichtung und Geschwindigkeit der Beteiligten. Unter Berücksichtigung von Ziel, Zeit und Komfort wird nach dem optimalsten Weg gesucht. Im Gegensatz zum Zellulären Automaten ist es ein dynamisches Modell, das nicht in diskrete Schritte unterteilt ist (Meyer, 2008).

Entscheidungs-Aktivitäts-Modelle (engl.: Activity Choice Models)

Entscheidungs-Aktivitäts-Modelle erweitern das Soziale-Kraft-Modell um die Verhaltensweisen von Fußgängern. Sie arbeiten mit einer kontinuierlichen Routenwahl in Raum und Zeit. Hier geht es jedoch nicht nur um den optimalsten Weg in Bezug auf Ziel, Zeit und Komfort, sondern es wird beispielsweise auch eine Funktion eingebaut, die den Fußgänger bei zu großem Aufwand wieder umkehren lässt (Duives, Daamen, & Hoogendoorn, 2013).

Geschwindigkeitsbasierte Modelle

Beim geschwindigkeitsbasierten Modell geht man nicht wie beim Soziale-Kraft-Modell davon aus, dass jedes Individuum eine Kraft ausübt, sondern dass die Individuen versuchen allen Hindernissen in ihrem Blickfeld auszuweichen. Dabei wird zwischen dem kürzesten Weg

zum Ziel und einem Weg mit möglichst wenigen Kollisionen abgewogen. Solche Modelle kommen aus der Spieleindustrie (Duives, Daamen, & Hoogendoorn, 2013).

Kontinuum-Modelle

Das Kontinuum-Modell ist ein makroskopisches Modell. Man verzichtet auf die Dynamik der einzelnen Individuen und sieht die Menschenmenge als globales Potentialfeld. Ebenfalls wie in anderen Modellen ergibt sich der Weg aus Ziel, Zeit und Komfort. Die dazu benötigten Differentialgleichungen kommen aus der Hydrodynamik und der Gaskinetik. Menschenmengen werden also ähnlich wie ein Strom aus Wasser betrachtet. Die einzelnen Wege bleiben bei diesem Modell unbeachtet (Duives, Daamen, & Hoogendoorn, 2013; Meyer, 2008).

Verhaltens-Modelle

Hier versucht man als Erweiterung der Entscheidungs-Aktivitäts-Modelle zusätzlich strategische, soziale und psychologische Aspekte miteinzubeziehen. Dazu reduziert man die Grundbewegungsparameter auf ein Minimum. Dafür können jedoch Einflüsse, wie Licht, Sicherheitsgefühl oder Schatten in das Modell miteinbezogen werden. Man untersucht Abweichungen von der optimalen Routenwahl aufgrund von sozialen Begegnungen (Duives, Daamen, & Hoogendoorn, 2013).

Netzwerk-Modelle

Die Wege der Fußgänger werden in kleinere Abschnitte unterteilt. Dazu werden mehrere Knotenpunkte definiert, die dann mit Kanten verbunden werden. Punkte sind beispielsweise Räume und Kanten sind die Wege zwischen den Punkten. Fußgänger werden als einzelne Fließobjekte behandelt, die ein bestimmtes Ziel, eine Geschwindigkeit und einen Platzbedarf haben. Diese werden dann auf das Netzwerk umgelegt (Duives, Daamen, & Hoogendoorn, 2013). Das Modell baut auf der Graphentheorie auf und es können Algorithmen, wie beispielsweise der Kürzesten-Wege-Algorithmus von Dijkstra, dafür angewendet werden (Diestel, 2010).

Hybrid-Modelle

Bei Hybridmodellen kombiniert man mikro- und makroskopische Modelle. Ziel ist es, die Vorteile beider Modelle zusammenzuführen (Duives, Daamen, & Hoogendoorn, 2013). Mikroskopische Modelle haben den Vorteil, dass sie jeden Weg einzeln darstellen und sich somit Verhaltensmuster und verschiedene Menschentypen implementieren lassen - zum Beispiel Personen unterschiedlichen Alters. Vorteil von makroskopischen Modellen ist, dass man beispielsweise mit Hilfe des Netzwerkmodells bessere Aussagen über das Gesamtbewegungsbild einer Menschenmenge machen kann. Man versucht hier die Ergebnisse beider Modelle zu kombinieren (Meyer, 2008; Kneidl, Hartmann, & Borrmann, 2011).

Wer genaueres über die einzelnen Modelle wissen möchte, erfährt dies in dem 2013 veröffentlichten Bericht *State-of-the-art crowd motion simulation models* von Duives, Daamen und Hoogendoorn. Im Folgenden geht es um den aus dem Projekt CrowdControl der Firma Siemens entstandenen Fußgängersimulator. Dieser wird auch in Kapitel 4 und 5 für die Evakuierungssimulation verwendet.

2.3 Fußgängersimulator CrowdControl

CrowdControl ist ein Projekt der Firma Siemens in Kooperation mit CMS. Ziel ist es, einen Fußgängersimulator zu entwickeln, der in Echtzeit möglichst originalgetreu und mit wenig Rechenleistung Bewegungen großer Menschenmengen simulieren kann (Wohllaib, 2009).

Simulationsmodell

Verwendet wird ein Zellulärer Automat, der mit einem Sechseckgitter arbeitet. Jedes dieser Zellen kann entweder mit einer Person gefüllt sein, leer sein oder als Hindernis definiert werden. Eine Wand ist zum Beispiel eine Kette von mehreren als Hindernis definierter Zellen. Das diskrete Modell berechnet die Positionen der Fußgänger nach jedem Zeitschritt neu. Dazu wird für jede Person eine Geschwindigkeit festgelegt, da schnellere Fußgänger beim Übergang zwischen zwei Zellen bevorzugt werden (Köster, Hartmann, & Klein, 2011).

Die Berechnungsregeln für die Personenübergänge sind an elektrodynamische Formeln angelehnt. Das Ziel wird als anziehendes Potential mit einer großen Reichweite definiert. Das Potential wird größer, je näher eine Person an ihr Ziel kommt. Andere Fußgänger oder Hindernisse werden als abstoßende Potentiale mit geringer Reichweite festgelegt. Auch hier haben nahe liegende Fußgänger eine größere abstoßende Wirkung, als weiter entfernte Personen. (Köster, Hartmann, & Klein, 2011).

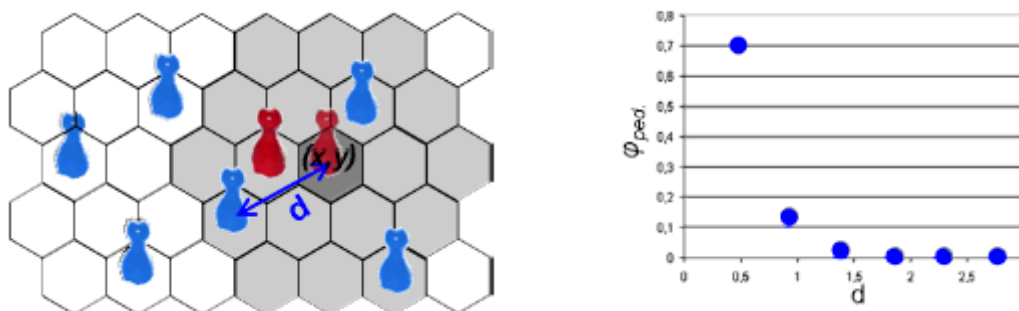


Abbildung 2.4: Das Schema zeigt die Berechnung des abstoßenden Potentials der umliegenden Fußgänger auf den Fußgänger an der Stelle (x,y) in Abhängigkeit der Entfernung. Es werden nur Fußgänger in der Nähe (grauer Bereich) beachtet. Die möglichen Potentiale können in der rechten Tabelle abgelesen werden (Köster, Hartmann, & Klein, 2011).

Als Berechnungsalgorithmus für den Weg dient die „Fast Marching Method“, die Methode des „schnellen Marschierens“. Darin geht man davon aus, dass die Personen an der Front einer Menschenmenge in ihrer Normalgeschwindigkeit gehen und erst die späteren durch andere Fußgänger abgebremst werden (Köster, Hartmann, & Klein, 2011).

Die Berechnung des Modells funktioniert ähnlich, wie beim Dijkstra-Algorithmus. Jede Zelle stellt einen Knoten dar, der durch Kanten mit den anderen Zellen verbunden ist. Nach jedem Zeitschritt werden die Kantengewichte neu berechnet und es kann ein optimaler Weg bestimmt werden (Dörries, 2013).

Problem dieses Modells ist, dass die Wege der Fußgänger an die Zellen gebunden sind und sie so nie den direkten Weg gehen können. Die Folge ist, dass sie mehr Zeit für die Bewältigung einer Strecke beanspruchen. Solche Effekte können jedoch weitgehend ausgeglichen werden, indem man die Geschwindigkeiten der Personen entsprechend erhöht.

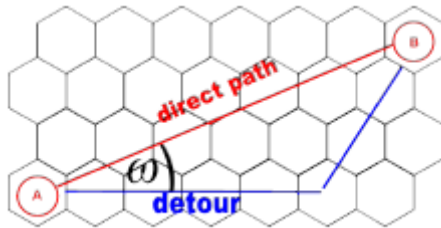


Abbildung 2.5: Die rote Linie wäre der kürzeste Weg. Mit Gitter ist jedoch die blaue Linie der kürzeste Weg (Köster, Hartmann, & Klein, 2011).

Mit diesen Verbesserungen kann man Probleme von Zellulären Automaten, die durch die Aufteilung in Raum und Zeit entstehen, weitgehend reduzieren. Das Ergebnis ist ein sehr effizientes Modell, das auch Phänomene der Selbstorganisation miteinbeziehen kann. Verglichen mit dynamisch arbeitenden Modellen benötigt es nur einen Bruchteil der Rechenleistung und das Ergebnis ist bei großen Flächen von vergleichbarer Qualität. Der Simulator schafft es so mit einem handelsüblichen Rechner bis zu 50.000 Fußgänger in Echtzeit zu simulieren (Duives, Daamen, & Hoogendoorn, 2013; Köster, Hartmann, & Klein, 2011).

Durch diese Eigenschaften ist das Modell sehr gut für die Erprobung einer Evakuierungssimulation am Beispiel der Station Marienhof in München geeignet. Dabei ist jedoch zu beachten, dass die benötigte Rechenleistung stark von den vorhandenen Hindernissen abhängt. Zwei sich gegenseitig behindernde Menschenströme benötigen ein Vielfaches der Rechenleistung von zwei sich nicht behindernden Menschenströmen.

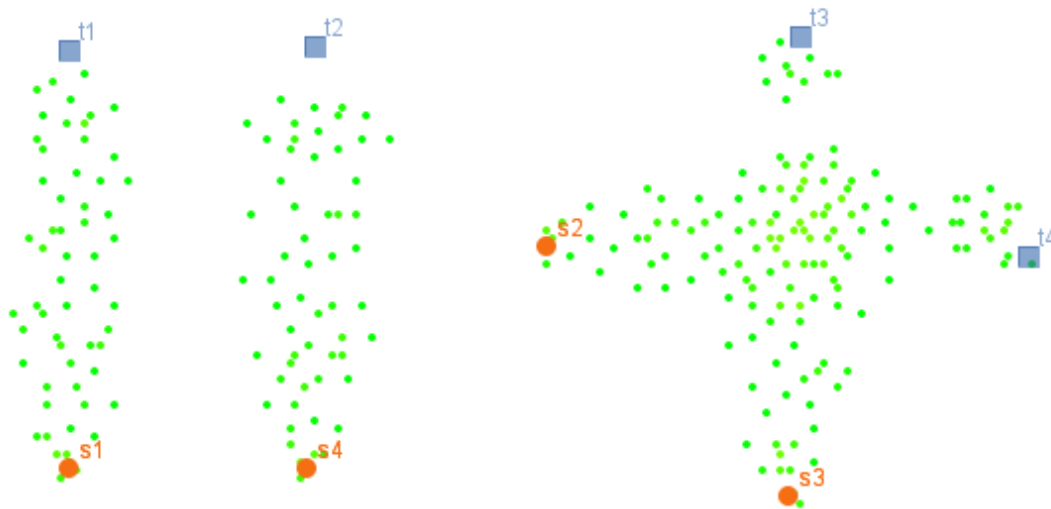


Abbildung 2.6: Die Abbildung zeigt zwei sich nicht behindernde Personenströme auf der linken Seite und zwei sich gegenseitig behindernde Personenströme auf der rechten Seite.

Anwendung des Simulators

Um den Simulator zu benutzen, erstellt man erst die Infrastruktur bzw. die Umgebung der Menschenmenge. Diese wird in Form von Hindernissen dargestellt. Sie können als Punkte, Flächen oder Wände (Linien) eingefügt werden. Anschließend kann man die Quellen und Ziele für die Fußgänger einzeichnen. Diese können Punkte oder Flächen sein. Im Falle einer Evakuierung der Station Marienhof können die Türen einer S-Bahn als Punkte und die auf die S-Bahn wartende Menschenmenge als Fläche betrachtet werden. Sobald die Quellen und Ziele im Simulator sind, kann man diese beliebig verbinden.

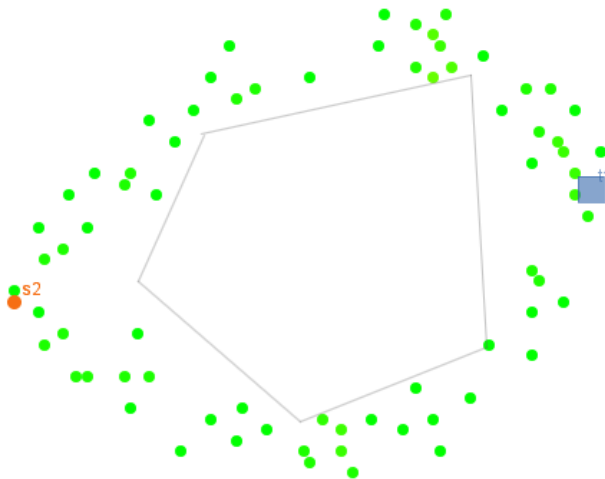


Abbildung 2.7: Die grauen Linien in der Mitte sind Wände, der rote Punkt eine Quelle und das blaue Quadrat ein Ziel. Quelle und Ziel wurden miteinander verbunden. Die kleinen Punkte stellen Personen dar, die von der Quelle zum Ziel gehen. (Screenshot aus dem Fußgängersimulator CrowdControl)

Wichtig für die Simulation ist, dass Quellen mit Zielen verbunden sein müssen, damit sich Fußgänger bewegen. Die Quellen können auch noch genauer definiert werden. Beispielsweise kann angegeben werden, wie viele Personen pro Sekunde eine Quelle verlassen oder in welchem Zeitraum Personen aus der Quelle kommen. Außerdem kann man bei mehreren Zielen angeben, wie viele Fußgänger zu welchem Ziel wollen. Die Fußgänger werden während der Simulation bei geringer Dichte grün und bei höheren Dichten gelb oder rot dargestellt. So kann man Problemstellen schnell finden und beheben. Nach der Simulation wird eine Textdatei erzeugt, in der die Ergebnisse abgespeichert werden. Dort befinden sich Parameter der Simulation, geometrische Beschreibungen, Wege der einzelnen Personen und die für die Simulation benötigte Prozessorzeit. Diese Textdatei wird zur 2D- und 3D-Nachbearbeitung verwendet (Barti, Crowd Control - Pedestrian Stream Simulation - Tutorial, 2010). Neben dieser „Ergebnisdatei“ werden jedoch auch die Ausgangsdaten für die Simulation gespeichert. Diese befinden sich in einem XML-Dokument. Mehr Informationen dazu gibt es im nächsten Unterpunkt.

Datenformat XML

Die Daten, wie Quellen, Ziele, Wände, etc. werden von dem Simulator im XML-Format gespeichert. Ausgeschrieben steht XML für Extensible Markup Language – eine erweiterbare Auszeichnungssprache für den Austausch von Daten. Diese dient dazu, hierarchisch strukturierte Daten in Form eines Textes darzustellen (Bray, 2006).

Eine XML-Datei enthält Elemente, die wiederum Elemente und Attribute enthalten können. Attribute sind die Eigenschaften von Elementen. Beispielsweise enthält ein Element vom Typ Wand ein Element vom Typ Punkt und die Attribute Wandname und Wandhöhe. Das Element vom Typ Punkt enthält wiederum die Elemente vom Typ x und y für die Lage im kartesischen Koordinatensystem. Man kann in das Element Wand mehrere Punkte einfügen und durch

deren Lage und Reihenfolge die Form der Wand festlegen. Der Vorteil solcher Austauschsprachen ist, dass sie für beliebig viele Fälle einsetzbar sind. Man kann so beliebig aussehende Wände in die hierarchische Struktur abspeichern. Ein Beispiel eines solchen ausgefüllten Textes im XML-Format befindet sich in der Anlage A.1. Dabei handelt es sich um ein für die Veranschaulichung sehr vereinfachtes Simulationsbeispiel mit einer Quelle, einem Ziel, deren Verbindung und einer Wand.

Für eine realistische Evakuierungssimulation ist eine gute Datengrundlage nötig. Das gilt sowohl für die Daten bezüglich der Anzahl und Wege der Fußgänger, als auch für die Infrastruktur. Hält man sich bei der Verteilung der Fußgänger an die Fluchtwege, ist die Simulation vergleichsweise einfach durchzuführen. Bezieht man jedoch auch mögliche Explosionen (Rauch, Druck) und versperrte Fluchtwege mit ein, wird die Simulation sehr aufwendig.

Diese Arbeit legt den Schwerpunkt auf die Infrastruktur. Da es in heutigen Planungsbüros üblich ist mit 2D-Plänen zu arbeiten, ist es eine einfache Möglichkeit, diese bereits vorhandenen digitalen Pläne (Gebäude bzw. Wege und Plätze) in ein Simulationsprogramm zu laden. So kann man relativ schnell reelle Szenarien für Evakuierungssituationen entwickeln und so Engpässe schon bei der Planung vermeiden. Dazu gibt es im nächsten Kapitel eine Einführung in den Planungsprozess von Bauvorhaben und die dazu verwendeten Zeichenprogramme.

