

# Diplomarbeit

*„Experimentelle und analytische Untersuchungen des  
Area Graph-based Mobility Models“*

Danny Tschirner

Betreuer:

Prof. Dr. H. Schweppe,

Prof. Dr.-Ing. habil. J. H. Schiller

M. Scholz

Institut für Informatik

Freie Universität Berlin

[tschirne@inf.fu-berlin.de](mailto:tschirne@inf.fu-berlin.de)

18.01.2006



# Inhaltsverzeichnis

<b>1 Einleitung.....</b>	<b>1</b>
1.1 Motivation.....	1
1.2 Ziel der Diplomarbeit.....	2
1.3 Überblick.....	2
1.4 System Modell.....	4
<b>2 Related Work.....</b>	<b>5</b>
2.1 Einfache Bewegungsmodelle.....	5
2.2 Komplexere Bewegungsmodelle.....	8
2.3 Zusammenfassung.....	12
<b>3 Beschreibung des Area Graph-based Mobility Models.....</b>	<b>13</b>
3.1 Basismodell.....	13
3.2 Erweiterungen.....	16
<b>4 Implementierung.....</b>	<b>20</b>
4.1 Sprache zur Beschreibung des Area Graph-based Mobility Models.....	20
4.2 Area Graph-based Mobility Model Simulator.....	22
4.3 Generierung der Bewegungsdateien für den Netzwerksimulator 2.....	25
<b>5 Verteilung der mobilen Knoten.....</b>	<b>27</b>
5.1 Analyse der Verweildauer.....	28
5.2 Analyse des Verbindungsgrades.....	38
5.3 Geschlossene Formel zur Vorhersage der Knotenverteilung.....	45
5.4 Beispiele zur Berechnung der mittleren Knotenanzahl.....	49
<b>6 Vorhersage der Nachrichtenverteilung.....</b>	<b>56</b>
6.1 Datenverteilung im Random Waypoint Mobility Model.....	57
6.2 Verteilung auf Geraden.....	66
6.3 Verteilung im Area Graph-based Mobility Model.....	69
6.4 Beispiele zur Berechnung der Nachrichtenverteilung.....	73

<b>7 Experimente mit Pull-Protokollen.....</b>	<b>77</b>
7.1 Verteilungsprotokolle.....	77
7.2 Experimentelle Untersuchungen.....	78
<b>8 Abschlussdiskussion und Ausblick.....</b>	<b>84</b>
8.1 Abschlussdiskussion.....	84
8.2 Ausblick.....	87
<b>9 Anhang.....</b>	<b>88</b>
9.1 Literaturverzeichnis.....	88

# 1 Einleitung

## 1.1 Motivation

Die mobile Kommunikation ist aus dem Alltag der Bevölkerung nicht mehr wegzudenken. Bereits 2002 nutzten weltweit über 1 Milliarde Menschen Mobilfunknetze [1]. Ein Ende des Trends ist kaum absehbar. Die Nutzung mobiler Geräte wie Handy, PDA (Personal Digital Assistant) oder Laptop verstärkt demnach das Auftreten von mobilen Ad-hoc Netzen (MANETs). In einem MANET können mobile Endgeräte, die sich in Funkreichweite befinden direkt eine Verbindung zueinander aufbauen, ohne dabei eine übergeordnete Struktur zu nutzen. Neben dem Empfangen und Senden eigener Daten, dient jedes Gerät zusätzlich als Router für andere Teilnehmer des Netzes. Das heißt, Teilnehmer die sich zueinander nicht in Funkreichweite befinden, können mit Hilfe anderer Endgeräte Daten austauschen. Da sich die Teilnehmer innerhalb des Netzes bewegen können, weisen MANETs in den meisten Fällen eine hohe Dynamik auf.

Existierende Routingprotokolle zur Datenverteilung in MANETs sind z.B. Dynamic Source Routing (DSR) [2] oder Ad Hoc On Demand Distance Vector (AODV) [3]. Um neue Protokolle für MANETs zu testen ist es in der Regel zu aufwendig und kostspielig entsprechende Szenarien real zu konstruieren. In den meisten Fällen wird hierfür auf Netzwerk-Simulatoren wie ns2 zurückgegriffen. Dazu wird das entsprechende Protokoll in einen Simulator integriert und im Anschluss in verschiedenen Szenarien simuliert. Anhand der Ergebnisse des Simulators kann dann das Protokoll verifiziert bzw. analysiert werden. Damit realistische Szenarien in Netzwerk-Simulatoren nachgestellt werden können, muss dieser in der Lage sein, die Bewegung mobiler Endgeräte zu simulieren. Eine exakte Abbildung der Bewegung aller Geräte ist in den meisten Fällen zu aufwendig, daher wird die Bewegung der Endgeräten mit Hilfe eines Bewegungsmodells simuliert. Solch ein Modell abstrahiert von der realen Bewegung und beschreibt diese in mathematischer Form.

Die am häufigsten verwendeten Bewegungsmodelle sind das *Random Waypoint Mobility Model* und das *Random Walk Mobility Model*. Diese zeichnen sich durch eine homogene Verteilung der Knoten aus. Das bedeutet, innerhalb des Simulationsbereiches sind die mobilen Endgeräte nahezu gleichmäßig verteilt. In der Realität existieren häufig heterogene Szenarien, aus diesem Grund sollten diese zur Analyse von Verteilungsprotokollen verwendet werden. Heterogen bedeutet hierbei, dass innerhalb des Modells Bereiche mit unterschiedlichen Dichten und verschiedenen Topologien existieren können. Neben den homogenen Bewegungsmodellen existieren auch komplexere Modelle wie das *City Section Mobility Model* oder das *Graph-Based Mobility Model*. Diese sind aber nur für sehr spezielle Szenarien anwendbar.

Ein Bewegungsmodell, welches den Entwurf vielfältiger, heterogener Szenarien ermöglicht, ist das *Area Graph-based Mobility Model* [4]. Mit diesem Modell können mehrere Bereiche definiert werden, in denen sich die mobilen Knoten frei bewegen können. In der Regel sind diese Bereiche mit einer hohen Dichte an mobilen Endgeräten, z.B. öffentliche Plätze oder Gebäude. Diese Bereiche sind zusätzlich über Wege miteinander verbunden, so können sich die mobilen Knoten zwischen den einzelnen Gebieten hin und her bewegen.

### 1.2 Ziel der Diplomarbeit

Im Rahmen dieser Arbeit wird das *Area Graph-based Mobility Model (AGbMM)* detailliert untersucht. Dazu wird einerseits ein Programm zur Simulation von *AGbMM* entwickelt und andererseits verschiedene Eigenschaften des Modells theoretisch analysiert werden. Der Schwerpunkt der Arbeit liegt auf der Analyse des *AGbMM*. Hierbei sollen sowohl grundlegende Attribute wie die Verteilung der mobilen Knoten innerhalb des Modells untersucht werden, als auch komplexere Attribute wie die Nachrichtenverteilung bei der Verwendung eines *AGbMM*.

Mit Hilfe der Analysen wird es möglich sein, unter Vorgabe eines *AGbMM*, die Aufteilung der mobilen Endgeräte auf die einzelnen Bereiche des Modells berechnen zu können. Bei der Nachrichtenverteilung wird für ein bekanntes Modell und eine festgelegte Startposition die Anzahl der Teilnehmer berechnet, die nach einer bestimmten Zeit die Nachricht bereits erhalten haben. Als Verteilungsprotokoll wird bei der Analyse einfaches Fluten untersucht. Alle Ergebnisse der theoretischen Untersuchungen werden fortwährend durch entsprechende Simulationen verifiziert.

Zusätzlich zu den theoretischen Untersuchungen mit einfachem Fluten, werden Untersuchungen mit verschiedenen Pull-Protokoll durchgeführt. Diese Ergebnisse werden anschließend mit Simulationen homogener Szenarien verglichen. Mit Hilfe dieser Resultate wird gezeigt, dass die Wahl eines Bewegungsmodells entscheidend für die Analyse eines Verteilungsprotokolls ist.

### 1.3 Überblick

Dieser Abschnitt gibt einen Überblick über die verschiedenen Kapitel der vorliegenden Arbeit. Das 2. Kapitel führt in die Thematik der existierenden Bewegungsmodelle ein. Kapitel 3 stellt das zu untersuchende Bewegungsmodell, das *Area Graph-based Mobility Model (AGbMM)*, vor. Zum einen erfolgt eine Beschreibung des Basismodells, zum anderen werden verschiedene Erweiterungen für das *AGbMM* vorgestellt. Kapitel 4 beschreibt die verschiedenen Implementierungen, die im Verlauf der Diplomarbeit entwickelt werden.

Der Schwerpunkt der Arbeit, die Analyse, beginnt im Kapitel 5. Hier wird die Verteilung der mobilen Knoten in einem *AGbMM* analysiert. Das bedeutet, es wird untersucht wie sich

## Einleitung

---

die mobilen Knoten auf die einzelnen Bereiche des *AGbMM* aufteilen. Dazu werden verschiedene Attribute des *AGbMM* und deren Auswirkung auf die Knotenverteilung genauer betrachtet. Anhand dieser Untersuchungen leiten sich die Formeln zur Berechnung der durchschnittlichen Knotenanzahl in bestimmten Bereichen des *AGbMM* ab. Der zweite Teil der Analyse in Kapitel 6 untersucht die Nachrichtenverteilung in einem *AGbMM* unter Verwendung des einfachen Fluten. Es wird zunächst die Verteilung innerhalb der Bereiche mit hoher Dichte und anschließend die Verteilung auf den Wegen analysiert. Im Anschluss werden die Ergebnisse zusammengeführt, um die Nachrichtenverteilung im *AGbMM* vorhersagen zu können.

Im 7. Kapitel werden Simulationen mit verschiedenen Pull-Protokoll unter Verwendung des *AGbMM* durchgeführt und diese Ergebnisse mit homogenen Bewegungsmodellen verglichen. Das letzte Kapitel diskutiert die durchgeführten Arbeiten und gibt einen Ausblick auf verschiedene weiterführende Untersuchungen mit dem *AGbMM*.

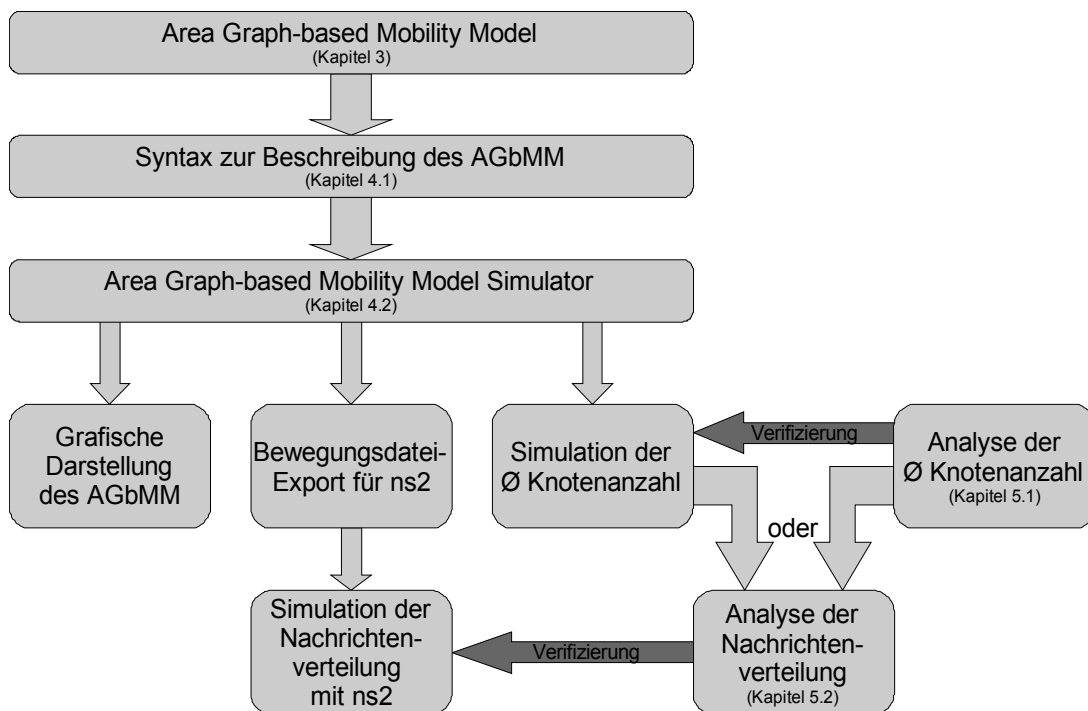


Abbildung 1: Überblick über die Module der Diplomarbeit

Einen Überblick über die durchgeführten Arbeiten und entwickelten Module ist in Abbildung 1 dargestellt. Sie zeigt, welche Module bzw. Arbeiten aufeinander aufbauen und in welchem Kapitel der Diplomarbeit diese beschrieben werden. Es ist zu erkennen, dass für die Entwicklung eines Simulators für *AGbMM* zunächst eine Syntax entworfen werden muss um diese Modelle beschreiben zu können. Die Abbildung zeigt außerdem, dass der zweite theoretische Abschnitt, die Analyse der Nachrichtenverteilung in Kapitel 6, die

Verteilung der mobilen Knoten innerhalb eines Modells benötigt. Diese Verteilung kann entweder durch Simulation mit dem entwickelten Simulator bestimmt oder mittels der Ergebnisse aus Kapitel 5 berechnet werden. Die dunklen Pfeile geben an, mit Hilfe welcher Module, sich die Ergebnisse der Analysen verifizieren lassen.