3 Planungsprozess von Bauvorhaben

Als Planungsprozess von Bauvorhaben bezeichnet man das gedankliche Entwickeln und Gestalten von Bauwerken. Die Ergebnisse werden in Form von Plänen aufgezeichnet. Der Detaillierungsgrad dieser Pläne ist abhängig von der aktuellen Bauphase. Die neun Leistungsphasen sind in der HOAI §3 (4) festgelegt (Dr. Ludewig & Dr. Schoser, 2009):

1. Grundlagenermittlung
2. Vorplanung
3. Entwurfsplanung
4. Genehmigungsplanung
5. Ausführungsplanung
6. Vorbereitung der Vergabe
7. Mitwirkung bei der Vergabe
8. Objektüberwachung (Bauüberwachung oder Bauoberleitung)
9. Objektbetreuung und Dokumentation

Der Detaillierungsgrad der Pläne steigt hier mit ansteigender Leistungsphase. In der Grundlagenplanung gibt es Bestandspläne und Standortpläne, in der Vorplanung Lagepläne, in der Entwurfsplanung einen Gesamtentwurf und in der Genehmigungsplanung Grundrisse, Ansichten, Schnitte etc. Danach, in der Ausführungsplanung, werden bereits Schalungs-, Bewehrungs-, Konstruktions- und Detailpläne angefertigt. Es wäre also sinnvoll, eine Evakuierungssimulation in die Leistungsphasen eins bis vier einzubinden. Je später eine Änderung der Pläne vorgenommen wird, desto mehr Arbeit ist auch mit dieser Änderung verbunden. Man müsste also während der Grundlagenplanung und Vorplanung Daten für Fußgängerzahlen und Wege ermitteln. Diese könnte man dann in der Entwurfs- und Genehmigungsplanung bei der Entwicklung des Gesamtentwurfs, der Grundrisse und Positionspläne miteinbeziehen und anschließend mit einer Simulation prüfen. Diese Pläne liegen in aller Regel als digitale 2D-Pläne vor. Diese können mit Zeichenprogrammen geöffnet und bearbeitet werden. Im folgenden Unterpunkt gibt es daher einen Einblick in die Nutzung von Zeichenprogrammen im Planungsprozess.

3.1 Zeichenprogramme (CAD) im Planungsprozess

Als CAD (Computer Aided Design) bezeichnet man den Entwurf und die Konstruktion von Produkten durch Computerprogramme. Es handelt sich dabei um die Digitalisierung des technischen Zeichnens. Mittlerweile werden die Softwareprodukte aber durch Einbindung von Planungs-, Berechnungs- und Informationssystemen immer komplexer. Dazu gehören zum Beispiel 3D-Darstellung, Assoziationen, vorgefertigte Bauteilobjekte und die daraus gewonnenen Möglichkeiten bezüglich Analyse und Simulation (Rank, Teil II: Ingenieuranwendungen, 2010).

Da Planungsbüros in der Regel noch mit konventionellen CAD-Systemen arbeiten, wird in dieser Arbeit der Fokus auch hier liegen. Also bei 2D-Software, die noch nahe an das analoge

technische Zeichnungen angelehnt ist. Dabei darf jedoch nicht das Potential von BIM (Building Information Modeling) für Simulationen außer Acht gelassen werden. Ziel dieser Arbeit ist es aber, die Evakuierungssimulation bestmöglich auf Basis der CAD-Pläne aus der Entwurfs- und Genehmigungsplanung durchzuführen. Dazu gibt es im nachfolgenden Teil einen Einblick in die Welt von CAD.

CAD-Systeme

CAD-Systeme können in fachspezifische Systeme, fachunabhängige Systeme, und Grafiksysteme (Kernsysteme) unterteilt werden (Rank, Teil II: Ingenieur Anwendungen, 2010).

Fachspezifische CAD-Systeme zeichnen sich dadurch aus, dass sie speziell für eine Fachrichtung entwickelt wurden. Also beispielsweise speziell für Bauingenieurwesen oder Architektur. Dazu gehören etwa SOFICAD von Sofistik, ALLPLAN/ALLPLOT von Nemetschek oder ArchiCAD von Graphisoft. In solchen Systemen sind bereits einige fachspezifische Elemente, wie Fenster oder Türen, vorgefertigt.

Fachunabhängige CAD-Systeme sind für mehrere Fachrichtungen konzipiert, also zum Beispiel neben Architektur auch für Maschinenbau oder Elektrotechnik. Dazu gehören AutoCAD von Autodesk und MICROSTATION von Bentley. In diesem Projekt wird das Programm AutoCAD Architecture verwendet. Das ist ein fachspezifischer Aufsatz für AutoCAD, der eine Vereinfachung zur Anfertigung von Bauplänen zum Ziel hat. Im Gegensatz zu fachspezifischen Systemen ist der Aufsatz hier aber erst nachträglich als Erweiterung entwickelt worden. Vorteil ist hier, dass das im Rahmen dieser Arbeit entwickelte Plug-In auch für die Basisversion von AutoCAD oder für einen anderen Programmaufsatz verwendet werden könnte.

Grafiksysteme (Kernsysteme) fungieren als Grundlage von CAD-Systemen. Sie bestehen aus mehreren Bibliotheken, die zur graphischen Bildverarbeitung genutzt werden können. Diese sind genormt und können länder-, betriebssystem-, und maschinenübergreifend verwendet werden. Sie sind schon bestehende Softwarebausteine und dienen als Vereinfachung bei der Entwicklung neuer CAD-Systeme. Beispiele sind HOOPS von TechSoft oder PHIGS, ein ANSI/ISO Standard zur Darstellung von 3D-Objekten.

Zusätzlich benötigt man noch graphische Grundfunktionen, die die Visualisierungssoftware mit dem Gerätetreiber verbinden und den Gerätetreiber (z.B. Grafikkarte), der die Verbindung zwischen Hard- und Software herstellt (Rank, Teil II: Ingenieur Anwendungen, 2010).

Dies stellt stark vereinfacht den allgemeinen Aufbau von CAD-Systemen dar. Er ist die Grundlage für die vielen Werkzeuge, die und heute durch CAD-Systeme zur Verfügung stehen. Dazu kommt im folgenden Teil eine kurze Zusammenfassung über die verschiedenen CAD-Funktionen.

CAD-Grundfunktionen

Die Basisfunktionen werden im Allgemeinen in Zeichenfunktionen, Konstruktionsfunktionen und Hilfsfunktionen unterteilt. Als Zeichenfunktionen im zweidimensionalen Raum bezeichnet man die Erstellung von Punkten, Linien, Rechtecken, Kreisen, Ellipsen, Bögen, Freiformkurven oder Texten.

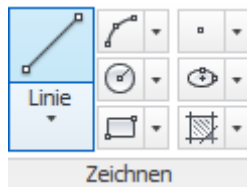


Abbildung 3.1: Zeichenfunktionen in Autodesk AutoCAD

Von Konstruktionsfunktionen können die graphischen Primitive dann weiter bearbeitet werden. Dazu gehören die Funktionen Löschen, Verschieben, Drehen, Kopieren, Spiegeln, Verzerren, Anfasen, Boolesche Operationen, Sweeping, Dehnen/Stutzen oder Versetzen.

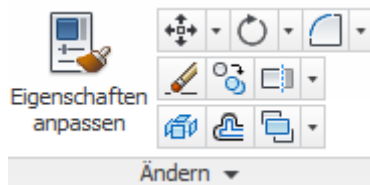


Abbildung 3.2: Konstruktionsfunktionen in Autodesk AutoCAD

Ergänzend gibt es noch einige Hilfsfunktionen, wie Messen, Zoomen, Objektfang, Raster, Orthogonalmodus oder die Linealfunktion. Diese dienen als Unterstützung für Arbeiten des Planers. All diese Funktionen können auch als Basis für komplexere Funktionen verwendet werden. Dazu gibt es Näheres im nächsten Unterpunkt (Rank, Teil II: Ingenieurwendungen, 2010).

Struktur eines CAD-Modells

Das Zeichnen eines CAD-Plans kann durch Zusammenfassen von mehreren Objekten zu einem Objekt, durch Ableiten von neuen Objekten aus bereits vorhandenen Objekten und durch Verändern von bestehenden Objekten stark vereinfacht werden. Diese Methodik wird durch eine horizontale, vertikale und inhaltliche Strukturierung sowie durch Bemaßungen und Assoziationen unterstützt (Rank, Teil II: Ingenieurwendungen, 2010).

Im Laufe einer Bauplanung entstehen viele Grundrisse, Schnitte und Detailpläne eines Projekts. Je nachdem, wie diese Pläne geordnet sind, hat man eine andere horizontale Struktur beziehungsweise Planstruktur. Bei der Station Marienhof liegen Grundrisse der verschiedenen Ebenen und einige Längs- und Querschnitte vor. Dafür sind drei unterschiedliche Ordner angelegt. Sinnvoll wäre es zudem, die Grundrisse nach den Höhenkoordinaten und die Schnitte der Reihe nach angelehnt an die Himmelsrichtungen zu ordnen.

In CAD-Programmen kann man in der Regel Layer erstellen und anschließend in diesem Layer zeichnen. In der Literatur wird diese Form der Strukturierung auch häufig vertikale

Struktur oder Folienstruktur genannt. Es handelt sich um mehrere Folien, die man übereinander legen kann. Der Vorteil ist hier, dass man einzelne Folien entfernen und wieder zurücklegen kann. Zeichnet man in einem Layer kann man diesen aus- und einblenden oder für die Bearbeitung sperren. Es ist zum Beispiel sinnvoll, Wände in einem anderen Layer zu zeichnen als Sicherheitsstreifen oder Bewehrung. Zusätzlich hat man die Möglichkeit, für einen Layer bestimmte Eigenschaften, wie Farbe oder Linienstärke festzulegen. Das ist nützlich für die Übersichtlichkeit beim Betrachten der Pläne. Für die Evakuierungssimulation ist die Bewehrung beispielsweise unwichtig und kann ausgeblendet werden.

Von inhaltlicher Strukturierung spricht man bei der Anwendung einer Vorlage oder eines Makros. Vorlagen sind vorgefertigte Objekte in der CAD-Bibliothek, die sich in der Planung häufig wiederholen. Dazu gehören Symbole, Muster, Schablonen, Planköpfe, Firmenzeichen, Schraffuren oder voreingestellte Parameter eines Plans wie Maßstab oder Maßeinheit. So müssen Schraffuren, etwa für Beton oder Holz, nicht jedes Mal wieder neu gezeichnet werden. Vorlagen können auch vom Nutzer angefertigt und in einer Bibliothek abgespeichert werden. Von einem Makro spricht man, wenn eine Reihe von Basisfunktionen hintereinander als eine Funktion angewendet wird. Im Gegensatz dazu spricht bei einer Zusammenfassung mehrerer Graphikprimitive von Blöcken. Fügt man eine Tür als Block ein, kann man sie nur in die vorhandene Zeichnung einfügen, diese wird jedoch nicht verändert. Bei einem Makro gibt es die Möglichkeit, dass eine vorhandene Wand beim Einfügen einer Tür gleich unterbrochen wird. Es wird also erst eine Öffnung in der Wand erstellt und dann die Tür eingefügt. Die Zeichnung ist somit schon konsistent, ohne dass der Planer die Wand noch manuell verändern muss. Ein Block oder Makros kann vom Nutzer selbst erstellt und in einer Bibliothek gespeichert werden.

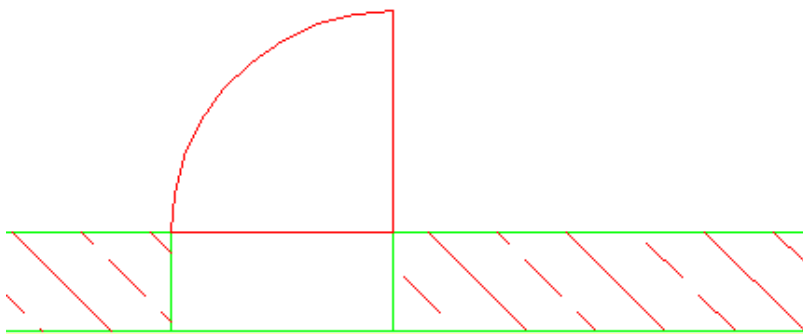


Abbildung 3.3: Screenshot einer Wand in Autodesk AutoCAD. Die Wand hat die Schraffur für Stahlbeton und beinhaltet eine Tür. Beim Einfügen der Tür wurde hier automatisch eine Aussparung für Wand und Schraffur eingefügt.

CAD-Software unterstützt den Planer beim Bemaßen der gezeichneten Objekte. Bei der Punktbemaßung muss der Nutzer nur die Punkte angeben, an denen bemaßt werden soll, und er kann dann noch den Abstand für die Beschriftung wählen. Der Rest wird automatisch von der Software übernommen. Ähnliches gilt für die Schnittbemaßung, bei der nur eine Linie und der Abstand für die Beschriftung angegeben werden muss. Das Programm bemaßt alle Punkte, an denen die Linie ein Objekt schneidet. Neben diesen Bemaßungen werden in CAD noch viele weitere Arten der Bemaßung angeboten.

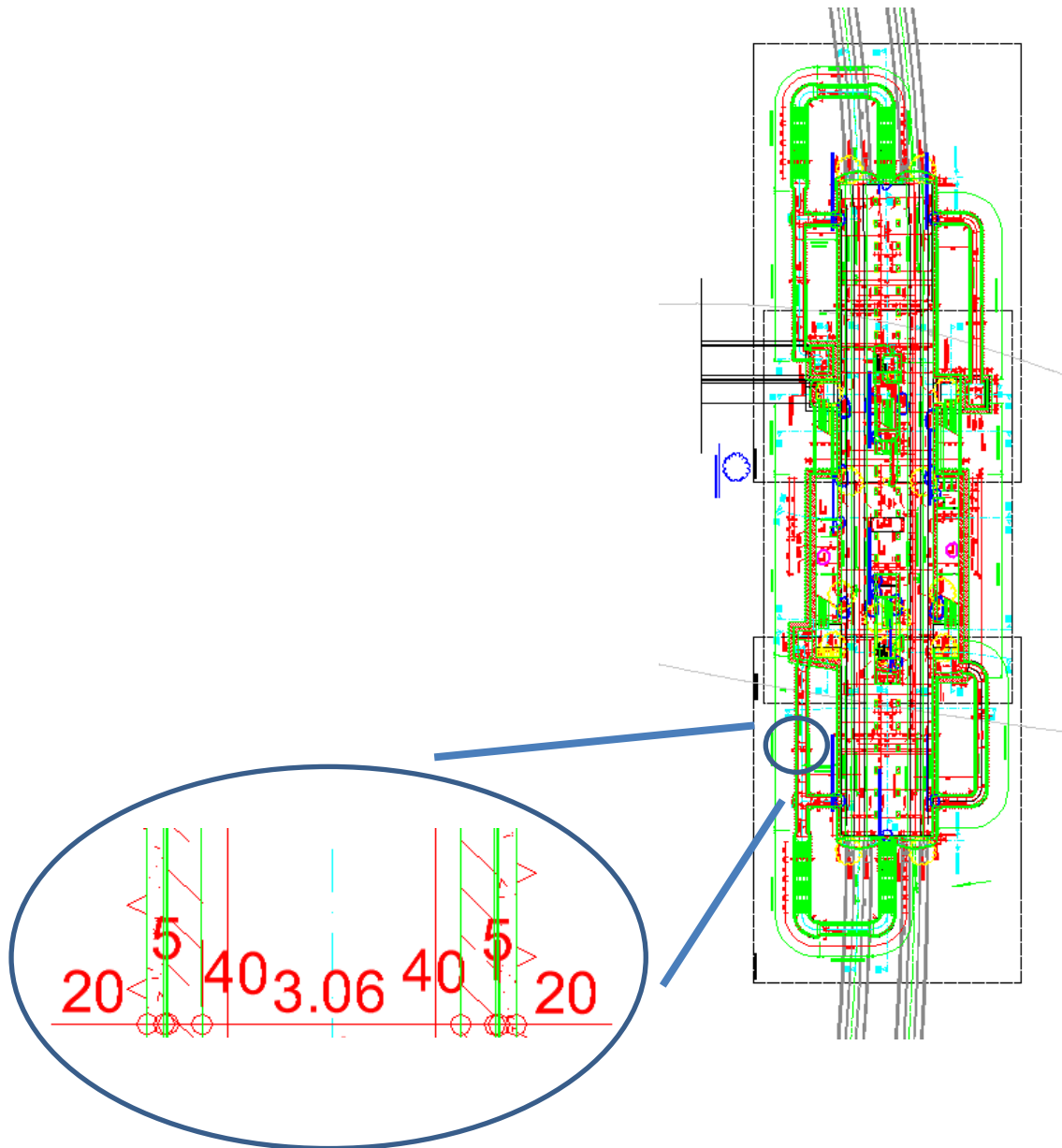


Abbildung 3.4: Schnittbemaßung eines Fußgängertunnels aus Autodesk AutoCAD

Als letzte Art der Unterstützung gibt es noch Assoziationen. Darunter versteht man die Verknüpfung von Elementen. Als Beispiel dient die Tür mit der eingefügten Wand. Verdreht man die Wand ohne Verknüpfung zwischen Tür und Wand, bleibt die Tür an ihrer Stelle und die Wand weist eine Lücke auf. Verknüpft man jedoch diese beiden Objekte, kann man die Wand beliebig verdrehen und verschieben. Die Tür bleibt an der Stelle in der Wand.

CAD-Software ist ein zentraler Bestandteil in der Entwicklung von Bauprojekten. Im nächsten Schritt werden die CAD-Pläne der Station Marienhof genauer betrachtet.