### **1.4 System Modell**

Ein denkbare Szenario eines MANETs sind Studenten auf einen Campus welche untereinander Informationen austauschen. Jeder Teilnehmer verfügt dazu über ein mobiles Endgerät, etwa ein Handy, PDA oder Laptop. Diese Geräte sind in der Lage untereinander Verbindungen aufzubauen und somit Informationen auszutauschen. Im Rahmen dieser Arbeit wird zur Nachrichtenverteilung WLAN (IEEE 802.11) als Technik verwendet. Es wird angenommen, dass alle Modelle über eine feste Anzahl von Teilnehmer verfügen. Das bedeutet, während der Simulation tritt kein Teilnehmer dem Modell bei oder verlässt selbiges. Bei der Simulation der Nachrichtenverteilung von mobilen Endgeräten, wird zusätzlich angenommen, dass alle Endgeräte über genügend Speicher- und Energiereserven verfügen, so dass es zu keinerlei Ausfällen kommt.



## 2 Related Work

Zur Simulation verschiedenster Bewegungen von Menschen, Fahrzeugen, Elementarteilchen oder beliebigen anderen Gegenständen existiert eine Vielzahl an Modellen. All diese Bewegungsmodelle haben Eines gemeinsam: Sie abstrahieren von bestimmten realen Bewegungen und/oder Szenarien und beschreiben diese in mathematischer Form. Erste Bewegungsmodelle, wie das *Random Walk Mobility Model*, wurden bereits im frühen 20. Jahrhundert zur Beschreibung der Bewegung von Elementarteilchen eingesetzt [5]. Andere Modelle wurden entwickelt, um die Bewegung von Menschengruppen innerhalb eines festgelegten Szenarios zu simulieren [5][6]. Seit einigen Jahren werden Bewegungsmodelle aber auch im Bereich der Mobilkommunikation eingesetzt und untersucht. Zum Beispiel müssen zur Verifizierung und Analyse von Verteilungsprotokollen, Netze mit einer hinreichenden Anzahl mobiler Endgeräte aufgebaut werden. Weil dieser Aufbau in den meisten Fällen zu aufwendig ist, werden MANETs im Allgemeinen simuliert. Ein Bewegungsmodell bildet dabei die Bewegung der mobilen Endgeräten nach.

Dieses Kapitel gibt einen Überblick über die Bewegungsmodelle, die für analytische Untersuchungen und die Simulation von MANETs verwendet werden. Viele Modelle wurden speziell zur Simulation von MANETs entwickelt. Andere Bewegungsmodelle, wie z.B. das *Random Walk Mobility Model*, wurden für andere Bereiche entworfen, in den letzten Jahren aber verstärkt zur Simulation von MANETs verwendet. Die Bewegungsmodelle in diesem Kapitel werden in zwei Kategorien eingeteilt: Zum einen die einfachen Bewegungsmodelle und zum anderen komplexere Bewegungsmodelle. Die einfachen Bewegungsmodelle zeichnen sich durch eine homogene Struktur aus. Im Gegensatz dazu beschreiben die komplexen Modelle sehr spezielle Topologien und sind somit oft nur für bestimmte Szenarien anwendbar. Das mit Abstand am häufigsten verwendete Modell ist das *Random Waypoint Mobility Model*.

### 2.1 Einfache Bewegungsmodelle

In diesem Abschnitt werden die drei folgenden, einfachen Bewegungsmodelle genauer vorgestellt: Das *Random Walk Mobility Model*, das *Random Waypoint Mobility Model* und das *Random Direction Mobility Model*. Alle Modelle werden in [5] beschrieben. Aufgrund der einfachen, homogenen Struktur werden diese Modelle häufig zur Simulation und Verifikation von Verteilungs- bzw. Routingprotokollen in MANETs genutzt.

Das *Random Walk Mobility Model* wurde bereits 1926 von Albert Einstein mathematisch beschrieben. Es sollte sprunghafte und nicht vorhersagbare Bewegungen, wie sie häufig in der Natur auftreten, nachahmen (Brownsche Bewegung). Die Knoten laufen hierbei von ihrer Aktuellen zu einer neuen Position innerhalb einer rechteckigen Fläche. Hierzu wird

## Related Work

---

eine zufällige Geschwindigkeit und ein zufälliger Winkel für die Bewegung des Knotens bestimmt. Der Winkel wird dabei gleichverteilt aus dem Intervall  $[0, 2\pi]$  und die Geschwindigkeit aus dem Intervall  $[v_{\min}, v_{\max}]$  gewählt. Jede Änderung der Bewegung erfolgt entweder nach einer bestimmten Zeit oder nach einer festgelegten Entfernung. Wenn ein Knoten während der Bewegung an den Rand der Fläche gelangt, ändert er seine Richtung und läuft weiter. Die neue Richtung wird hierbei durch den Eintrittswinkel der Bewegung bestimmt.

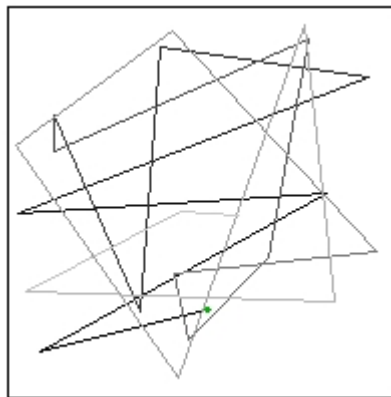


Abbildung 2: Bewegung eines mobilen Knotens im  
*Random Waypoint Mobility Model*

Das am häufigsten verwendete Modell zu Simulation von MANETs ist das *Random Waypoint Mobility Model*. Dieses Modell erweitert das *Random Walk Mobility Model* um Wartezeiten zwischen den einzelnen Richtungsänderungen der Bewegung. Außerdem wird beim *Random Waypoint Mobility Model* nicht mehr die Bewegungsrichtung zufällig gewählt. Vielmehr wählt der Knoten eine zufällige Position im Simulationsbereich und bewegt sich auf diese zu. Dazu wählt der Knoten eine zufällige Geschwindigkeit aus dem Intervall  $[v_{\min}, v_{\max}]$  mit der er sich in Richtung Zielposition bewegt. Hat der Knoten sein Ziel erreicht, wartet er dort einen zufälligen Zeitraum, dieser ist gleichverteilt aus dem Intervall  $[t_{\min}, t_{\max}]$ . Nach Ablauf der Wartezeit wählt der Knoten eine neue Zielposition und der Vorgang beginnt von vorn. Ein Beispiel für die Bewegung eines einzelnen Knotens entsprechend dem *Random Waypoint Mobility Model* ist in Abbildung 2 zu sehen. Dabei kennzeichnen hellere Strecken, zeitlich weiter zurückliegenden Bewegungen.