3.2 CAD-Pläne der Station Marienhof

Bei den Plänen der Station Marienhof handelt es sich um Ergebnisse aus der Genehmigungsplanung. Sie stammen vom Planungsbüro Obermeyer aus München, das den Zuschlag für dieses Projekt erhalten hat. Angefertigt wurden die Pläne mit dem Programm AutoCAD von Autodesk und liegen daher im DWG-Format vor. Das Bauvorhaben ist durch die 40 Meter Tiefe, die fünf Ebenen, den Anschluss an den Bestand und die vielen Tunnel sehr komplex. Eine gute Planstruktur ist daher für die Übersichtlichkeit von zentraler Bedeutung. Es sind insgesamt sieben Grundrisse, bestehend aus den fünf Ebenen Oberfläche, Sperrgeschoss, Treppengeschoss, Verteilerebene und Bahnsteigebenen und zwei Detailplänen der Bahnsteigebenen, vorhanden. Zusätzlich gibt es elf Querschnitte und elf Längsschnitte. Der Gesamtplan enthält also insgesamt 29 Pläne.

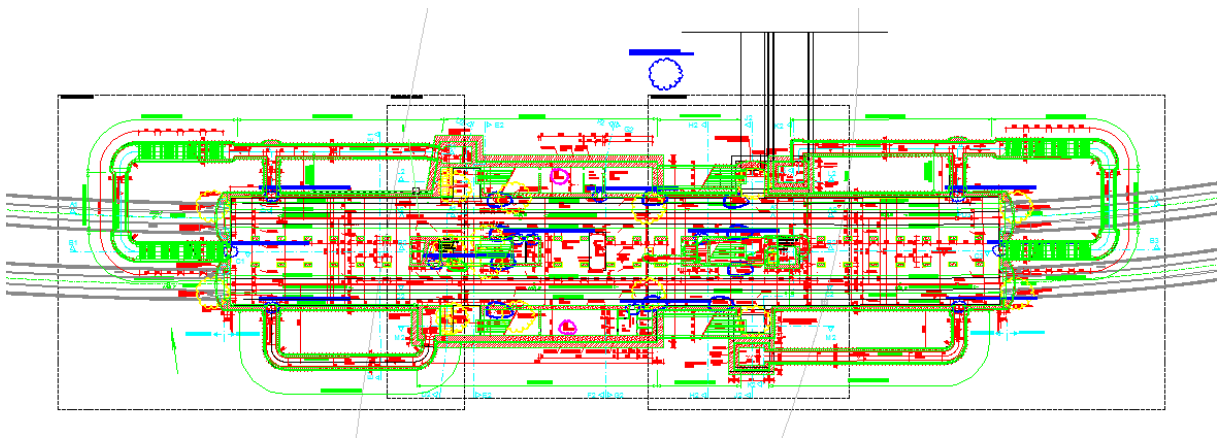


Abbildung 3.5 Screenshot aus dem Programm AutoCAD Architecture von Autodesk. Bei dem Plan handelt es sich um den Grundriss der Bahnsteigebenen für die neu geplante Station Marienhof des Planungsbüros Obermeyer.

Durch die große Anzahl an Plänen ist eine gut geführte Planstruktur von zentraler Bedeutung. Zusätzlich ist durch die vielen Informationen in dem Plan eine Folienstruktur sehr wichtig. So kann man die für eine Evakuierungssimulation unwichtigen Teile, wie Bewehrung oder Schnitte einfach ausblenden.

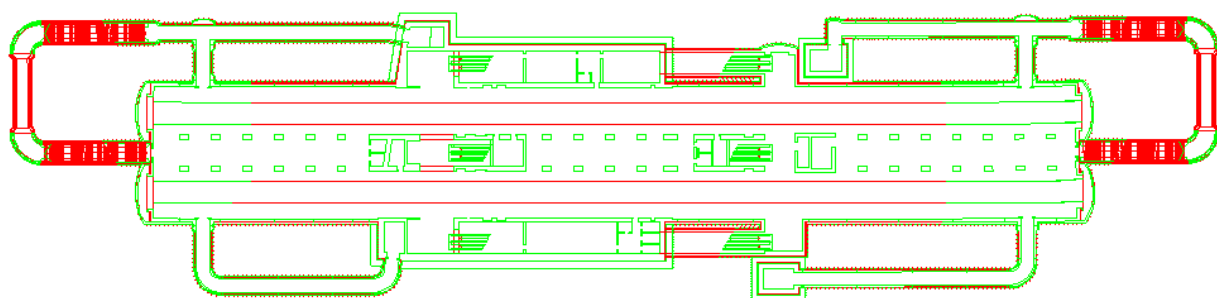


Abbildung 3.6: Bei diesem Screenshot sind alle für eine Fußgängersimulation unwichtigen Layer abgeschaltet.

Man sieht an diesem Beispiel sehr schön, dass die Folienstruktur einen großen Teil zur Übersichtlichkeit der Pläne beiträgt. Für die spätere Übertragung in den Fußgängersimulator wird nun das DWG-Datenformat genauer erklärt.

Datenformat DWG

DWG ist weltweit eines der meist verwendeten Formate im Bereich der Konstruktion. Es enthält alle Daten, die beim Arbeiten mit AutoCAD entstehen und benötigt werden. Dazu gehören Entwürfe, geometrische Daten, Karten oder Fotos. 1982 wurde DWG gemeinsam mit dem Programm AutoCAD von Autodesk herausgebracht (Autodesk, 2013).

Anzumerken ist hier, dass die Dokumentation der Dateistruktur nicht frei erhältlich ist. Das Datenformat wird hauptsächlich von Autodesk und ein paar kleineren Softwareherstellern genutzt. Man kann hier nicht einfach, wie bei einer XML-Datei, den Aufbau der Datei anschauen und die Daten herauskopieren und verändern. Zugriff zu den Daten erhält man nur über CAD-Software, wie AutoCAD, oder über eine vom Hersteller bereitgestellte Programmierschnittstelle. Diese Programmierschnittstelle erlaubt es, Daten aus der DWG-Datei auszulesen und in eine XML-Datei zu exportieren. Um eine Evakuierungssimulation für den Nutzer möglichst einfach zu gestalten kam daher die Idee, eine Benutzeranwendung zu entwickeln, mit der man XML-Dateien für die Evakuierungssimulationen direkt aus AutoCAD exportieren kann. Das hat den Vorteil, dass man dem Nutzer während des Exportvorgangs Hilfestellungen geben kann und nicht den bestehenden Plan ändern muss. Genaueres wird im nächsten Kapitel erörtert.

4 Programmzusatz (Plug-In) zur vereinfachten Datenübertragung

Für einen Programmzusatz, mit dem man Daten aus AutoCAD in eine XML-Datei exportieren kann, benötigt man eine Software, mit der man gewünschte Objekte auswählen und diese anschließend in eine XML-Datei übersetzen kann. Hier ist es besonders wichtig, dass die XML-Datei genau dem Format entspricht, das von dem Fußgängersimulator CrowdControl verwendet wird. Ansonsten kommt es beim Öffnen der Datei zu Fehlermeldungen und es kann keine Simulation durchgeführt werden. Da es eine solche Anwendung in AutoCAD standardmäßig nicht gibt, ist es notwendig einen Programmzusatz zu entwickeln. Für diesen Programmzusatz benötigt man Autodesk AutoCAD, eine integrierte Entwicklungsumgebung zum Schreiben des Quellcodes, eine Programmierschnittstelle für den Zugriff auf die Geometriedaten, einen Softwareentwicklungskasten mit vielen schon bestehenden Werkzeugen und einen Assistenten zum Verknüpfen der Entwicklungsumgebung mit AutoCAD (Autodesk, 2013). Um die Daten in eine XML-Datei zu exportieren, wird eine Objektstruktur erstellt. Genauer zu dem Programmzusatz wird in den folgenden Unterpunkten beschrieben.

4.1 Programmzusatz (Plug-In) für AutoCAD

Ein Plug-In ist als Programmzusatz zu verstehen, der die Funktionen einer Basissoftware erweitert und nicht als eigenständiges Programm verwendet werden kann. Man hat hier entweder die Möglichkeit, auf vorhandene Bibliotheken des Basisprogramms, bestehend aus Unterprogrammen und Lösungsmöglichkeiten, zurückzugreifen oder externe Bibliotheken zu verwenden. Dadurch können die Funktionen des Programms beliebig erweitert werden (Lackes, Siepermann, & Kollmann, 2013). Im Folgenden wird für die Erzeugung dieser Funktionen als Hilfsmittel die Entwicklungsumgebung Microsoft Visual Studio verwendet.

Entwicklungsumgebung Microsoft Visual Studio

Visual Studio ist eine integrierte Entwicklungsumgebung. Hier wird ein Programm in einer Programmiersprache geschrieben und dann von Visual Studio in ein ausführbares Programm übersetzt (Rank, Teil III: Softwareentwicklung mit Visual Basic.NET, 2010) - in diesem Fall in eine dll-Datei.

Der Vorteil solcher Compiler ist, dass sie Fehler anzeigen und man Schritt für Schritt durch das Programm debuggen kann. Die Werte der einzelnen Variablen sind so nach jedem Teilschritt ablesbar. Das kann beim Programmieren sehr hilfreich sein. Als Programmiersprache wird hier VisualBasic.NET verwendet. Dies ist eine sehr anschauliche Programmiersprache, die häufig für Windows-Anwendungen verwendet wird. Sie ist eine eigenständige und objektorientierte Programmiersprache für das Microsoft.NET Framework. Dieses Framework ist eine Softwaretechnologie, um verschiedene Dienste und Anwendungen

auf einer Plattform zu verbinden (Rank, Teil III: Softwareentwicklung mit Visual Basic.NET, 2010). Um im Quellcode auf die Bibliotheken der Basissoftware, hier AutoCAD, zugreifen zu können, werden von einigen Herstellern Programmierschnittstellen zur Verfügung gestellt.

Programmierschnittstelle (API)

Eine API (Application Programming Interface) ist eine Programmierschnittstelle, die es externer Software ermöglicht, an das Programm anzubinden. Das Ganze geschieht auf Quelltextebene. Hiermit hat der Programmierer Zugriff auf Datenmodelle, Hardware oder die grafische Benutzeroberfläche eines Programms. Allgemein bezeichnet man den Zugriff auf eine Bibliothek als Programmierschnittstelle (3scale NetWorks S.L., 2011). Eine Bibliothek ist hier eine thematisch zusammenhängende Sammlung von Unterprogrammen oder Lösungswegen (Duden, 2013). Diese bestehen aus Wertedefinitionen, Deklarationen, Funktionen, Klassen, Namensräumen, etc.

Die Bibliotheken werden als Verweis in das Visual-Studio-Projekt eingefügt. Anschließend können die einzelnen Namensräume in den Quellcode importiert werden. Die Bibliotheken sind durch das Betriebssystem oder die Installation des zu erweiternden Programms meist schon auf dem Computer vorhanden, da die installierten Programme selbst auf diese Bibliotheken zugreifen. Um den Zugriff auf Bibliotheken für Entwickler zu vereinfachen, werden häufig Softwareentwicklungskästen zur Verfügung gestellt.

Softwareentwicklungskasten (SDK)

In der SDK (Software Development Kit) befinden sich verschiedene Werkzeuge, Anwendungen und Dokumentationen, die den Entwicklern bei der Arbeit behilflich sein sollen (Doberenz & Gewinnus, 2013).

Häufig werden diese von Herstellern der Plattformen oder Geräten zum freien Download bereitgestellt. Autodesk stellt für AutoCAD SDKs der gängigsten Programmiersprachen zur Verfügung. Der Entwickler kann hier Quellcodes besonders wichtiger Programmwerkzeuge nachlesen und diese selbst verwenden und weiterentwickeln. Die von Autodesk kostenlos zur Verfügung gestellte SDK für AutoCAD kann von der Internetseite unter dem Namen ObjectArx heruntergeladen werden.

In der ObjectArx befinden sich Beispiele, wie man Objekte (Kreise, Linien) erstellt, diese verändert, Informationen vom Nutzer bekommt, AutoCAD Befehle erzeugt und ausführt oder dem Nutzer Hilfestellungen bei der Ausführung von Funktionen gibt. Dies kann bei der Programmierung sehr nützlich sein. Auch die meisten Funktionen des Plug-Ins dieser Arbeit beruhen auf Beispielen der ObjectArx von Autodesk. Als zusätzliches Hilfsmittel bei der Entwicklung eines Plug-Ins dient ein Assistent.

Assistent (Wizard)

Der Begriff Wizard bedeutet Zauberer. Es ist ein Assistent, der den Nutzer von Software durch Hilfestellungen, beispielsweise beim Ausfüllen von Formularen oder bei der Installation von Programmen, unterstützt. Dabei wird eine interaktive Oberfläche genutzt, um dem Nutzer Schritt für Schritt durch mehrere Dialoge eine ergonomische Dateneingabe zu ermöglichen (Doberenz & Gewinnus, 2013). Dies macht die Arbeit benutzerfreundlicher und effizienter.

Autodesk stellt dazu den AutoCAD .NET Wizard kostenlos zur Verfügung. Dieser kann als Vorlage mit Visual Studio geöffnet werden. Darin sind einige Verweise zu Bibliotheken und die Verknüpfung von Visual Studio mit AutoCAD bereits voreingestellt. Es sind also einige Einstellungen, die zur Entwicklung eines AutoCAD-Plug-Ins notwendig sind, bereits vorgefertigt. Findet das Programm notwendige Verweispfade nicht automatisch, wird der Nutzer dazu aufgefordert, die Pfade anzugeben. Möchte man jedoch auf Bibliotheken zurückgreifen, die von dem weiterzuentwickelnden Programm unabhängig sind, muss man die Verweise zu den Bibliotheken eigenständig, also ohne Assistenten, einfügen. Ein Beispiel hierfür ist der XML-Serializer im nächsten Unterpunkt.

4.2 XML-Serializer

Hiermit konvertiert man die Eigenschaften und Felder eines Objekts mit Hilfe einer Objektstruktur in ein serielles Format (hier XML). Ziel ist die Speicherung und Übertragung von Daten. Die zu speichernden Daten werden im Quellcode als Klassen, Felder, Eigenschaften, primitive Typen, Arrays, Xml-Element-Objekte oder Xml-Attribut-Objekte beschrieben (Microsoft, 2013).

Um den XML-Serializer zu verwenden benötigt man zuerst einen Verweis zur Bibliothek System.Xml (in system.xml.dll) und muss im Quellcode den Namensraum System.Xml.Serialization importieren. Diese Bibliothek ist auf allen Windows Rechnern standardmäßig vorinstalliert. Hat man den Namensraum importiert, erhält man Zugriff auf sämtliche Werkzeuge der XML-Serialisierung.

Um eine Serialisierung durchzuführen muss man im Quellcode zuerst eine Klassenstruktur entwickeln, die der Struktur der zu erstellenden XML-Datei genau entspricht. Jedes Element wird darin als Klasse dargestellt, welche wieder Elemente anderer Klassen enthalten kann. Die Elemente einer Klasse können auch als Array deklariert werden. Dadurch wird beispielsweise gewährleistet, dass eine Wand durch beliebig viele Punkte in einer Reihe definiert werden kann. Die Mindestanzahl beträgt jedoch zwei Punkte pro Wand. Außerdem muss man darauf achten, dass in den Klassen zwischen normalen Elementen und Attributen zu unterscheiden ist. Um eine deklarierte Variable als Attribut auszuweisen benötigt man einen extra Hinweis.

Anschließend kann man Variablen als diese Klassen deklarieren und sie mit Daten versehen. Zusätzlich sind noch Verweise zwischen den deklarierten Variablen nötig. Also, ein Verweis, dass eine Variable „Punkt“ Bestandteil einer Variable „Wand“ ist. Sind die Variablen bestückt

und die Verweise erstellt, kann man die Serialisierung starten. Man serialisiert dann zum Beispiel die Variable, die als die oberste Klasse in der Klassenhierarchie deklariert ist. Ist diese Variable eine Wand, werden alle Punkte und Koordinaten mit in das XML-Dokument eingefügt. Serialisiert man hingegen einen Punkt, wird nur dieser Punkt mit seinen Koordinaten serialisiert.

Dies waren die wichtigsten Punkte zur Erstellung eines AutoCAD Plug-Ins für die Übertragung von Daten aus AutoCAD in ein XML-Dokument. Da das Ziel dieser Arbeit jedoch ein möglichst benutzerfreundliches Programm ist, soll noch eine grafische Benutzeroberfläche, zur einfacheren Bedienung für den Nutzer in AutoCAD, erstellt werden. Ansonsten müsste der Planer die benötigten Befehle auswendig wissen und unten in der Befehlsleiste von AutoCAD eingeben. Um das zu verhindern, kann man in AutoCAD eine neue graphische Benutzeroberfläche erstellen bzw. diese erweitern.

4.3 Graphische Benutzeroberfläche (GUI)

Die GUI (Graphical User Interface) ist eine graphische Benutzeroberfläche zur benutzerfreundlichen Bedienung von Programmen. Sie ist Teil des Betriebssystems und vieler Programme. Durch sie können Programme, Dienste und Funktionen per Mausklick von Buttons aufgerufen werden. GUIs zeichnen sich dadurch aus, dass sie sehr übersichtlich aufgebaut sind, und sie können je nach Programm verschiedene Funktionsbereiche, Menüleisten, Symbolleisten, Funktionsleisten, Statusleisten, Taskleisten oder Bildlaufleisten enthalten (Dictionary, 2013).

In AutoCAD kann man so eine Benutzeroberfläche mit dem Befehl `_CUI` erstellen oder die vorhandene Multifunktionsleiste bearbeiten. Es können Registerkarten, Gruppen und Befehle mit einer Graphik erstellt werden.



Abbildung 4.1: Screenshot aus AutoCAD mit einer bearbeiteten Benutzeroberfläche. Neu sind hier die Registerkarte `Ped Sim` mit der Gruppe `Ped Sim` und die fünf Funktionen.

Ist die Benutzeroberfläche erst einmal erstellt, kann sie als `cuix`-Datei abgespeichert und zu einem späteren Zeitpunkt wieder geöffnet werden. Der große Vorteil solcher Benutzeroberflächen ist, dass sie in den meisten Programmen gleich aufgebaut sind und sich der Nutzer so leichter in den Aufbau und die Funktionen einfinden kann. Damit ein Benutzer dieses Plug-In verwenden kann, muss er unter dem Befehl `_CUI` diese Benutzeroberfläche einstellen und anschließend unter dem Befehl `_NETLOAD` die aus dem Compiler Visual Studio erstellte `dll`-Datei laden. Diese enthält die im Quellcode definierten Befehle mit den entsprechenden Funktionen.

Betätigt man die Befehle in der Benutzeroberfläche, ohne die dll zu laden, wird der Befehl zwar in der Benutzerleiste eingefügt, er kann jedoch nicht ausgeführt werden, da er nicht vorhanden ist. Man könnte jedoch die Befehle des Plug-Ins ausführen, ohne die neue Benutzeroberfläche zu laden, indem man die Befehle in der Befehlszeile eingibt.

Das waren die allgemeinen Rahmenbedingungen zum Erstellen eines Plug-Ins in AutoCAD, zum Erstellen eines XML-Serializers und zur Einbindung der erzeugten Befehle in die Benutzeroberfläche von AutoCAD. Nun kommt der wichtigste Teil des Plug-Ins, der Quellcode zum Erstellen der neuen Funktionen und Anwendungen.

4.4 Workflow für den Quellcode des Plug-Ins

Bei der Grundlagenermittlung für ein Plug-In, mit dem man Objekte aus AutoCAD in Form von Wänden in den Fußgängersimulator CrowdControl laden kann, wird relativ schnell klar, dass dazu drei Funktionen notwendig sind. Dies ist einmal die Funktion Area, mit der man den Bereich festlegen kann, also die Größe des später im Fußgängersimulator angezeigten Bereichs in Abhängigkeit der X- und Y-Achse. Als zweites wird die Funktion Wall benötigt, mit der man Objekte, die meist als Linien, Polylinien oder Bögen vorliegen, auswählen kann. Als Letztes benötigt man für eine Übertragung von Objekten aus AutoCAD in den Fußgängersimulator noch die Funktion ConvertToXML, mit der man die ausgewählten Daten in eine XML-Datei exportiert, die dann mit dem Fußgängersimulator geöffnet werden kann.

Zusätzlich soll es auch möglich sein, mit dem Plug-In Quellen und Ziele für die Fußgängerströme in den CAD-Plan einzufügen und zu übertragen. Dazu sind noch die Funktionen Source (Quelle) und Target (Ziel) notwendig.



Abbildung 4.2: Diese Abbildung zeigt von links nach rechts die Funktionen Area, Wall, Source, Target und ConvertToXML.

Im Folgenden werden die einzelnen Funktionen genauer erläutert. Dabei ist anzumerken, dass der Nutzer bei den Benutzereingaben vom Programm Hilfestellungen bekommt und die Funktionen bei längerem Verweilen mit der Maus auf dem Funktionsbutton nochmal genauer erklärt werden.

Area

Diese Funktion dient zur Festlegung des Bereichs für die Simulation. Als Erstes wird in jeder Funktion ein Befehl erstellt, so dass die Funktion in AutoCAD über die Benutzereingabe aufgerufen werden kann. Danach wird die Ansicht auf oben gestellt und parallel zur X-Achse ausgerichtet. In der Regel sind diese Eigenschaften voreingestellt. Ist dies jedoch nicht der

Fall, gäbe es ohne diese Anpassung Probleme bei der Berechnung der Punkte. Die Koordinatensysteme von Plan und Zeichenfenster könnten ansonsten ungleich sein. Um die Eigenschaften zu verändern muss man zuerst das aktuelle Datenmodell von AutoCAD öffnen und nach der Änderung wieder schließen.

Als Nächstes wird ein neuer Layer mit dem Namen „Bereich PedSim“ und der Farbe Gelb erstellt. Dazu bestimmt man das aktuelle Dokument und das aktuelle Datenmodell. Danach wird eine Transaktion gestartet und der Layertable (Tabelle mit Layern) zum Lesen geöffnet. Nun kann man Farbe und Namen des Layers festlegen. Danach stellt man den Layertable auf den Lesemodus und fügt den neuen Layer zur Transaktion und zum Layertable hinzu. Anschließend kann man den Blocktable (Tabelle mit Blöcken) zum Lesen öffnen und den Layertable über die Transaktion zum Blocktable hinzufügen. In einer DWG-Datei werden einzelne Elemente in Form von Blöcken zusammengefasst. Der Blocktable ist eine Tabelle aus mehreren solcher Blöcke. Der neue Layer ist jetzt in dem Datenmodell gespeichert und die Transaktion wird wieder beendet. Ist dies geschafft, müssen die Abmessungen des Bereichs bestimmt werden.

Diese werden mit der Prozedur `GetPointsFromUser` bestimmt. Darin wird der Nutzer nach dem ersten Punkt des Bereichs links unten und nach dem zweiten Punkt des Bereichs rechts oben gefragt. Diese Punkte kann der Nutzer per Mausklick in das Zeichenfenster festlegen. Nun wird, ähnlich wie bei der Erstellung des Layers, eine Transaktion gestartet und ein Blocktable erstellt, der auf den Lesemodus gestellt wird. Aus den zwei vom Benutzer angegebenen Punkten kann ein rechteckiges Polygon erstellt werden, was zum Blocktable hinzugefügt wird. Jetzt kann die Transaktion wieder geschlossen werden und der Layer des Polygons wird als „Bereich PedSim“ festgelegt. Der Bereich erscheint nun als gelbes Rechteck auf dem Bildschirm.

Zur Benutzereingabe wird noch ein Befehl geschickt, der das Bild zum Polygon des Bereichs zoomt und einer, der das Layerfenster öffnet. Dabei wird der Nutzer aufgefordert alle Layer, die in der Zeichnung vorhanden sind auf die Notwendigkeit für die Simulation zu prüfen. Sind die Layer dafür nicht notwendig, können sie einfach ausgeschaltet werden. Beispielsweise können so Schnittlinien ausgeblendet werden. Dies spart dem Nutzer viel Zeit, da er später bei der Objektauswahl einfach alle Objekte der Zeichnung auswählen und diese exportieren kann. Der Sortieraufwand zwischen für die Simulation wichtigen und unwichtigen Objekten ist so geringer.

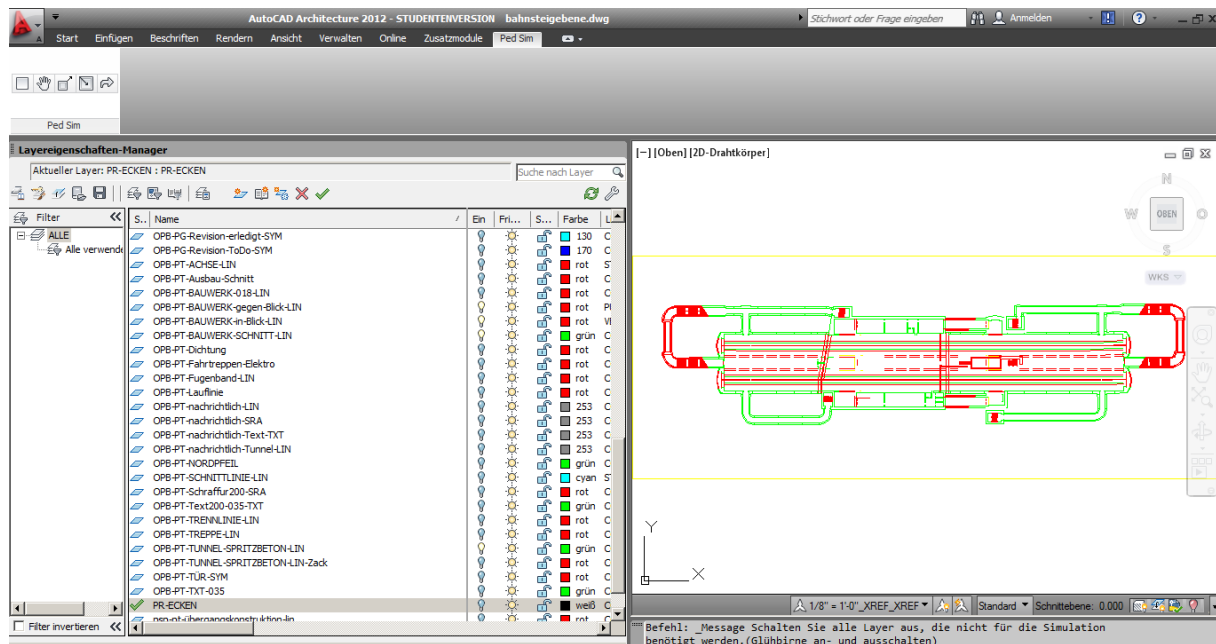


Abbildung 4.3: Screenshot nach dem Auswählen des Bereichs. Der Nutzer kann auf der linken Seite ganz einfach Layer ein- und ausblenden.

SelectWall

Bei der Funktion SelectWall zum Auswählen von Wänden wird ebenfalls ein neuer Layer erstellt. Allerdings mit dem Namen „Wand PedSim“ und der Farbe Grün. Genauer dazu wurde bereits in der Funktion Area erklärt. Anschließend kommt die Prozedur SelectObjectsOnscreen. Mit dieser kann man Objekte aus dem Zeichenfenster von AutoCAD auswählen und in den Layer „Wand PedSim“ kopieren. Dazu müssen wieder das aktuelle Dokument und das Datenmodell bestimmt werden, um eine Transaktion zu starten. Es wird ein Filter eingebaut, der nur bestimmte Objekte aufnimmt. In der Funktion Filterselectionset wird der aktuelle Dokumenteneditor bestimmt. Danach werden mit einer TypedValue-Array die Filterkriterien bestimmt. Es sollen nur Linien, Polylinien und Bögen ausgewählt werden können. Das Filterkriterium wird anschließend in ein SelectionFilter-Objekt eingefügt.

Nun wird nach einem im Zeichenfenster gewählten Objekt gefragt und überprüft, ob ein Objekt gewählt wurde. Es lassen sich also nur die gewünschten Objekte auswählen. Diese werden am Ende der Funktion zurückgegeben. Nun geht man Schritt für Schritt durch alle gewählten Objekte. Dazu öffnet man das gewählte Objekt zum Lesen, wobei man extra einstellen muss, dass auch gesperrte Objekte gelesen werden können. Ansonsten könnten gesperrte Elemente und Elemente nach gesperrten Elementen nicht mehr ausgelesen werden.

Anschließend kopiert man die Objekte mit der Prozedur SingleCopy. Dazu wird wieder eine Transaktion gestartet, der Blocktable zum Lesen geöffnet und der Block-Table-Record-Model-Space zum Schreiben geöffnet. Der Block-Table-Record-Model-Space ist ein Block, der mehrere Elemente beinhaltet (AutoCAD Help, 2013).

Jetzt kann man das Objekt kopieren, den Layer festlegen und es zum Blocktable hinzufügen. Man muss hier jedoch zwischen Linien, Polylinien und Bögen unterscheiden, da man nicht

einfach das gewählte Objekt an sich kopieren kann, sondern es spezifisch, beispielsweise als Linie, kopieren muss. Jetzt können beide Transaktionen wieder beendet werden.

Mit dieser Funktion kann man also Linien, Polylinien und Bögen auswählen, kopieren und die kopierten Objekte in den Layer „Wand PedSim“ speichern. Alle diese Objekte sind somit in der Zeichnung doppelt vorhanden, jedoch in verschiedenen Layern. Die aktuelle Zeichnung muss also nicht verändert, sondern nur erweitert werden.

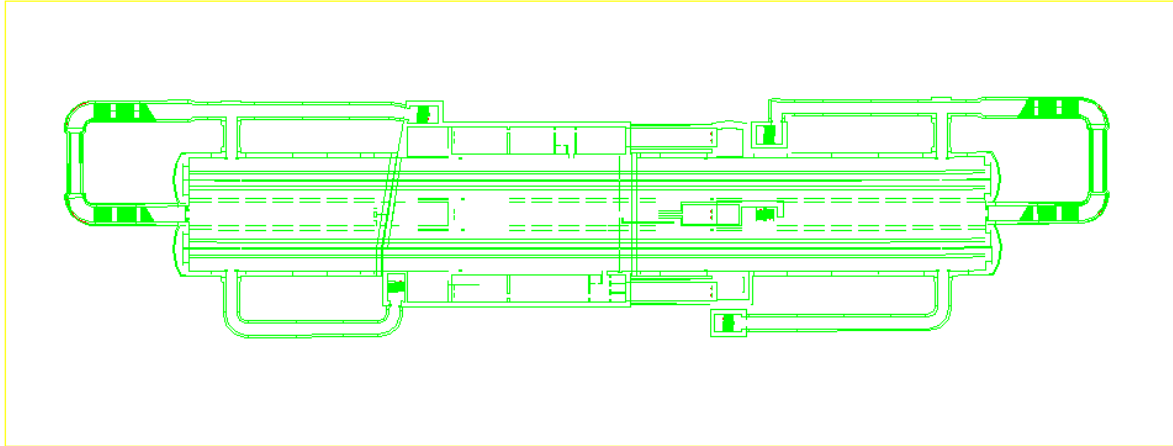


Abbildung 4.4: Nun wurden alle Layer in den Layer „Wand PedSim“ kopiert. Darunter liegen jedoch auch noch die alten Objekte mit ihren ursprünglichen Layern.

Zusätzlich kann man jetzt noch Quellen und Ziele einfügen. Diese Funktionen werden in den nächsten beiden Unterpunkten beschrieben.

Source

Bei der Funktion zum Einfügen von Quellen wird ein Layer mit dem Namen „Source PedSim“ und der Farbe Rot erstellt (siehe Funktion Area). Die Prozedur GetPointsFromUser ist ebenfalls wie bei der Funktion Area aufgebaut. Hier wird ein quadratisches Polygon um den gewählten Punkt erzeugt. In dieser Form werden die Quellen auch in dem Fußgängersimulator CrowdControl gespeichert.

Target

Diese Funktion dient zum Einfügen von Zielen für die Fußgängerströme. Sie ist fast identisch mit der Funktion Source. Es wird ein Layer mit dem Namen „Target PedSim“ in der Farbe Blau erstellt und ebenfalls ein quadratisches Polygon um den vom Benutzer gewählten Punkt erzeugt.

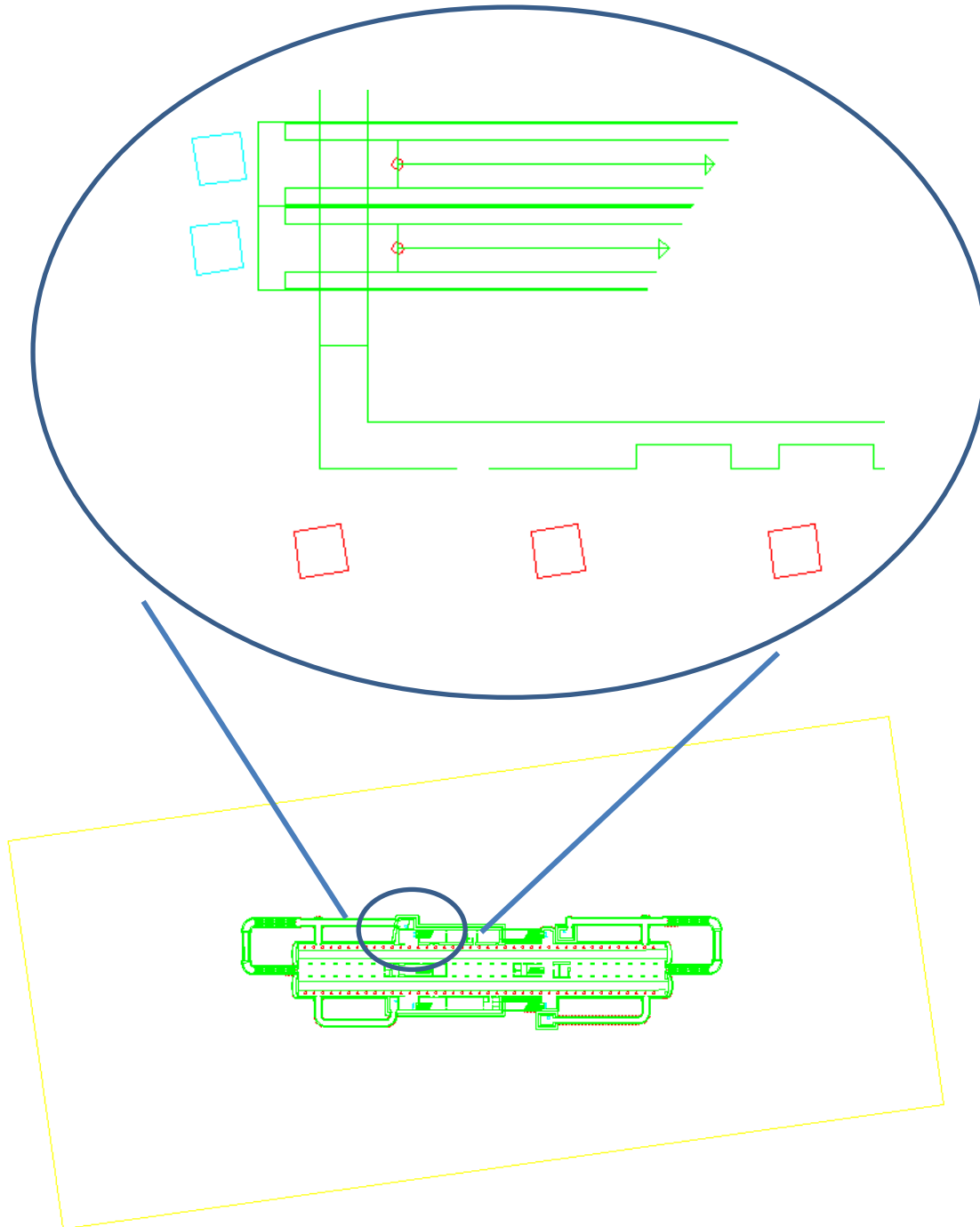


Abbildung 4.5: In diesem Beispiel wurden in AutoCAD bereits der Bereich (gelbes Rechteck), die Wände (grüne Linien), Quellen (rote Quadrate) und Ziele (blaue Quadrate) ausgewählt. Die Daten sind jetzt abgespeichert und müssen nur noch in eine XML-Datei exportiert werden.