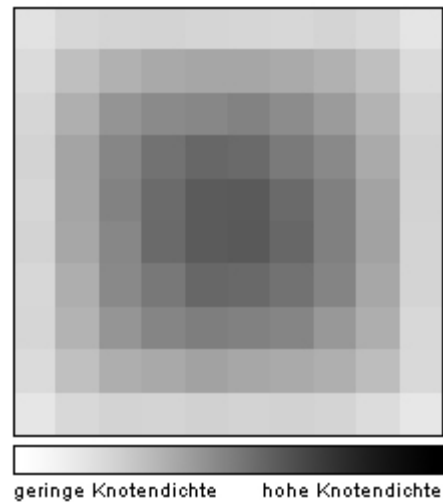


Abbildung 3: Knotenverteilung im  
*Random Waypoint Mobility Model*

Die meisten Simulationen, die das *Random Waypoint Mobility Model* benutzen, positionieren die Knoten zu Beginn der Simulation gleichverteilt über die gesamte Fläche. Diese Aufteilung der Knoten entspricht aber nicht der Verteilung, die durch die Bewegung der Knoten im Modell entsteht. Das heißt, wenn die Knoten zu Beginn der Simulation gleichmäßig verteilt werden, muss zuerst eine gewisse Zeit simuliert werden, bis sich die Knoten entsprechend des Modells verteilt haben. Ein Beispiel für die Knotenverteilung entsprechend des *Random Waypoint Mobility Models* zeigt Abbildung 3. Dazu wurde die Bewegung von 500 Knoten für 30 Minuten simuliert. Je dunkler ein Bereich des Bildes ist, umso mehr Knoten befinden sich im Mittel in diesem Ausschnitt. Man erkennt, dass sich im Zentrum der Fläche mehr Knoten befinden als an den Rändern des Simulationsbereiches. Das folgt daraus, weil die Bewegung der Knoten mit einer höheren Wahrscheinlichkeit entlang des Mittelpunktes verläuft als entlang eines anderen Bereichs des Modells. Diese ungleichmäßige Verteilung der mobilen Knoten entspricht aber in vielen Bereichen der Realität. So bewegen sich Menschen, auf großen Plätzen, mit einer höheren Wahrscheinlichkeit entlang des Mittelpunktes des Platzes. Gleichermäßen ist in Messe- oder anderen Hallen die Konzentration von Menschen im Zentrum höher als an den Rändern der Räume.

Ein Bewegungsmodell bei dem die Häufung der mobilen Knoten im Zentrum vermieden werden soll, ist das *Random Direction Mobility Model* [7]. Wie beim *Random Walk Mobility Model*, wählen die Knoten hierbei eine zufällige Richtung aus dem Intervall  $[0, 2\pi]$ . Der Knoten geht dann entlang dieser Richtung bis er an den Rand des Simulationsbereiches gelangt. Dort wartet er eine gewisse Zeit und wählt wieder eine Richtung in die er geht. Eine Erweiterung des *Random Direction Mobility Model* ist das *Modified Random Direction Mobility Model*, hierbei sind die Knoten nicht mehr gezwungen bis zum Rand des Modells

zu laufen, sondern können vorher anhalten und ihre Richtung ändern. Der Knoten wartet dann an diesem Punkt und wählt anschließend eine neue Richtung.

Die drei vorgestellten einfachen Modelle haben folgende Gemeinsamkeiten: alle sind gedächtnislos, haben eine rechteckige Simulationsfläche und sie bewegen die Knoten unabhängig voneinander über die gesamte Simulationsfläche. Gedächtnislos heißt hierbei, dass die Bewegung eines Knotens unabhängig von der vorherigen Bewegung ausgeführt wird, dadurch kann es zu abrupten Richtungsänderungen bei der Bewegung kommen (siehe Abbildung 2). Die Form der Simulationsfläche ist nicht entscheidend für die Kategorie der Modelle, denn ohne viel Mehraufwand könnten die Modelle auch eine beliebige konvexe Form haben. Vielmehr ist die Bewegung der Knoten innerhalb der Fläche ausschlaggebend für die Bezeichnung „einfache Bewegungsmodelle“. Denn bei allen drei Modellen können sich die Knoten über die gesamte Fläche frei bewegen, ungeachtet dessen, dass bestimmte Bereiche der Fläche höhere Dichten aufweisen als andere. Im folgenden Abschnitt werden drei Modelle präsentiert, bei denen nicht mehr jede Position der Simulationsfläche für die Knoten erreichbar ist.

## 2.2 Komplexere Bewegungsmodelle

Bei den im vorherigen Abschnitt vorgestellten Modellen bewegen sich die Knoten über die gesamte Simulationsfläche. Diese Art der Bewegung entspricht jedoch kaum der Realität, denn Menschen bewegen sich auf vorgegebenen Wegen oder fahren mit Autos auf Straßen. Das heißt, meistens gibt es für die Bewegung bestimmte Restriktionen und nicht jeder kann und wird sich überall hin frei bewegen. In diesem Abschnitt werden drei Bewegungsmodelle vorgestellt, die die Bewegung der Knoten innerhalb der Simulationsfläche einschränken. Dabei sind die einzelnen Modelle jeweils für sehr spezielle Szenarien anwendbar. So beschreibt etwa das *City Section Mobility Model* die Bewegung von Menschen oder Fahrzeugen auf einem Straßennetz. Das *Graph-Based Mobility Model* bewegt die Knoten auf einem Graphen wie auch das *Obstacle Mobility Model*, dieses verfügt zusätzlich über Hindernisse, durch welche sich die Knoten auf vordefinierten Wegen bewegen können.

Das *City Section Mobility Model* wird in [8] beschrieben. Die Simulationsfläche repräsentiert ein Straßennetz, welches den Ausschnitt einer Stadt darstellt, etwa in Form eines Rasters wie es von amerikanischen Großstädten bekannt ist. Die Anordnung der Straßen und deren Geschwindigkeitsbegrenzungen ergeben sich aus dem zu simulierenden Szenario. Vor Beginn der Simulation wird jeder Knoten auf einer zufälligen Straße platziert. Anschließend wählt jeder Knoten einen Zielpunkt (eine Kreuzung auf dem Raster) und nimmt den schnellsten Weg dorthin. Weil sich die Geschwindigkeiten der Straßen unterscheiden, muss hierbei der kürzeste Weg nicht zwingend der schnellste Weg sein. Hat der Knoten sein Ziel erreicht, wartet er dort eine zufällige Zeit, sucht sich ein neues Ziel und der Vorgang beginnt von vorn.