Nun sind alle erforderlichen Daten, die der Nutzer mit dem Plug-In erstellen kann, vorhanden. Der Bereich ist festgelegt, die Wände wurden ausgewählt und Quellen und Ziele eingezeichnet. Jetzt müssen die Daten der erstellten Objekte nur noch richtig in ein XML-Dokument exportiert werden. Dazu dient die Funktion ConvertToXML.

ConvertToXML

Betätigt man die Funktion ConvertToXML, werden der Bereich, Wände, Quellen und Ziele in eine XML-Datei exportiert. Als Erstes benötigt man dazu einen Speicherpfad, wo die XML-Datei gespeichert werden soll. Dies geht über die Funktion SaveFileDialog. Mit dieser Funktion wird ein Fenster geöffnet, in dem man Speicherort und Dateinamen angeben kann. Beides zusammen wird dann von der Funktion zurückgegeben. Diese Funktion ist schon weitgehend vorgefertigt. Es müssen jedoch noch ein paar Parameter, wie Titel, Dateityp und Festplatte, angegeben werden.

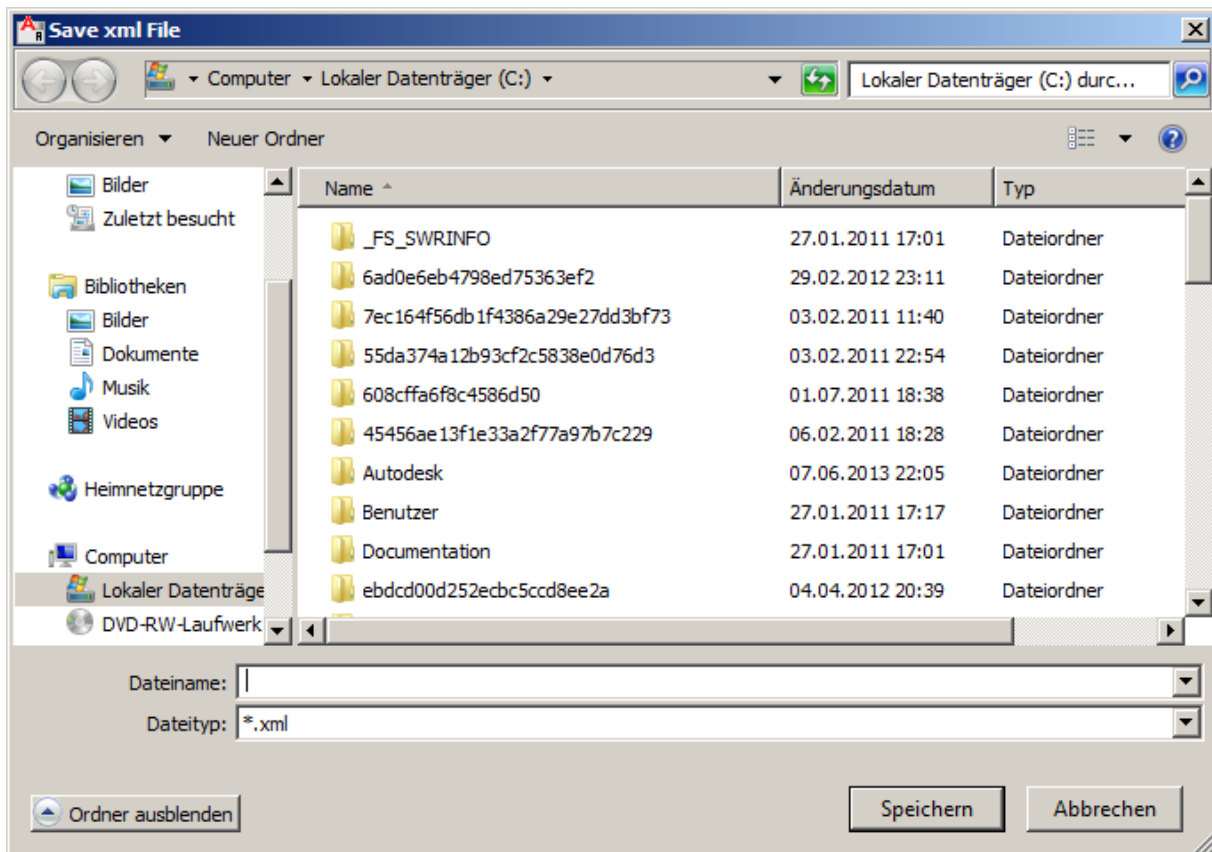


Abbildung 4.6: Die Funktion SaveFileDialog wird sehr häufig verwendet. Man kann hier den Speicherpfad und den Dateinamen eingeben.

Danach kommt die Prozedur Exportieren. Hier werden die Elemente und Attribute der XML-Datei belegt. Dazu benötigt man eine Klassenstruktur, wie sie in Abschnitt 4.2 (XML-Serializer) erklärt wird. Hier werden zum Beispiel die Simulationsversion oder die Art des Koordinatensystems mit Daten belegt. Dazu deklariert man eine Variable als die entsprechend erstellte Klasse und kann die Daten darin abspeichern. Diese Werte können nicht aus der CAD-Datei entnommen werden und werden mit Standardwerten belegt.

Anders sieht es mit den Werten des Bereichs, der Wände, der Quellen und der Ziele aus. Die Daten des Bereichs können durch die Längen des Rechtecks (Polygon) in X- und Y-Richtung bestimmt werden und dann in das Element Topology eingefügt werden.

Bei den Wänden, Quellen und Zielen ist es einfacher, die Daten aus dem Datenmodell der AutoCAD-Datei zu beziehen. Das hat den zusätzlichen Vorteil, dass die bereits erstellten Wände auch nach einem Neustart des Programms wieder exportiert werden können. Dazu

muss man wieder das aktuelle Dokument und das aktuelle Datenmodel bestimmen und eine Transaktion starten. Anschließend stellt man den Blocktable und den Block-Table-Record-Model-Space auf den Lesemodus. Jetzt kann man die Objekte einzeln durchgehen. Das aktuelle Objekt wird hier speziell nach seinem angehörigen Layer betrachtet.

Ist der Layer „Wand PedSim“, wird nach der Reihe ein fortlaufender Name eingefügt (wall1, wall2,..., walln). Je nachdem, ob es sich um eine Linie, einen Bogen oder eine Polylinie handelt, werden zu der Wand zwei oder mehrere Punkte hinzugefügt. Für Linien verwendet man dazu den Anfangs- und den Endpunkt. Bei Polylinien ohne Bögen können die einzelnen Knoten verwendet werden. Bei Polygonen muss man jedoch unterscheiden, ob die Polylinien geschlossen sind oder nicht. Diese Eigenschaft ist in AutoCAD als Attribut gespeichert. Ist dies der Fall, muss der erste Punkt des Polygons zum Schluss nochmal in die Wand eingefügt werden. Ansonsten ist das Polygon im Fußgängersimulator nicht geschlossen.

Bögen sind im Simulator nicht definiert und müssen daher als Kette einzelner Linien einen Bogen nachahmen. Sie kommen einzeln oder als Bestandteil von Polylinien vor und sind im Datenmodel von AutoCAD durch Anfangs-, End- und Mittelpunkt und die Drehrichtung festgelegt. Die Punkte zwischen Anfangs- und Endpunkt sind also nicht direkt abgespeichert. Daher ist es notwendig Zwischenpunkte zu berechnen. Der Bogen wird dafür näherungsweise pro Viertelbogen in acht Abschnitte unterteilt.

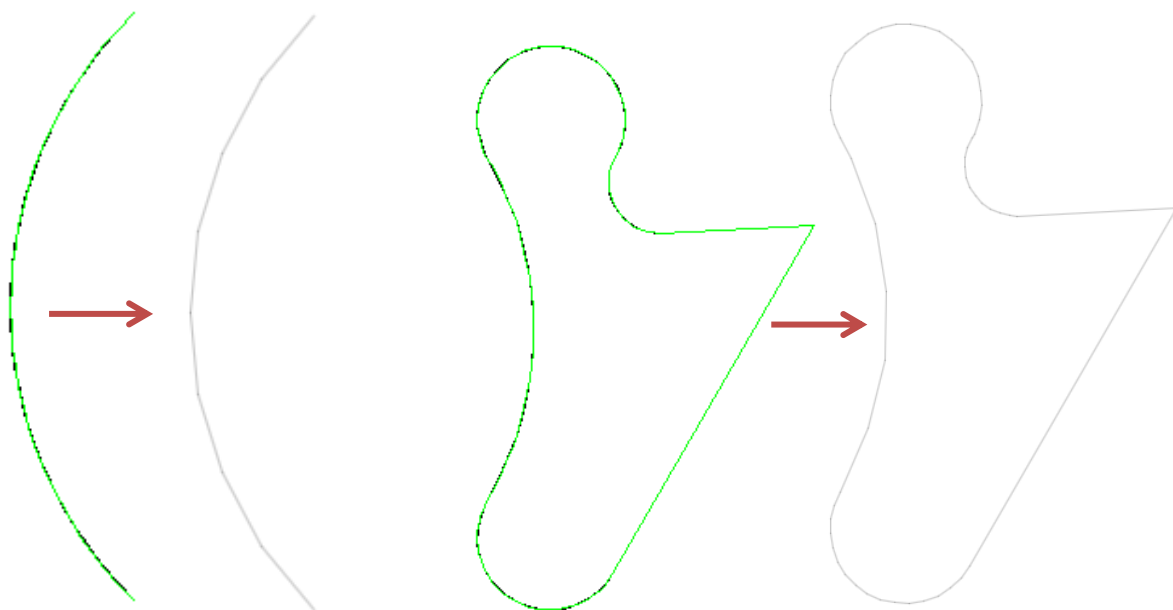


Abbildung 4.7: linke Seite: Ein Viertelbogen wird aus AutoCAD in den Simulator exportiert. Man erkennt hier sehr gut die Unterteilung in acht Abschnitte.
rechte Seite: Ein geschlossenes Polygon wird aus AutoCAD in den Simulator exportiert

Wichtig ist auch, dass zwei aufeinanderfolgende Punkte einer Wand einen Abstand von mehr als 0,002 Einheiten haben müssen. Dies ist wahrscheinlich auf die unverhältnismäßig kleine Größe im Gegensatz zu einem Grid des Simulators zurückzuführen. Ansonsten kann der Simulator ab dieser Stelle die XML-Datei nicht mehr lesen. Sind die Koordinaten der Punkte berechnet, muss noch die Koordinate der links unten liegenden Ecke des Polygons „Bereich PedSim“ abgezogen werden. Ansonsten liegen die Wände nicht in dem gewählten Bereich.

Gleiches gilt für die Berechnung der Quellen und Ziele. Diese werden nach dem gleichen Prinzip wie die Wände benannt. Allerdings müssen diese in der XML-Datei als Location eingefügt werden. Der Name dieser Locations kann dann entweder bei den Quellen oder den Zielen aufgeführt werden.

Ist das aktuelle Objekt richtig eingeteilt und die Dateien richtig in der Klassenhierarchie abgespeichert, kann das nächste Objekt betrachtet werden. Wichtig ist, dass alle Objekte in der Klassenhierarchie abgespeichert sind. Nur so stimmt die spätere Ausgabe bei der Serialisierung. Die Wände, Quellen und Ziele werden in Reihen gespeichert. Dabei ist darauf zu achten, dass diese immer richtig angepasst werden und die Daten an die richtige Stelle gespeichert werden. Ansonsten kann es passieren, dass einzelne Werte überschrieben werden. Sind alle Werte abgespeichert, kann die Transaktion beendet werden.

Nun, da alle benötigten Daten in der Klassenhierarchie vorhanden sind, nimmt man die Variable der Klasse, die alle anderen Klassen enthält und serialisiert diese. Diese Prozedur muss man davor bereits in die „oberste“ Klasse einfügen. Zum Speichern werden der vorher abgefragte Speicherpfad und Dateiname verwendet. Ist die Datei abgespeichert, kann sie mit CrowdControl geöffnet und ein Evakuierungsszenario entwickelt werden.

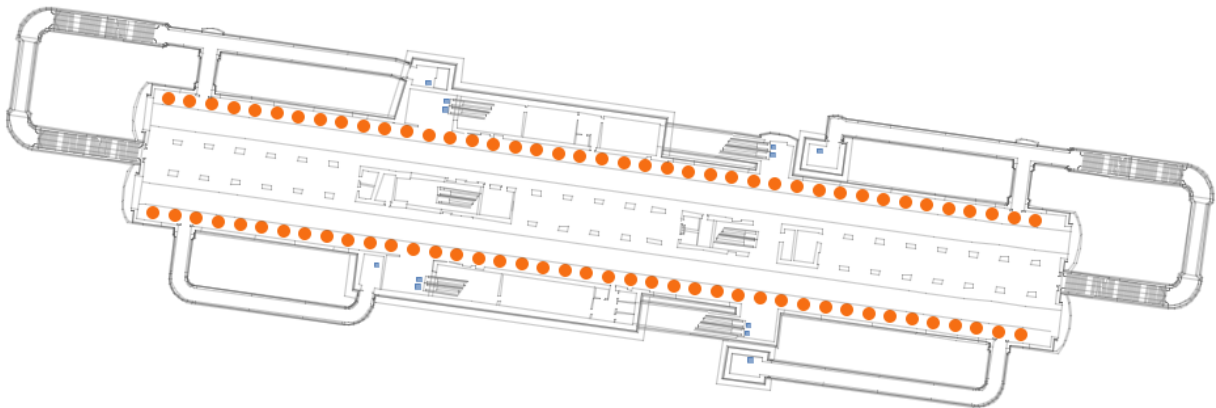


Abbildung 4.8: Nun wurden die Daten in eine XML-Datei exportiert und von dem Simulator CrowdControl geöffnet.

Im nächsten Punkt werden die Rahmenbedingungen für eine solche Simulation entwickelt und durchgeführt.

5 Ergebnisse

Durch die Verwendung des Plug-Ins kann eine Fußgängersimulation anhand der Pläne einer Genehmigungsplanung relativ einfach durchgeführt werden. Wo die Möglichkeiten und Grenzen einer Simulation auf der Basis von CAD-Plänen liegen, soll in den nächsten Teilen am Beispiel der Bahnsteigebene der Station Marienhof deutlich gemacht werden.

5.1 Durchführung einer Simulation anhand der übertragenen Daten

Um eine Simulation durchzuführen, müssen einige Annahmen getroffen werden. Es wird davon ausgegangen, dass die Personen auf dem mittleren Bahnsteig zwischen den beiden S-Bahnlinien auf den Zug warten und die ankommenden Fahrgäste in Richtung der beiden äußeren Seiten aussteigen. Daher ist der Fall mit den meisten zu evakuierenden Personen, dass zwei volle Züge ankommen und der mittlere Bereich voll mit wartenden Personen ist. Bei zwei nebeneinander liegenden Treppen wird davon ausgegangen, dass eine Treppe eine Rolltreppe und die andere eine normale Treppe ist.

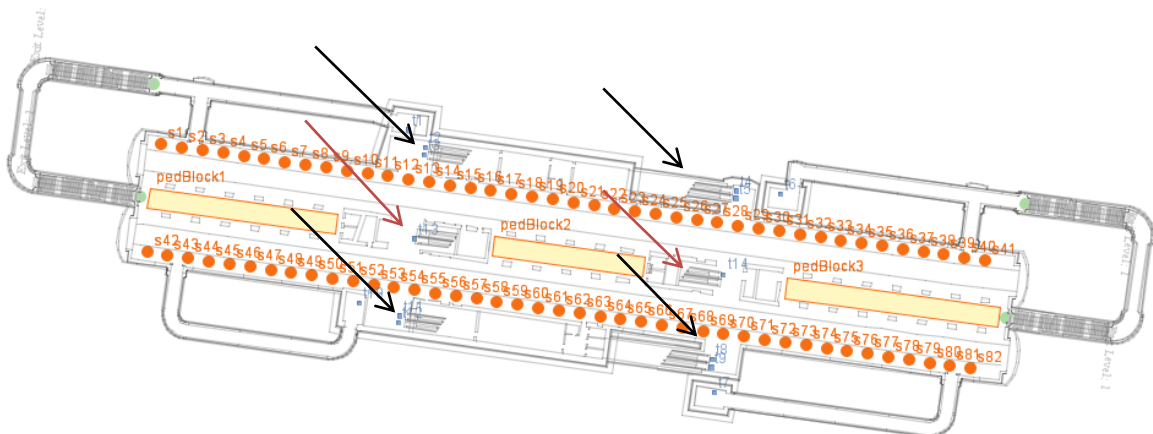


Abbildung 5.1: An jedem Pfeil befinden sich zwei Treppen. Davon ist jeweils eine der beiden eine Rolltreppe und die andere eine normale Treppe. Die roten Pfeile (jeweils in der Mitte) sollen verdeutlichen, dass hier die Rolltreppen nach unten fahren und somit nicht für die Evakuierung zur Verfügung stehen.

Die Rolltreppen fahren entsprechend beim mittleren Bahnsteig nach unten und bei den äußeren nach oben. Für die Evakuierung werden nur Fahrtreppen, die nach oben fahren, miteinbezogen.

In München fahren meist S-Bahnen von der DB-Baureihe 423 (Interessengemeinschaft S-Bahn München e.V., 2008).



Abbildung 5.2: Münchener S-Bahn der DB-Baureihe 423 (Länge und Anzahl der Türen nicht originalgetreu!) (Wikipedia die freie Enzyklopädie, 2004)

Diese haben bei einer Länge von 67 Metern 192 Sitz- und 352 Stehplätze. Bei einer Länge von 200 Metern ergeben sich so 1600 Insassen. Diese können auf 36 Türen (eine Seite) mit jeweils 44 Fahrgästen verteilt werden. Für das Verlassen des Zuges wird eine Dauer von 15 Sekunden angenommen.

Die Passanten im Wartebereich werden mit der Fläche hinter den Sicherheitsstreifen veranschlagt. Diese ergibt sich aus der Fläche des blauen Rechteckes minus den Flächen der roten Rechtecke aus Abbildung 5.3.