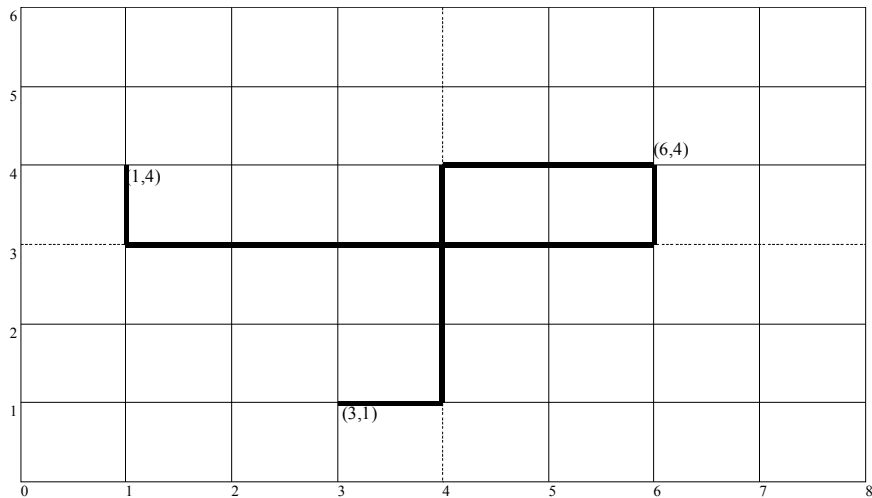


Abbildung 4: City Section Mobility Model

Ein Beispiel eines *City Section Mobility Models* und die dazugehörige Bewegung eines mobilen Knotens zeigt Abbildung 4. Die Straßen bilden ein Raster aus 7 horizontalen und 9 vertikalen Linien. Die jeweils mittleren Straßen (5. Vertikale und 4. Horizontale) haben eine doppelt so hohe zulässige Geschwindigkeit wie die anderen Straßen. Der Knoten wird anfänglich auf Position (3,1) platziert und wählt als erstes Ziel den Punkt (6,4). Der Knoten bewegt sich dazu zuerst von (3,1) nach (4,1), anschließend von (4,1) nach (4,4) und erreicht das Ziel (6,4) ausgehend vom Punkt (4,4). Diese Strecke wird gewählt, weil hierbei der Zeitaufwand am geringsten ist. Denn auf dem Abschnitt von (4,1) nach (4,4) ist die Geschwindigkeit doppelt so hoch wie auf den anderen Straßen. Deutlicher wird es bei der Strecke von (6,4) nach (1,4). Hierzu fährt der Knoten eine länger Strecke, um an das Ziel zu gelangen, ist aber wegen der höheren Geschwindigkeit auf der mittleren Straße immer noch schneller, als würde er den direkten Weg von (6,4) nach (1,4) nutzen. Um mehr Szenarien abbilden zu können, wurde das *City Section Mobility Models* um folgende Eigenschaften erweitert: Einfügen von Pausen an beliebigen Kreuzungen, Verteilung der Knoten in Abhängigkeit der Tageszeit und Berücksichtigung der Beschleunigung bei der Bewegung von Knoten.

Obwohl das *City Section Mobility Model* die Bewegung der Knoten realistischer abbildet als die einfachen Bewegungsmodelle, so ist es doch nur für sehr spezielle Szenarien anwendbar. Flexiblere Topologien lassen sich mit dem *Graph-Based Mobility Model* simulieren. Dieses wurde in [9] vorgestellt. Die Art der Bewegung der Knoten ist ähnlich dem *City Section Mobility Model*, aber nicht auf die horizontale und vertikale Richtung beschränkt. Das *Graph-Based Mobility Model* nutzt einen Graphen zur Beschreibung der Bewegung. Die Knoten des Graphen stellen Sehenswürdigkeiten oder andere Plätze, wie z.B. Einkaufszentren, dar. Die Kanten des Graphen beschreiben die Verbindungen wie Straßen

## Related Work

---

oder Bahnverbindungen zwischen den Orten. Der Graph ist zusammenhängend, d.h. es existiert eine Verbindungen zwischen allen Knoten des Graphen.

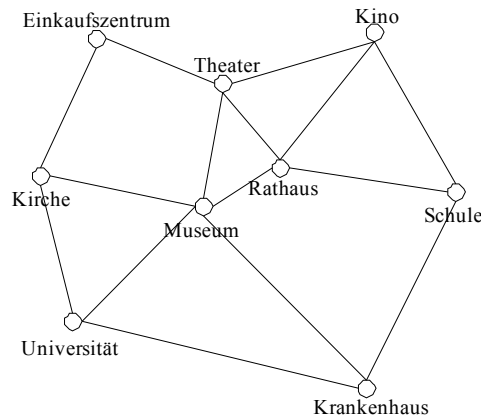


Abbildung 5: Graph-Based Mobility Model

Die mobilen Knoten werden zu Beginn zufällig auf die Sehenswürdigkeiten verteilt. Anschließend bewegt sich jeder Knoten zu einer anderen Sehenswürdigkeit, dazu wird ein anderer zufälliger Knoten des Graphen als Ziel bestimmt und der kürzeste Weg dorthin gewählt. Am Ziel angekommen wartet der Knoten für eine gewisse Zeit und sucht sich ein neues Ziel. Abbildung 5 zeigt einen Graphen, der die Sehenswürdigkeiten einer Stadt nachbildet wie er vom *Graph-Based Mobility Model* verwendet wird.

Im Vergleich zum *City Section Mobility Model* lassen sich mit dem *Graph-Based Mobility Model* mehr Szenarien nachbilden. Es erlaubt zudem eine flexiblere Bewegung der Knoten, weil diese nicht mehr auf ein Raster wie beim *City Section Mobility Model* beschränkt ist. Von Nachteil ist aber die Darstellung der Sehenswürdigkeiten als Knoten des Graphen und somit als feste Punkte in der Ebene. Hierdurch ist die Position aller Knoten, die zu einer bestimmten Zeit die gleiche Sehenswürdigkeit besuchen, die Gleiche. Das heißt, alle Knoten die sich an einer Sehenswürdigkeit aufhalten, befinden sich in Funkreichweite zueinander, da ihre Entfernung Null ist. Diese Annahme ist aber gerade bei Szenarien mit großen Gebäuden und geringen Funkreichweiten ungeeignet.

Sowohl das *City Section Mobility Model* als auch das *Graph-Based Mobility Model* berücksichtigen keine Hindernisse (wie etwa Gebäude) für die Bewegung und/oder die Funksignale, vielmehr bewegen sie die Knoten nur auf vorgegebenen Wegen über die Simulationsfläche. Beim dritten komplexeren Bewegungsmodell werden die Wege, auf denen sich die Knoten bewegen, durch Hindernisse, genauer Gebäude, bestimmt. Das *Obstacle Mobility Model* wurde in [10] vorgestellt und besteht aus drei Teilen: den Hindernissen, dem Wege-Graph und der Wege-Auswahl. Die Hindernisse oder Gebäude können die Form eines beliebigen Polygons haben. Andere Formen wie z.B. Kreise lassen sich somit durch Polygone mit einer großen Anzahl an Eckpunkten beliebig genau approximieren. In [10] werden nur Hindernisse in der Form von Rechtecken verwendet.

## Related Work

---

Sind die Hindernisse festgelegt, lässt sich aus ihnen der Wege-Graph des *Obstacle Mobility Models* bestimmen. Dazu werden alle Eckpunkte der Hindernisse als Punkte auf einer Ebene genommen und aus diesen Punkten ein Voronoi Graph erstellt. Als Knoten des Wege-Graphen werden zum einen die Knoten des Voronoi Graphen und zum anderen alle Schnittpunkte der Voronoi Kanten mit dem Rand der Simulationsfläche und alle Schnittpunkte von Voronoi Kanten mit den Hindernissen (sozusagen die Türen der Gebäude) übernommen.

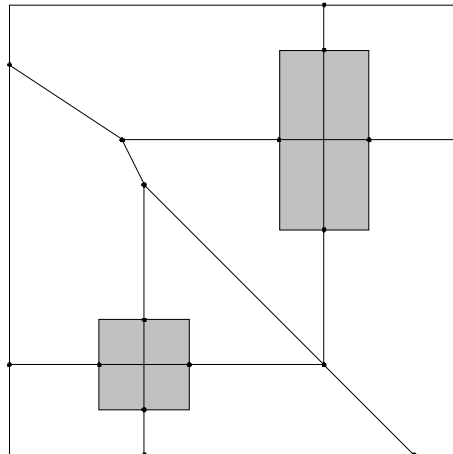


Abbildung 6: *Obstacle Mobility Model*

Ein Beispiel für einen Wege-Graphen ist in Abbildung 6 dargestellt, dabei stellen die grauen Rechtecke die Hindernisse dar. Es ist zu erkennen, dass durch die Konstruktion des Voronoi Graphen die Wege genau in der Mitte zwischen zwei Gebäuden und durch die Gebäude selbst verlaufen. Der letzte Teil des *Obstacle Mobility Models* ist die Wege-Auswahl, in [10] wurde dafür die kürzeste Wege Strategie verwendet. Das heißt, die Knoten laufen immer auf dem kürzesten Weg zu ihrem nächsten Ziel. Zu Beginn der Simulation werden die Knoten zufällig auf den Wegen positioniert. Anschließend verläuft die Bewegung der Knoten wie folgt: jeder mobile Knoten wählt einen Knoten des Wege-Graphen als Ziel, hiernach bewegt er sich auf dem kürzesten Weg zum Ziel, dort angekommen wartet er eine gewisse Zeit und der Vorgang beginnt von vorn.

Bei allen hier vorgestellten komplexen Bewegungsmodellen können sich die Knoten nicht mehr über den gesamten Simulationsbereich bewegen, vielmehr bewegen sich die Knoten auf vordefinierten Wegen von einem Ort zum Nächsten. Dabei kann bei allen Modellen der Weg aus mehreren Streckenabschnitten bestehen. Wie bei den einfachen Modellen ist die Bewegung der Knoten weiterhin unabhängig von der Bewegung anderer Knoten und gedächtnislos, d.h. jeder Knoten bewegt sich ohne Berücksichtigung der vorherigen Bewegung.

Neben den hier vorgestellten Bewegungsmodellen (*City Section Mobility Model*, *Graph-Based Mobility Model* und *Obstacle Mobility Model*) gibt es weitere Modelle, die für spezielle Szenarien oder Bewegungen anwendbar sind. In [11] wird z.B. ein Modell vorgestellt, bei dem die Knoten ihre Richtung nicht mehr abrupt ändern wie etwa beim *Random Waypoint Mobility Model*. Hier bewegen sich die Knoten in einer Kurve auf ihr nächstes Ziel zu. Auch zur Bewegung mehrerer Knoten in Abhängigkeit von einander gibt es verschiedene Ansätze (*Group Mobility Models*). In [5] werden hierzu fünf Bewegungsmodelle vorgestellt, die jeweils eine Gruppe von Knoten in Abhängigkeit zueinander bewegen. Die verschiedenen *Group Mobility Models* werden hier aber nicht detaillierter vorgestellt, weil sie kaum Relevanz bei der Simulation von MANETs haben.

### 2.3 Zusammenfassung

In diesem Abschnitt wurden sowohl homogene Bewegungsmodelle wie das *Random Walk Mobility Model*, das *Random Waypoint Mobility Model* und das *Random Direction Mobility Model* als auch komplexe Bewegungsmodelle wie das *City Section Mobility Model*, das *Graph-Based Mobility Model* und das *Obstacle Mobility Model* vorgestellt. Obwohl die homogenen Bewegungsmodelle die Realität kaum abbilden, werden diese am häufigsten zur Simulation und Analyse von Verteilungsprotokollen verwendet, insbesondere das *Random Waypoint Mobility Model*. Das liegt einerseits daran, dass die Modelle relativ einfach umzusetzen und zu analysieren und andererseits in vielen Netzwerk-Simulatoren bereits implementiert sind. Die komplexen Bewegungsmodelle beschreiben die Realität detaillierter, sind aber häufig nur für sehr spezielle Szenarien anwendbar. So kann mit dem *City Section Model* zwar eine amerikanische Innenstadt simuliert werden, aber zur Simulation der Bewegung von Studenten auf einem Campus ist dieses Modell eher ungeeignet.

Es wird also ein Bewegungsmodell benötigt, mit dem sich die Realität gut abbilden lässt, um so für eine Vielzahl von Szenarien anwendbar zu sein. Die Knoten sollten heterogen verteilt werden, d.h. es sollte Bereiche unterschiedlicher Dichten geben. Außerdem muss das Modell einfach anzuwenden sein und entworfene Szenarien ohne viel Aufwand erweitert werden können. Ein Modell welches über all diese Eigenschaften verfügt, ist das *Area Graph-based Mobility Model*. Dieses Bewegungsmodell wird im nächsten Kapitel vorgestellt und detailliert beschrieben.



### 3 Beschreibung des Area Graph-based Mobility Models

Wie im vorherigen Kapitel beschrieben, gibt es eine Vielzahl verschiedener Bewegungsmodelle. Der Nachteil all dieser Modelle ist, dass sie die Realität kaum bzw. nur sehr unpräzise abbilden. So bewegen sich beim *Random Waypoint Mobility Model* die mobilen Knoten zufällig auf einer rechteckigen Fläche, was nur für sehr wenige Szenarien realistisch ist, z.B. die Bewegung von Menschen auf einem öffentlichen Platz. Bei den komplexen Modellen (wie z.B. dem *City Section Mobility Model*, dem *Graph-Based Mobility Model* oder dem *Obstacle Mobility Model*) bewegen sich die Knoten auf Geraden, z.B. auf einem Raster oder auf einem Graphen. Die Bereiche mit höherer Dichte, wie etwa die Sehenswürdigkeiten beim *Graph-Based Mobility Model*, sind immer einzelne Punkte in der Ebene. Das bedeutet, alle Knoten, die sich zur gleichen Zeit an einer bestimmten Sehenswürdigkeit befinden, haben die gleichen Koordinaten und zwar so lange, bis sie sich von der Sehenswürdigkeit fortbewegen. Das entspricht nicht der Realität, denn meistens existieren größere Bereiche oder Flächen mit einer hohen Dichte, wie etwa öffentliche Plätze oder Einkaufszentren und diese sind über Wege miteinander verbunden. Um solche heterogenen Szenarien beschreiben und simulieren zu können, wurde das *Area Graph-based Mobility Model (AGbMM)* entwickelt. In diesem Abschnitt wird zum einen das Basismodell vorgestellt, wie es erstmals in [4] beschrieben wurde. Zum anderen werden Erweiterungen des *AGbMM* vorgestellt, die während der Arbeit entstanden sind. Diese Erweiterungen machen das *AGbMM* flexibler und demnach für mehr Szenarien anwendbar. Um Verwechslungen zwischen den mobilen Knoten und den Knoten eines Graphen zu vermeiden, wird im weiteren Verlauf der Arbeit der Begriff Cluster als Synonyme für die Knoten des Graphen eines *AGbMM* verwendet.