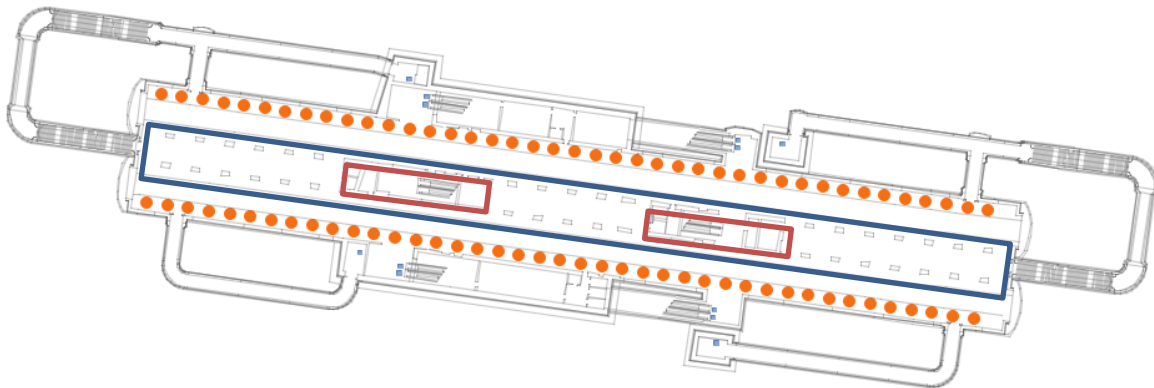


Abbildung 5.3: Die Fläche der wartenden Passanten ergibt sich aus der Fläche des blauen Rechteckes minus der Fläche der roten Rechtecke.

So kommt man auf eine Fläche von 2.154 Quadratmeter, wodurch sich mit einer Dichte von zwei Fußgängern pro Quadratmeter eine Anzahl von 4308 Personen ergibt. Rolltreppen bewegen sich im Schnitt mit 0,6 Meter pro Sekunde (How Stuff Works, 2013). Bei maximal zwei nebeneinander stehenden Personen kommt man so auf eine Beförderung von drei Personen pro Sekunde. Der gleiche Wert wird für die normalen Treppen verwendet, da sie in etwa über die gleichen Kapazitäten verfügen. Fahrstühle sind nicht berücksichtigt.

Für die Durchführung öffnet man in AutoCAD den Grundriss der Bahnsteigebene und lädt das erstellte Plug-In zum Exportieren von XML-Dateien. Man wählt den Bereich und die wichtigen Wände. Punktquellen (S-Bahn-Türen) und Punktziele (Treppen) können bereits in AutoCAD eingefügt werden. Die XML-Datei wird nun erstellt und mit CrowdControl geöffnet. Zuerst müssen alle Quellen mit Zielen verbunden werden. Dazu werden nur sich in der Nähe befindende Ziele mit den Quellen verbunden, da davon ausgegangen wird, dass sich die Personen an den Fluchtplan halten. Ansonsten behindern sich die Fußgänger gegenseitig und die Evakuierung verzögert sich um ein Vielfaches. Um solche Behinderungen zu vermeiden, wird der mittlere Bereich mit den wartenden Personen in drei Teile unterteilt, die jeweils mit anderen Zielen verbunden werden.

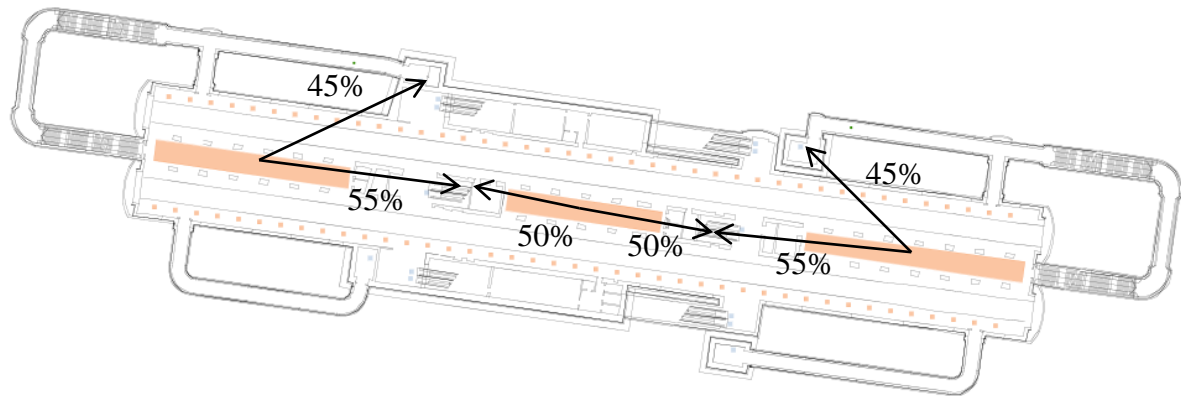


Abbildung 5.4: Hier sieht man die drei Flächenquellen der wartenden Personen mit Pfeilen zu den Zielen und der jeweiligen Aufteilung der Personen.

Danach wird ein neues Level (Stockwerk) mit einer Höhe von zehn Metern eingefügt und mit vier Treppen verbunden (Fluchtweg für einen Teil der wartenden Fahrgäste). Die drei Flächenquellen werden dann mit den Zielen verbunden. Wenn kein anderer Weg vorhanden ist, gehen die Fußgänger automatisch über die Treppe.

Im Anschluss werden die Konfigurationen der Quellen und Ziele angepasst. Während von den Bahntüren fünfzehn Sekunden lang drei Personen pro Sekunde abgegeben werden, stehen die wartenden Personen von Anfang an bereit. Die Ziele können 3 Personen pro Sekunde aufnehmen.

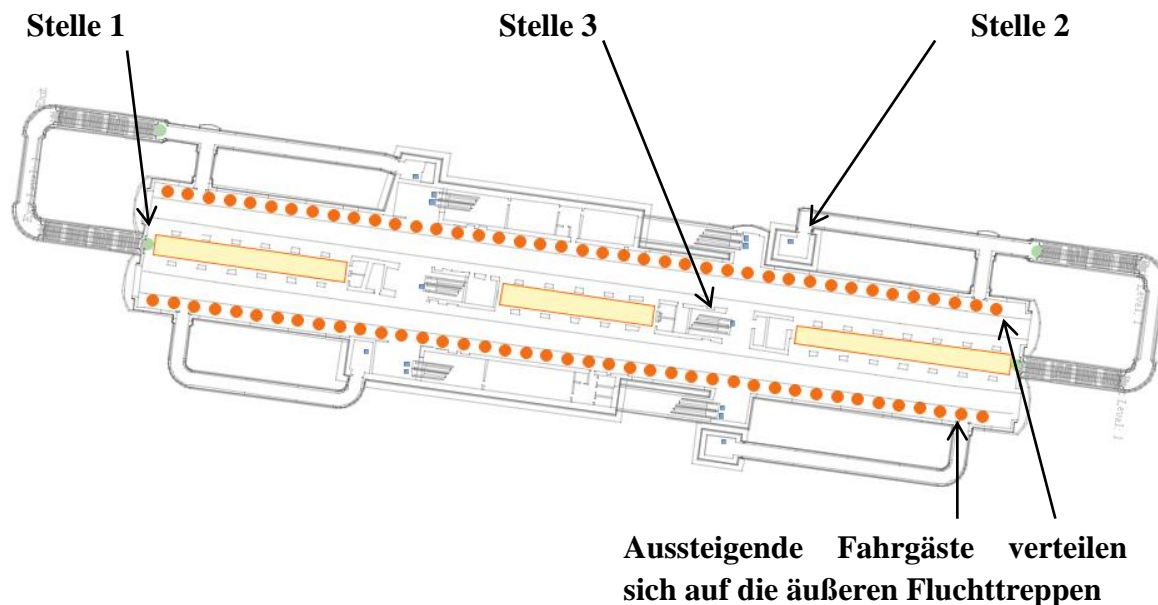


Abbildung 5.5: Entwickeltes Evakuierungsszenario für die Bahnsteigebene der S-Bahn-Station Marienhof. Die mittleren orangenen Rechtecke stellen die wartenden Personen und die orangenen Punkte die Bahntüren dar. Die blauen Quadrate sind Ziele an den Treppen. Außen werden die Fluchtweg der wartenden Gäste durch Treppen mit dem nächst höherem Stockwerk verbunden.

Führt man die Simulation durch, sieht man sehr anschaulich, wie eine Evakuierung ablaufen könnte. In den folgenden Abbildungen ist der Stand der Simulation in mehreren Zeitabschnitten dargestellt.

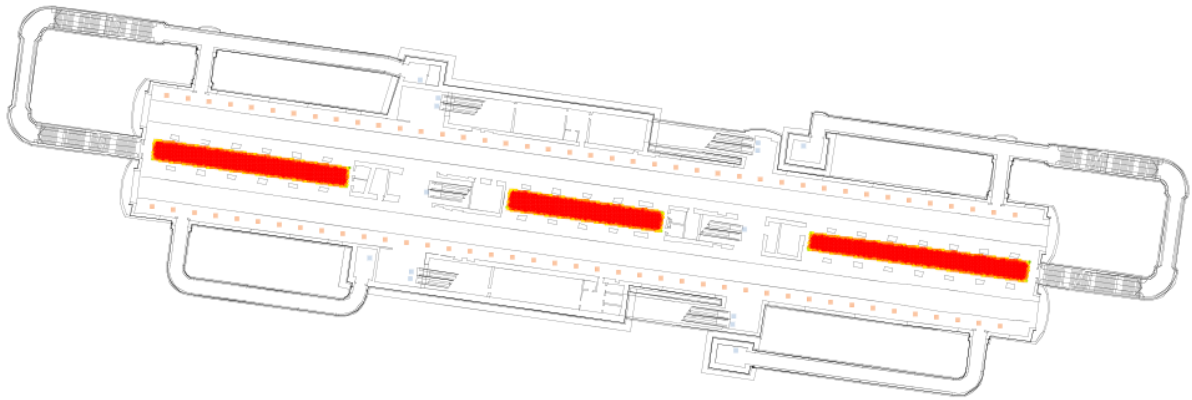


Abbildung 5.6: Stand vor Beginn der Simulation

Auf den drei Flächenquellen stehen die Personen von Anfang an bereit.

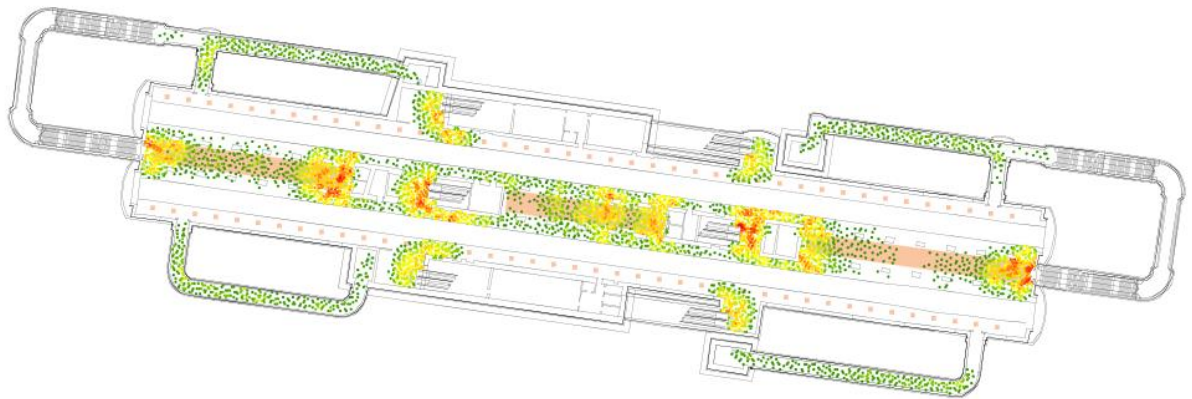


Abbildung 5.7: Stand nach 60 Sekunden

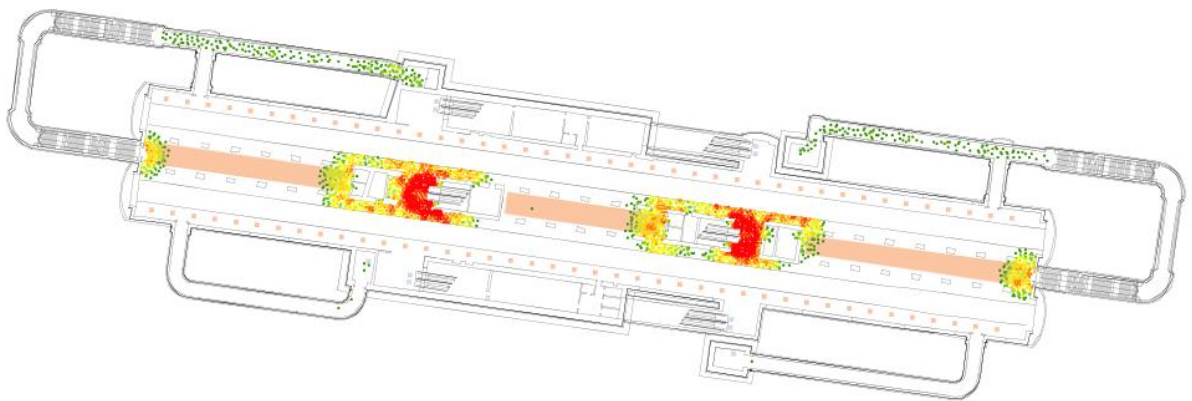


Abbildung 5.8: Stand nach 180 Sekunden (3 Minuten)

An dem Simulationsstand nach drei Minuten sieht man, dass nun die letzten Fahrgäste der äußeren Bahnsteige die Bahnsteigebene verlassen.

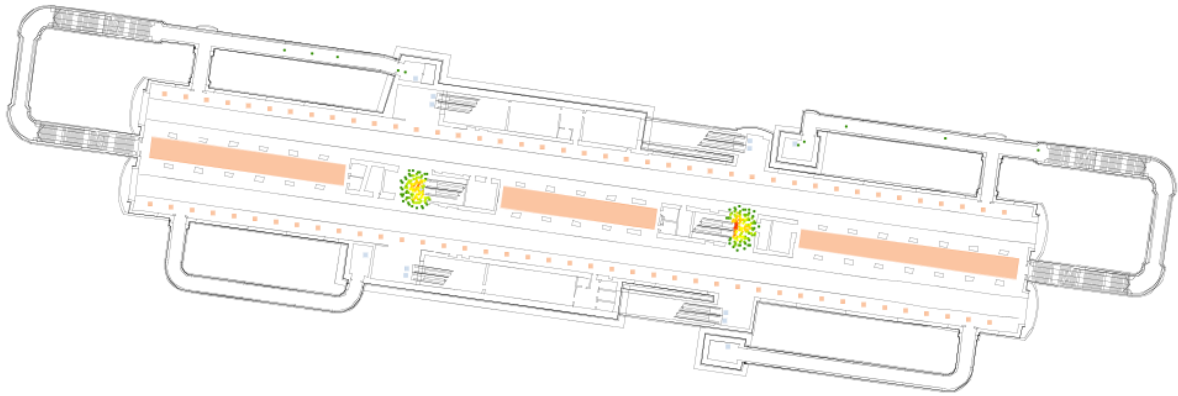


Abbildung 5.9: Stand nach 480 Sekunden (8 Minuten)

Nach acht Minuten haben die meisten Personen die Bahnsteigebene verlassen. Bis es alle Personen geschafft haben dauert es jedoch über elf Minuten.

Für Personen, die aus der S-Bahn kommen, kommt es bei der Evakuierung zu keinen größeren Engstellen.

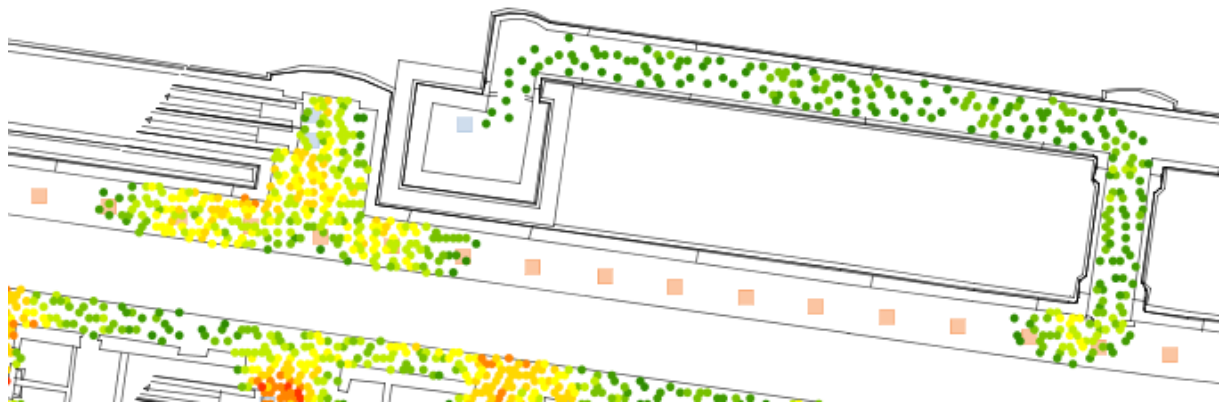


Abbildung 5.10: Die Personen aus der S-Bahn können die Bahnsteigebene ohne größere Probleme verlassen. (Stelle 2 aus Abbildung 5.5)

Die Personen, die auf die S-Bahn gewartet haben, müssen sich jedoch auf die äußeren und inneren Fluchtwege verteilen. Hier kommt es besonders vor den Treppen zu Engstellen.

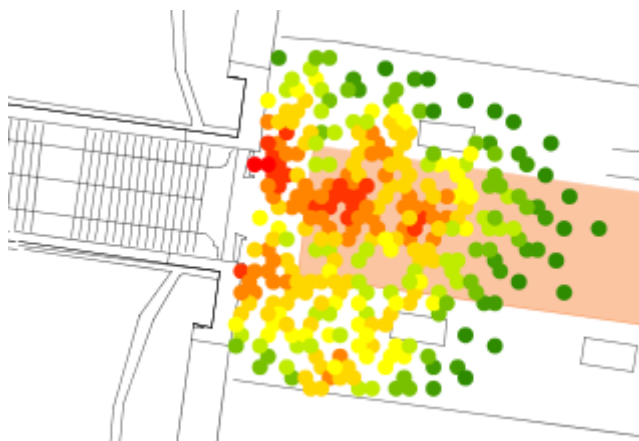


Abbildung 5.11: Vor der Treppe kommt es zu erhöhten Personendichten. (Stelle 1 aus Abbildung 5.5)

Außerdem sind besonders die Stellen gefährlich, an denen der Bahnsteig in der Mitte bebaut ist. Hier sieht man in der Simulation an der roten Farbe der Personen, dass sie Gefahr laufen, auf die Schienen gedrückt zu werden. Das könnte in Kombination mit einer einfahrenden S-Bahn sehr gefährlich werden. Um eine sichere Evakuierung zu garantieren sollte diese Stelle besonders genau geprüft werden.

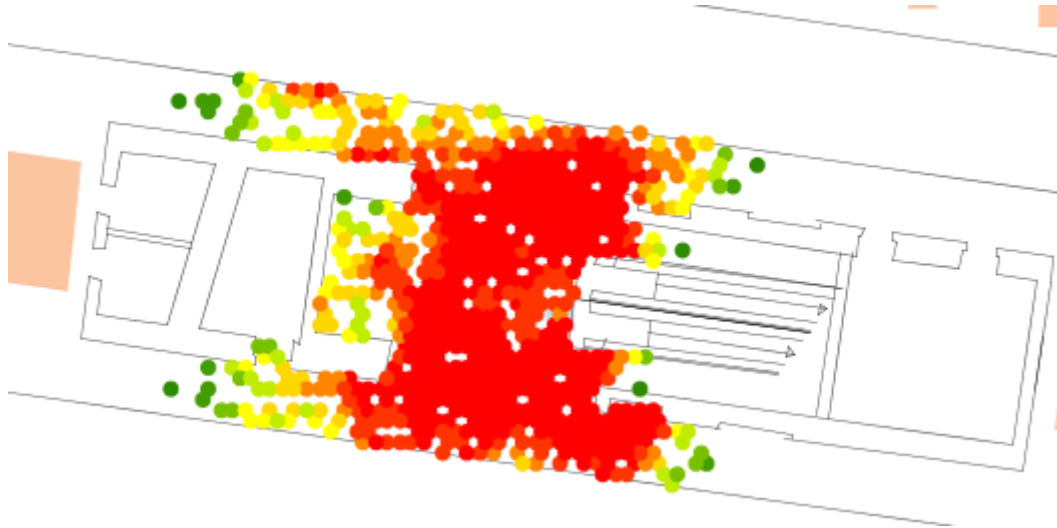


Abbildung 5.12: An den Stellen, wo der mittlere Bahnsteig bebaut ist, kommt es zu hohen Menschendichten. Die Personen laufen Gefahr, auf die Gleise gedrückt zu werden. Ebenso, wie in Abbildung 5.4, stauen sich die Personen auch hier vor der Treppe. (Stelle 3 aus Abbildung 5.5)

Man sieht an diesem Beispiel sehr schön, dass man mit solch einer Simulation sehr anschaulich die Engstellen von Fluchtplänen erkennen und diese schon während der Planungsphase anpassen kann. Es ist jedoch zu beachten, dass sich die Personen im Fall eines Brandes oder eines Terroranschlags auch anders verhalten könnten, als es der Fluchtplan vorsieht. Solche Einflüsse haben eine sehr große Auswirkung auf die Evakuierungszeit und sind in der Planung nur schwierig zu prüfen. Die Probleme und Möglichkeiten einer solchen Evakuierungssimulation werden in den folgenden Punkten dargestellt.

5.2 Probleme und Möglichkeiten

Evakuierungssimulationen bieten viele Möglichkeiten zur Gestaltung effektiverer und sicherer Fluchtwege. Allerdings gibt es auch noch einige Probleme auf Seiten der Simulatoren und der Einbindung in den Planungsprozess von Bauvorhaben. Im Folgenden werden die bei der Simulation entstandenen Probleme genauer aufgezeigt, aber auch Möglichkeiten für die Behebung genannt.

Simulation

Ein Problem ist, dass für eine Simulation sehr viele Annahmen notwendig sind. Während die mögliche Anzahl von Personen und die Kapazitäten der einzelnen Wege (definiert durch Quellen und Ziele) sehr genau bestimmt werden können, ist über die Wahl des Zieles der Personen nur wenig bekannt. Dies muss jedoch vom Planer genau angegeben werden. Zwar

werden die Fluchtwege im Bahnhof genau ausgedeutet, bei der Evakuierung könnten aber auch einzelne Wege versperrt sein. All diese Möglichkeiten zu prüfen, ist sehr aufwendig. Das gilt besonders, weil sich gegenseitig hindernde Personen sehr stark auf die Evakuierungszeit und die benötigte Rechenleistung auswirken. Das führt zu dem nächsten Problem der Evakuierungssimulation.

Obwohl im Beispiel nur die Bahnsteigebene betrachtet wird, wurde das Limit für den verwendeten Rechner zur Echtzeitberechnung schon erreicht. Verwendet wurde ein Athlon II Dual-Core Prozessor mit zwei Kernen mit jeweils 2,1 GHz. Die Simulation von 1000 Sekunden dauerte damit beim ersten Versuch mehrere Stunden. Legt man jedoch die Personen auf die Fluchtwege um, geht die Berechnung um ein Vielfaches schneller. Dies zeigt sowohl die Notwendigkeit von festgelegten Fluchtwegen als auch die hohe benötigte Rechenleistung von sich gegenseitig hindernden Fußgängern. Als Folge von wenig strukturierten Fluchtwegen ergeben sich somit höhere Evakuierungs- und Berechnungszeiten. Erst nach der Entwicklung geeigneter Fluchtwege konnten sowohl die Rechenzeit als auch die Evakuierungszeit auf einen vernünftigen Wert reduziert werden.

Am meisten half dabei, die wartenden Personen des mittleren Bahnsteigs auf vier verschiedene Bereiche mit verschiedenen Zielen aufzuteilen. Die Rechenzeit konnte so von 7,4 Stunden auf 19 Minuten reduziert werden. Durch diese Veränderung kamen auch viel kürzere Evakuierungszeiten heraus. Während am Anfang nach 15 Minuten noch kein Verlassen der Bahnsteigebene der Fußgänger des mittleren Bahnsteigs abzusehen war, hatten die Personen nach der Änderung schon nach sieben Minuten die Ebene verlassen. Der Nutzer muss also schon vor der Simulation einen groben Fluchtplan haben. Dieser kann dann nach und nach optimiert werden.

Der Simulator ist zwar für 50.000 Personen ausgelegt, dafür ist jedoch bei vielen Behinderungen sehr viel Rechenzeit nötig. Empfehlenswert wäre hier ein schnellerer Prozessor, als im Versuch verwendet wurde. Anzumerken ist hier, dass nur eine Ebene betrachtet wurde. Da eine realistische und vollständige Evakuierungssimulation jedoch nur durch die Betrachtung aller Ebenen möglich ist, würde bei einer vollständigen Simulation noch mehr Rechenleistung benötigt werden.

Außerdem ist ein geübter Planer von Nöten. Am sinnvollsten wäre es also, die Evakuierungssimulation an die Planer zu übergeben, die sich auch jetzt schon um die Fluchtpläne für die Genehmigungsplanung kümmern. Zu empfehlen wäre, dass sich diese dann mit entsprechenden Hochleistungsrechnern ausstatten. Dies ist für ein wirtschaftliches Arbeiten mit einem Simulator von Vorteil. Denkbar wäre auch eine Einrichtung von zentralen Hochleistungsrechnern, die bei Bedarf von den Planungsbüros genutzt werden könnten.

Da Fußgängersimulatoren schon sehr weit entwickelt sind und bei entsprechender Rechenleistung sehr gute Ergebnisse liefern, wäre es wünschenswert, diese Systeme besser in AutoCAD einzubinden. Probleme und Möglichkeiten bei der Anwendung des AutoCAD-Plug-Ins werden im Folgenden aufgezeigt.

Plug-In

Ein Problem ist, dass die einzelnen Objekte in den CAD-Plänen oft nicht eindeutig in Layer eingeteilt sind. Es bestehen beispielsweise mehrere Layer für gleichartige Objekte, wie Wände, die unzureichend benannt werden. Es ist somit sehr aufwendig zu erkennen, welche Layer für die Simulation notwendig sind und welche nicht. Gäbe es weniger und eindeutig benannte Layer, die möglicherweise noch in Ordner gegliedert wären, wäre es viel einfacher wichtige Objekte zu finden und die anderen auszublenden. Dies müsste schon von den Planern der Entwurfs- und Genehmigungsplanung berücksichtigt werden. Großes Potential liegt hier im Bereich Building Information Modeling. Hier stehen schon vorgefertigte Objektklassen, wie Wände oder Treppen, zur Verfügung und werden vom Programm differenziert betrachtet.

Ein weiteres Problem bei der Anwendung des Plug-Ins ist, dass es in AutoCAD sehr viele mögliche Einstellungen durch den Nutzer gibt. So ist bei dem Grundriss der Bahnsteigebene ein Layer gesperrt und die Ansicht verdreht. Folglich musste das Programm erst aktualisiert werden, um richtig arbeiten zu können. Die Ansicht wird beim Betätigen der Funktion „Bereich“ parallel zur X-Achse des Koordinatensystems ausgerichtet. Ansonsten stimmen die Werte in der XML-Datei nicht. Solche Programmfehler könnten auch bei anderen, noch nicht bekannten, Einstellungen auftreten.

Ein weiteres Problem des Plug-Ins ist auch, dass Quellen und Ziele zwar bereits in AutoCAD eingefügt werden können, deren Eigenschaften jedoch nicht bearbeitet werden können und sie nicht miteinander verbunden werden können. Dazu müsste man den Nutzer in der Benutzereingabe nach den Eigenschaften fragen und eine Funktion entwickeln, mit der man Quellen mit Zielen verbinden kann. Für eine solche Funktion könnte man wieder einen neuen Layer erstellen und mit diesem Linien zeichnen. Ist der Linienanfang in der Nähe einer Quelle und das Ende in der Nähe eines Ziels, könnte man diese in der XML-Datei als verbunden ausgeben. Ziel könnte es also sein, alle Funktionen des Simulators in das Plug-In zu integrieren. Das könnte soweit führen, dass der Simulator nur noch als Präprozessor im Hintergrund läuft und die Simulation in AutoCAD ausgeführt und angezeigt werden kann. Somit könnten alle Funktionen von AutoCAD und die des Simulators kombiniert werden. Viele Zeichenfunktionen sind in dem Simulator nicht vorhanden und könnten das Arbeiten stark vereinfachen. Außerdem wäre so nur noch eine graphische Benutzeroberfläche notwendig.

Zusätzlich müsste man dem Plug-In auch noch die Funktion zum Einfügen von mehreren Levels einbauen. Das geht im Moment nur durch Kopieren und Einfügen im XML-Text. Dazu müsste man die Grundrisse nacheinander mit der Angabe des Stockwerks abspeichern und anschließend serialisieren. Der Nutzer müsste mit verschiedenen CAD-Plänen arbeiten und die Grundrisse deckungsgleich übereinanderlegen. Dies kann kompliziert werden, da die einzelnen Pläne keinen räumlichen Bezug zueinander haben. Man muss diese erst noch in Höhe, Drehung und Position miteinander in Verbindung setzen. Großes Potential bieten zu diesem Thema BIM-Systeme wie REVIT von Autodesk. Sie arbeiten selbst mit verschiedenen Ebenen, die miteinander in Bezug stehen. Die Integration von Levels könnte hier für den Nutzer einfacher gestaltet werden. Außerdem bestehen hier auch genauere Einteilungen in Wände, Treppen, Türen, etc. Diese Objekte könnten automatisch für die Simulation

verwendet werden und es müssten nur noch Funktionen für Quellen, Ziele, Verbindungen und die jeweiligen Einstellungen eingefügt werden. Die Durchführung einer solchen Simulation wäre dadurch stark vereinfacht.

Da der Austausch von Modellen auf Basis von BIM-Systemen nur in einigen Regionen etabliert ist (Zweifel, 2010), ist die Verwendung des AutoCAD-Plug-Ins trotzdem sinnvoll. Die Übertragung der Pläne geht so viel einfacher und schneller als bei dem Import einer DXF-Datei. Die Pläne können einfach mit den Funktionen von AutoCAD verändert und anschließend exportiert werden. Schwachstellen innerhalb eines Stockwerks können so sehr einfach erkannt und verändert werden. Es ist also sinnvoll, die Fußgängersimulation nicht nur für ein noch wenig verwendetes BIM-System bereitzustellen, sondern sie auch in den aktuell praktizierten Planungsprozess einzubinden. Die Unternehmen scheuen sich bei BIM vor der Einarbeitung in neue und für sie komplizierter erscheinende Systeme, obwohl die Technik sehr viel Potenzial hat. Solange es sich noch nicht durchgesetzt hat, kann man aber Fußgängersimulation auch jetzt schon durchführen. Unterstützt wird dieses Argument von der Tatsache, dass Fußgängersimulatoren auch in 2D und nicht, wie BIM-Systeme, in 3D arbeiten.

6 Ausblick in die Zukunft

Zusammenfassend lässt sich sagen, dass Fußgängersimulation ein sehr interessantes Werkzeug zur Planung von Fluchtwegen darstellt. Die Genauigkeit dieser Modelle wird sich erst im Laufe der Jahre herausstellen und verbessern lassen. Hauptproblem ist die Verschiedenheit der Evakuierungsursachen. Bei einem Brand ist zum Beispiel von zentraler Bedeutung, wo dieser ausbricht. Dies hat Auswirkungen auf das Ziel der Fußgänger. Man kann jedoch schon jetzt sehr anschaulich Problemstellen erkennen und diese durch entsprechende Maßnahmen beseitigen. Dafür kann man die Kapazität eines Ziels reduzieren oder auf null stellen und die Auswirkungen überprüfen.

6.1 Fußgängersimulation für die Planung von Fluchtwegen

Sinnvoll wäre es, besonders bei Gebäuden oder Plätzen, an denen große Menschenmengen auftreten, eine Evakuierungssimulation verpflichtend einzuführen. Dazu gehören zum Beispiel Bahnhöfe oder Großveranstaltungen. Hier könnten die größten Problemstellen schon während der Planung besser erkannt werden. Am einfachsten wäre es, diese Aufgabe den Ingenieurbüros zu übertragen, die sich schon jetzt um den Fluchtplan bei der Genehmigungsplanung kümmern. Diese haben schon einen Großteil des nötigen Knowhows zu Fluchtplänen.

Ein weiterer Vorteil hierbei wäre, dass diese Planer schon jetzt fest in den Bauplanungsprozess integriert sind. Der Aufwand der Durchführung einer Simulation in Verbindung mit einem leistungsfähigen und benutzerfreundlichen AutoCAD-Plug-Ins ist dabei nicht besonders groß.

Für die Miteinbeziehung der verschiedenen, noch voneinander unabhängigen Ebenen, könnte man sich relativ einfach Abhilfe schaffen. Beispielsweise könnte man den Simulator durch eine Funktion erweitern, mit der man Punkte verschiedener Ebenen mit einem bestimmten Widerstand verbindet, die nicht genau übereinander liegen müssen. Dieser Widerstand könnte von der Entfernung zwischen den Punkten in der Ebene und der Höhe abhängig gemacht werden.

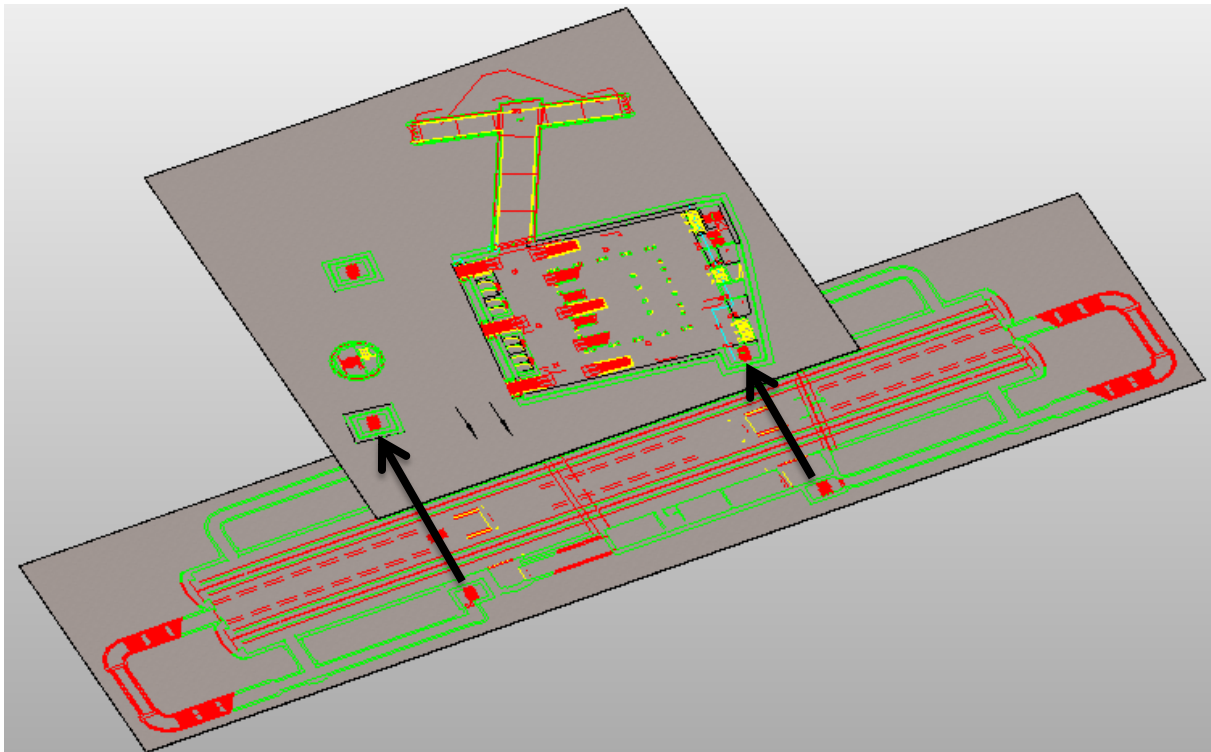


Abbildung 6.1: Abgebildet sind die Bahnsteigebene und die Verteilerebene der Station Marienhof. Einzelne Punkte im Treppenhaus könnten hier unter Angabe eines Widerstands miteinander verbunden werden.