#### 3.1 Basismodell

Das *AGbMM* wurde zum ersten Mal in [4] vorgestellt und beschrieben. Das Modell basiert auf einem gerichteten und gewichteten Graphen  $G=(V,E)$ . Das Modell besteht aus einer Menge von Clustern  $V$ , welche die Knoten des Graphen  $G$  repräsentieren und als rechteckige Flächen dargestellt sind. Außerdem gibt es eine Menge von Verbindungen  $E$ , welche jeweils zwei Cluster miteinander verbinden und die Kanten des Graphen darstellen. Die Gewichtung  $w(e)$  einer Kante  $e$  des Graphen  $G$  entspricht der Wahrscheinlichkeit, dass ein mobiler Knoten über diese Kante das entsprechende Cluster verlässt. Das heißt, das Gewicht  $w(e)$  einer Kante  $e$  ist eine Zahl zwischen 0 und 1 und die Summe aller ausgehenden Kanten eines Clusters ist immer 1. Jeder Knoten des Graphen hat ein Zeitintervall  $[t_{min}, t_{max}]$  als Attribut, womit die Aufenthaltsdauer eines mobilen Knoten innerhalb dieses Clusters bestimmt wird. Betritt ein mobiler Knoten ein Cluster, wird

## Beschreibung des Area Graph-based Mobility Models

---

gleichverteilt aus dem Intervall  $[t_{min}, t_{max}]$  ein Wert bestimmt, der dann als Aufenthaltsdauer für diesen Knoten innerhalb des Clusters dient.

Die Cluster repräsentieren Bereiche mit höherer Dichte, in denen sich die mobilen Knoten frei bewegen können. Die Kanten stellen Wege oder Straßen dar, die die Cluster miteinander verbinden. Die Simulation der Bewegung eines mobilen Knotens besteht aus zwei Teilen: Die Bewegung innerhalb eines Clusters und die Bewegung auf den Kanten. Innerhalb der Cluster bewegen sich die Knoten entsprechend dem *Random Waypoint Mobility Model* mit einer zufälligen Geschwindigkeit aus einem Intervall  $[v_{min}, v_{max}]$ . Nach Ablauf der Aufenthaltsdauer eines Knotens innerhalb eines Clusters, verlässt er dieses mit einer zufälligen Geschwindigkeit und läuft entlang einer zufällig gewählten, inzidenten Kante zum benachbarten Cluster. Dabei entscheiden die Gewichtungen, welche der Kanten genutzt wird und ein Intervall  $[v_{min}, v_{max}]$ , mit welcher Geschwindigkeit sich die mobilen Knoten auf der Kante bewegen. Sobald der Knoten das neue Cluster betritt, wird ihm für dieses eine zufällige, gleichverteilte Aufenthaltsdauer zugewiesen. Anschließend bewegt sich der Knoten erneut entsprechend dem *Random Waypoint Mobility Model*, bis seine Aufenthaltsdauer in diesem Cluster abläuft und sich der Vorgang wiederholt.

Neben der Aufenthaltsdauer, der Position und der Dimension der Fläche verfügt jeder Knoten des Graphen (Cluster) außerdem noch über weitere Attribute. Diese beschreiben das interne *Random Waypoint Mobility Model*, dazu gehören: die minimale bzw. maximale Geschwindigkeit eines mobilen Knotens und die minimale bzw. maximale Wartezeit eines Knotens bis zur nächsten Bewegung. Die Knoten wählen also nach betreten eines Clusters eine zufällige Position und bewegen sich auf diese mit zufälliger und gleichverteilter Geschwindigkeit zu. Am Ziel angekommen, wartet der mobile Knoten eine zufällige, gleichverteilte Zeit und wählt anschließend eine neue Zielposition. Die Kanten des Graphen verfügen ebenfalls über ein Intervall mit minimaler und maximaler Geschwindigkeit, aus dem zufällig und gleichverteilt die Geschwindigkeit bestimmt wird, mit der sich ein Knoten auf dieser Kante bewegt.

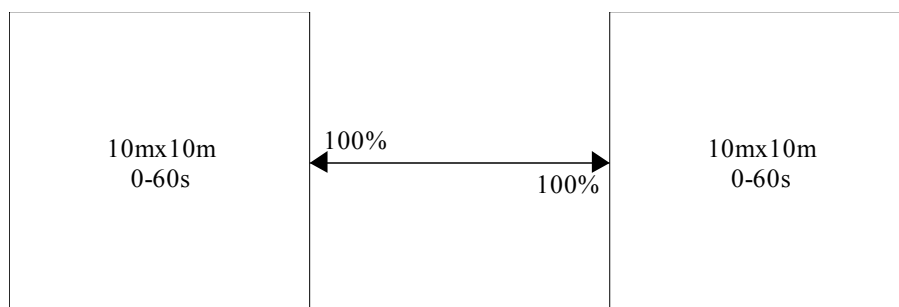


Abbildung 7: Area Graph-based Mobility Model

Abbildung 7 zeigt ein einfaches Beispiel eines *AGbMM*. Der Graph besteht aus zwei Clustern (die zwei quadratischen Flächen) und zwei gerichteten Kanten. Die Cluster haben

## Beschreibung des Area Graph-based Mobility Models

eine Länge und Breite von jeweils 10 Metern und die Aufenthaltsdauer eines Knoten innerhalb eines Clusters beträgt immer zwischen 0 bis 60 Sekunden. Die Gewichtung der beiden Kanten ist jeweils 1, weil jedes Cluster genau eine inzidente Kante hat. Zusätzlich sei die Geschwindigkeit der mobilen Knoten sowohl auf den Kanten als auch innerhalb der Cluster genau 1m/s. Die Wartezeit der Knoten, zwischen den einzelnen Bewegungen innerhalb eines Clusters, beträgt 0 bis 10 Sekunden.