Dazu ist es notwendig, mehrere Ebenen in die XML-Datei exportieren zu können. Das wäre mit einer Benutzereingabe des Index der Ebene und der Höhe relativ einfach durchzuführen. Evakuierungssimulationen können also auch mit bestehender CAD-Technologie sinnvoll durchgeführt werden. Der Aufwand hierfür ist jedoch etwas größer als bei neueren Building Information Systemen.

6.2 Building Information Modeling (BIM)

Mit dieser innovativen Planungstechnologie wird das in der Planung befindliche Bauwerk in 3D modelliert und soll während des gesamten Lebenszyklus verwendet werden (Rank, Teil II: Ingenieurwissenschaften, 2010). Hier sind schon vorgefertigte Objekte extra für die Bauindustrie vorhanden. Dazu gehören Wände, Ebenen, Treppen, Türen, etc.

Aus diesem Modell kann für jeden beliebigen Schnitt ein Bauplan erstellt werden. So könnte auch automatisch eine Datei mit allen Strukturdaten für eine Fußgängersimulation erstellt werden. Der Planer müsste dann nur noch Quellen, Ziele, Verbindungen und die dazu gehörigen Eigenschaften angeben. Eine Fußgängersimulation wäre mit noch weniger Arbeit verbunden. Außerdem könnte man auch hier die Simulation direkt im BIM-Programm anzeigen und die Fußgänger, wie in Computerspielen, als 3D-Objekte durch das Gebäude laufen lassen. Das wäre ein zusätzlicher anschaulicher und sehr massentauglicher Nebeneffekt. In der Theorie wäre das die beste Lösung. Das Problem von BIM ist nur, dass es sich in deutschen Planungsbüros nur sehr langsam durchsetzt. Ein Grund hierfür ist, dass die einzelnen Büros sich auf ein System einigen und ihre Daten ordentlich strukturiert in das Modell integrieren müssten. Das ist bei den vielen Beteiligten bei einem Bauprojekt keine

einfache Aufgabe. Vor allem, weil aufgrund der Ausschreibungen häufig einander unbekannte Ingenieurbüros zusammenarbeiten. Die Synergieeffekte eines solchen Systems konnten sich bisher noch nicht richtig durchsetzen. Die Zukunft wird zeigen, ob sich CAD-Systeme halten oder sich BIM-Systeme etablieren werden. Der Trend geht in Richtung BIM.

An einem Plug-In, wie dem aus dieser Arbeit kommt man jedoch nicht vorbei. Besonders benutzerfreundlich wäre es, die Fußgängersimulation komplett in das CAD- oder BIM-System zu integrieren. Die Benutzeroberfläche des Simulators fällt dann weg. Der große Vorteil eines Plug-Ins ist hier, dass die Randbedingungen für die Simulation soweit wie möglich automatisiert werden und nur möglichst wenig Input des Nutzers nötig ist. Das minimiert den Aufwand der Durchführung einer Simulation. Zusätzlich soll das Programm möglichst intuitiv bedienbar sein und der Nutzer durch Hilfestellungen unterstützt werden. Dies stellt die Bereitschaft von Büros für die Anwendung solcher Programme sicher. Der Einarbeitungsaufwand soll möglichst gering sein. Ähnliches gilt auch für BIM-Systeme. Schafft man es, die Programme so einfach zu gestalten, dass sie jeder bedienen kann, dann steht auch der Zukunft der Evakuierungssimulation im Bauplanungsprozess nichts mehr im Wege. Katastrophen wie bei der Loveparade 2010 in Duisburg könnten so vermieden werden. Auf Grund fehlgeleiteter Besucherströme und Fehlplanungen kamen damals im Eingangsbereich 21 Besucher ums Leben (Dörries, 2013).

Anhang A

A.1 Beispiel einer XML-Datei

```

<simulation version="v2.2">
  <scenario coordsystem="this" dx="0.0" dy="0.0" rotation="0.0">
    <topology maxx="100.0" maxy="100.0" name="default">
      <level height="0.0" index="0" name="Level_0">
        <location name="t1_Location">
          <point x="69.81539888682747" y="83.1734693877551"/>
          <point x="70.81539888682747" y="83.1734693877551"/>
          <point x="70.81539888682747" y="84.1734693877551"/>
          <point x="69.81539888682747" y="84.1734693877551"/>
        </location>
        <location name="s1_Location">
          <point x="34.19387755102041" y="41.429499072356215"/>
          <point x="35.19387755102041" y="41.429499072356215"/>
          <point x="35.19387755102041" y="42.429499072356215"/>
          <point x="34.19387755102041" y="42.429499072356215"/>
        </location>
        <wall name="wall1">
          <point x="36.54916512059369" y="75.13914656771799"/>
          <point x="64.56400742115028" y="52.69016697588126"/>
          <height>3.5</height>
        </wall>
      </level>
      <way name="way1">
        <intermediatetarget>t1_Location</intermediatetarget>
      </way>
    </topology>
    <traffic name="default">
      <source name="s1">
        <location>s1_Location</location>
        <velocityprofile>DEFAULT_GUI_VELOPROFILE</velocityprofile>
        <tqm>default_source</tqm>
        <way probability="1.0">way1</way>
      </source>
      <target name="t1">
        <location>t1_Location</location>
        <tqm>default_target</tqm>
      </target>
      <tqm name="default_source">
        <timedata end="1000" numberpersons="3000" start="0"/>
      </tqm>
      <tqm name="default_target">
        <timedata end="1000" numberpersons="12000" start="0"/>
      </tqm>
      <velocityprofile name="DEFAULT_GUI_VELOPROFILE">
        <meanvelocity>1.34</meanvelocity>
        <deviation>0.26</deviation>
      </velocityprofile>
    </traffic>
  </scenario>
  <gui>
    <gui_info_for_level levelindex="0"/>
  </gui>
</simulation>

```

Abbildung A.1: XML-Datei aus einer Beispielsimulation des Simulators CrowdControl

A.2 Vorgang der Installation mit Einführung in das Plug-In

Benötigte Programme: Autodesk AutoCAD, Simulator CrowdControl Version 2.2 der Siemens AG

- AutoCAD öffnen
- In die Befehlsleiste den Befehl `_menuload` eingeben und die `cuix`-Datei `PedSim` der beiliegenden CD-Rom laden. Im Menü sollte nun ein Button mit der Aufschrift „PedSim“ erscheinen.

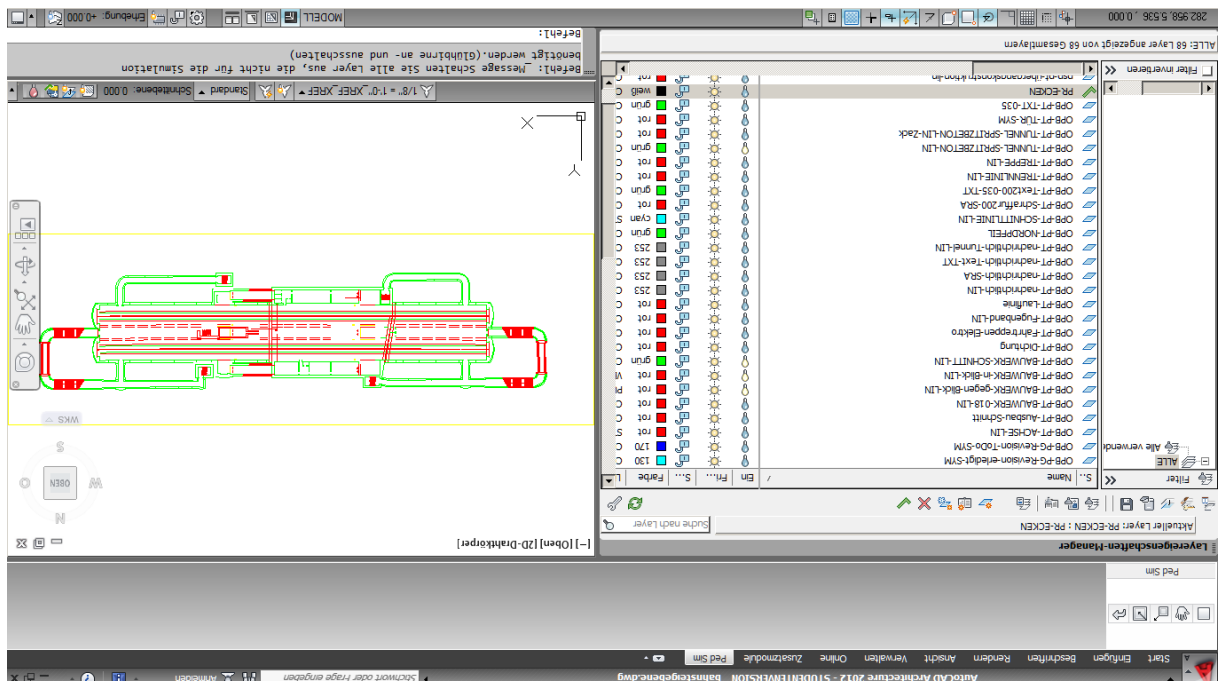


- Befehl `_netload` eingeben und die `dll`-Datei „Plug-In PedSim“ der beiliegenden CD-Rom laden. Die Befehle des Plug-Ins können nun verwendet werden.
- In der Menüleiste „PedSim“ wählen

- **Bereich wählen**


Dazu im Zeichenfenster einen Punkt links unten und einen Punkt rechts oben anklicken.

Nun sollte der Bildschirm in etwa so aussehen:



Im linken Fenster können Sie nun gewünschte Layer ein- und ausblenden.

Layer einblenden: 

Layer ausblenden: 


Das Layerfenster kann anschließend wieder geschlossen werden.

- **Wände wählen** 

Gewünschte Wände auswählen und mit „Enter“ oder rechtem Mausklick bestätigen. Die gewählten Objekte werden dadurch kopiert und im neu angelegten Layer „Wand PedSim“ abgespeichert.

- **Quellen wählen** 

Gewünschten Punkt angeben. Es sollte ein rotes Quadrat sichtbar werden.

- **Ziele wählen** 

Gewünschten Punkt angeben. Es sollte ein blaues Quadrat sichtbar werden.


- Daten mit dem Befehl ConvertToXML  an den gewünschten Ort speichern und einen Namen angeben
- Simulator CrowdControl öffnen
- Die erstellte XML-Datei mit dem Befehl File Open laden

- **Quellen und Ziele mit dem Befehl Add Way verbinden** 

Quelle mit linker Maustaste wählen

Ziel mit linker Maustaste wählen

Mit zwei Rechtsklicks auf der Maus bestätigen

- Das Szenario kann nun beliebig erweitert und verändert werden. Alle Funktionen sind sehr gut in dem Tutorial des Simulators beschrieben (Barti, Crowd Control - Pedestrian Stream Simulation - Tutorial, 2010)
- Simulation starten 

Anhang B

B.1 Compact Disc

Auf der beigefügten Compact Disc sind folgende Datenenthalten:

- Der schriftliche Teil der Arbeit als Worddokument
- Das Evakuierungsszenario der Bahnsteigebene als XML und Textdatei
- Die cuix-Datei Plugin Ped-Sim für die Benutzeroberfläche von AutoCAD
- Die dll-Datei PedSim Plug-In für das Plug-In im Ordner PedSim Plug-In
- Das Visual Studio Projekt PedSim Plug-In
- Der Ordner CUIx PedSim mit der Benutzeroberflächen für verschiedene CAD-Systeme
- Die PowerPoint Präsentation des Vortrags mit Video
- Videotutorial für das Plug-In

Literaturverzeichnis

- Österreichische Nationalbibliothek.* (9. Dezember 1881). Abgerufen am 22. Juli 2013 von <http://anno.onb.ac.at/cgi-content/anno?apm=0&aid=nfp&datum=18811209&seite=02>
- Wikipedia die freie Enzyklopädie.* (2004). Abgerufen am 29. Oktober 2013 von http://upload.wikimedia.org/wikipedia/commons/0/0f/423_gespiegelt.png?uselang=de
- AutoCAD Help.* (2013). Abgerufen am 28. Oktober 2013 von <http://docs.autodesk.com/ACD/2013/ENU/index.html?url=files/GUID-BF06F786-DDA6-4603-B5E5-25A35A4130A3.htm,topicNumber=d30e724088>
- Autodesk.* (26. Juli 2013). Abgerufen am 26. Juli 2013 von <http://www.autodesk.de/adsk/servlet/pc/index?siteID=403786&id=18369989>
- Dictionary.* (2013). Abgerufen am 31. Oktober 2013 von <http://dictionary.reference.com/browse/GUI>
- Duden.* (28. Oktober 2013). Abgerufen am 2013. Oktober 28 von Duden: <http://www.duden.de/rechtschreibung/Programmbibliothek>
- How Stuff Works.* (2013). Abgerufen am 29. Oktober 2013 von <http://science.howstuffworks.com/transport/engines-equipment/escalator1.htm>
- 3scale NetWorks S.L. (2011). *3scale.* Retrieved Oktober 28, 2013, from <http://www.3scale.net>: <http://www.3scale.net/wp-content/uploads/2012/06/What-is-an-API-1.0.pdf>
- Autodesk. (2013). *Autodesk - My First Plug-in Training.* Retrieved Juli 29, 2013, from <http://usa.autodesk.com/adsk/servlet/index?id=18162650&siteID=123112>
- Barti, D. E. (2010). *Crowd Control - Pedestrian Stream Simulation - Tutorial.* Siemens AG.
- Bray, T. (2006, September). *Extensible Markup Language (XML) 1.0 (Forth Edition) - Origin and Goals.* World Wide Web Consortium.
- Deutsche Bahn. (22. Juli 2013). *Projekt 2.stammstrecke-muenchen.* Abgerufen am 22. Juli 2013 von <http://www.2.stammstrecke-muenchen.de/das-projekt/die-2-stammstrecke/>
- Deutsche Bahn. (22. Juli 2013). *www.2.stammstrecke-muenchen.de.* Abgerufen am 22. Juli 2013 von <http://www.2.stammstrecke-muenchen.de/das-projekt/bahnhofe/>
- Diestel, R. (2010). *Graph Theory.* Berlin: Springer.
- Doberenz, W., & Gewinnus, T. (2013). *Visual Basic 2012 - Grundlagen und Profiwissen.* Wintersdorf: Carl Hanser Verlag München.

- Dörries, B. (24. Juli 2013). *Süddeutsche Zeitung*. Abgerufen am 24. Juli 2013 von <http://www.sueddeutsche.de/panorama/loveparade-in-duisburg-die-vermeidbare-katastrophe-1.1680573>
- Dr. Ludewig, & Dr. Schoser. (30. April 2009). *HOAI*. Abgerufen am 25. Juli 2013 von http://www.hoai.de/Bundesrat_Drucksache_395-09.pdf
- Duives, D. C., Daamen, W., & Hoogendoorn, S. P. (2013). State-of-the-art crowd motion simulation models. Stevinweg 1, 2628 CN Delft, The Netherlands: Delft University of Technology, Department of Transport & Planning.
- Franz Joseph I., & Graf Eduard Taaffe. (17. Januar 1883). *Österreichische Nationalbibliothek*. Abgerufen am 22. Juli 2013 von <http://alex.onb.ac.at/cgi-content/alex?apm=0&aid=lgn&datum=18830004&seite=00000051&zoom=2>
- Hauff, S. (22. Juli 2013). *Projekt 2.stammstrecke-muenchen*. Abgerufen am 22. Juli 2013 von <http://www.2.stammstrecke-muenchen.de/das-projekt/die-2-stammstrecke/>
- Interessengemeinschaft S-Bahn München e.V. (2008). Abgerufen am 29. Oktober 2013 von <http://www.igsbahn-muenchen.de/et423.shtml>
- Kneidl, A., Hartmann, D., & Borrmann, A. (2011). *Using a multi-scale model for simulating pedestrian behaviour*. Berlin Heidelberg: Springer-Verlag.
- Köster, G., Hartmann, D., & Klein, W. (2011). *Microscopic pedestrian simulations: From passenger exchange to regional evacuation*.
- Lackes, R., Siepermann, M., & Kollmann, M. (25. Oktober 2013). *Gabler Wirtschaftslexikon*. Abgerufen am 25. Oktober 2013 von <http://wirtschaftslexikon.gabler.de/Archiv/76201/plugin-v7.html>
- Meyer, B. (2008). *Encyclopedia of Complexity and System Science*. Berlin: Springer.
- Microsoft. (2013). *msdn*. Abgerufen am 29. Juli 2013 von <http://msdn.microsoft.com/de-de/library/system.xml.serialization.xmlserializer%28v=vs.80%29.aspx>
- Rank, E. (2010). Teil II: Ingenieur Anwendungen. In E. Rank, *Computerorientierte Methoden* (S. 66). München: Technische Universität München, Fakultät Bauingenieur- und Vermessungswesen, Lehrstuhl für Computation in Engineering.
- Rank, E. (2010). Teil III: Softwareentwicklung mit Visual Basic.NET. In E. Rank, *Computertorientierte Methoden*, (S. 72). München: Lehrstuhl für Computation in Engineering.
- Wohllaib, N. (2009). *Siemens*. Abgerufen am 23. Juli 2013 von http://www.siemens.com/innovation/apps/pof_microsite/_pof-fall-2009/_html_de/menschen-retten-mit-dem-rechner.html
- Zweifel, G. (2010). *Building Information Modeling - Potential für die digitale Kette*. Zürich: ETH Zürich.

Eidesstaatliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Bachelor-Thesis selbstständig angefertigt habe. Es wurden nur die in der Arbeit ausdrücklich benannten Quellen und Hilfsmittel benutzt. Wörtlich oder sinngemäß übernommenes Gedankengut habe ich als solches kenntlich gemacht.

Ich versichere außerdem, dass die vorliegende Arbeit noch nicht einem anderen Prüfungsverfahren zugrunde gelegen hat.

München, 6. Mai 2014

Heiner Kunkel

Heiner Kunkel