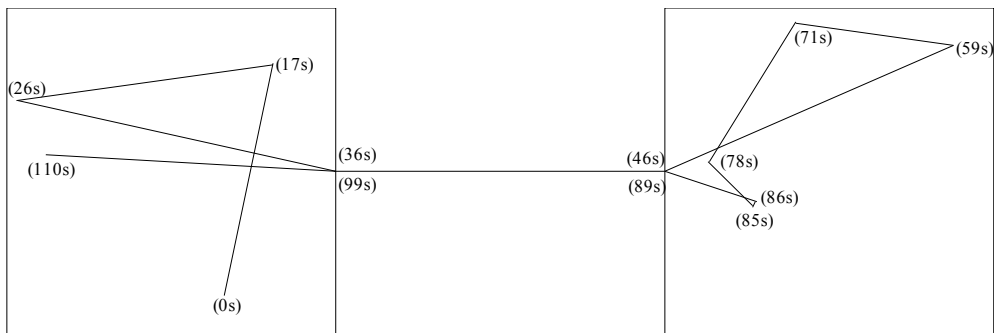


Abbildung 8: Bewegung eines mobilen Knotens innerhalb eines AGbMM

Die Bewegung eines mobilen Knotens entsprechend dem Modell aus Abbildung 7 ist in Abbildung 8 dargestellt. Der Knoten beginnt im linken Cluster und läuft darin zu zwei weiteren Positionen. Der Knoten erreicht die Positionen nach 17 und 26 Sekunden. Während der Knoten an der dritten Position wartet, endet seine Aufenthaltsdauer in diesem Cluster und er bewegt sich zur Kante und erreicht diese nach 36 Sekunden Simulationszeit. Nach insgesamt 46 Sekunden betritt der Knoten das rechte Cluster und erhält für dieses eine Aufenthaltsdauer von ca. 40 Sekunden. Während dieser Zeit geht der Knoten zu vier weiteren Positionen. Auf dem Weg zur fünften Position endet die Aufenthaltsdauer während der Bewegung und der Knoten verlässt das Cluster. Nach 89 Sekunden betritt der Knoten die Kante und begibt sich auf den Weg zum linken Cluster, dieses erreicht er nach insgesamt 99 Sekunden.

Innerhalb eines Clusters wird die Bewegung durch ein *Random Waypoint Mobility Model* beschrieben. Demnach muss, analog zum *Random Waypoint Mobility Model*, die Knotendichte im Zentrum eines Clusters höher sein als an den Randbereichen. Zusätzlich verlassen die mobilen Knoten in regelmäßigen Abständen die Cluster und gehen über inzidente Kanten in eine anderes Cluster. Das hat zur Folge, dass in einem Cluster die Bereiche nahe der inzidenten Kanten eine höhere Dichte aufweisen als andere Gebiete innerhalb des Clusters. Abbildung 9 zeigt die mittlere Verteilung von 100 mobilen Knoten innerhalb des zuvor beschriebenen Modells. Hierbei kennzeichnen dunkle Bereiche, Gebiete mit einer hohen Dichte und umgekehrt. Die Verteilungen der Knoten, mit einer höheren Dichte im Bereich der Ausgänge und im Zentrum der Cluster, entspricht der Wirklichkeit. Denn auch in der Realität, ist die Konzentration in der Mitte eines Raumes und an den Ausgänge oftmals höher als in anderen Bereichen.

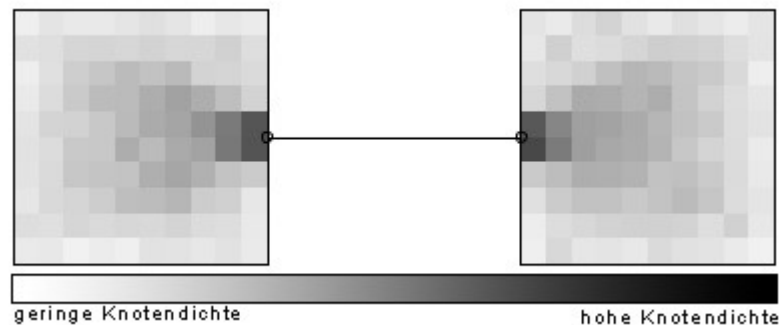


Abbildung 9: Verteilung der Knoten innerhalb eine AGbMM

Um die Bewegung mobiler Knoten entsprechend dem *AGbMM* simulieren zu können, müssen die Knoten zu Beginn der Simulation eine Position innerhalb des Modells haben. Das heißt, jedem Knoten muss vor Beginn der Simulation eine Startposition zugewiesen werden. Im Rahmen dieser Arbeit werden die mobilen Knoten initial innerhalb der Cluster platziert. Dabei werden die Knoten entsprechend der Aufenthaltsdauern auf die Cluster verteilt. In den Clustern werden die Knoten anfänglich gleichverteilt positioniert. Es sind demnach zu Beginn der Simulation eines *AGbMM* keine mobilen Knoten auf den Kanten des Modells. Infolgedessen ist ein gewisser Vorlauf der Simulation erforderlich, bevor die mobilen Knoten entsprechend des *AGbMM* verteilt sind. Das heißt, bevor realistische Messungen oder Simulationen mit dem *AGbMM* durchgeführt werden können, muss zunächst eine bestimmte Zeit simuliert werden.

### 3.2 Erweiterungen

Das *AGbMM* ist kein Bewegungsmodell im herkömmlichen Sinn, vielmehr ist es ein Meta-Modell. Das heißt, das *AGbMM* nutzt vorhandene Bewegungsmodelle zur Beschreibung komplexer Bewegungen. Im Basismodell wurde innerhalb der Cluster das *Random Waypoint Mobility Model* zur Beschreibung der Bewegung verwendet. Neben dem *Random Waypoint Mobility Model*, als internes Bewegungsmodelle der Cluster, kann theoretisch jedes beliebige Bewegungsmodell verwendet werden. Es wäre demnach sogar möglich das *AGbMM* rekursiv zu verwenden und somit beliebig verschachtelte Modelle zu konstruieren. Im Verlauf der Implementierung des *AGbMM* sind verschiedene Erweiterungen zum Basismodell des *AGbMM* entstanden. Durch diese Ergänzungen lassen sich komplexere Modelle definieren und dementsprechend mehr Szenarien mit dem *AGbMM* beschreiben. Zusätzlich können entworfene Modelle wiederverwendet, Modelle hierarchisch anordnet und auf verschiedenen Abstraktionsebenen betrachtet werden.

Das erweiterte *AGbMM* basiert weiterhin auf einem gerichteten Graphen  $G$ , dieser wird aber im Gegensatz zum Basismodell anders beschrieben. Der Graph  $G$  besteht aus einer Menge von Knoten  $V$  (Clustern) und Kanten  $E$ . Jeder Knoten  $v \in V$  hat eine gewisse Anzahl von

## Beschreibung des Area Graph-based Mobility Models

---

Verbindungsendpunkten und eine Kante verläuft jeweils zwischen zwei beliebigen Endpunkten der Cluster. Innerhalb der Knoten des Graphen wird die Bewegung eines mobilen Knotens durch ein beliebiges Bewegungsmodell beschrieben (im Gegensatz zum Basismodell, hier wurde innerhalb der Cluster immer das *Random Waypoint Mobility Model* verwendet). Dieses kann analog zum Basismodell ein *Random Waypoint Mobility Model*, aber auch rekursiv ein *AGbMM* sein.

Wenn ein mobiler Knoten ein Cluster betritt, wird seine Bewegung innerhalb dieses Knotens für eine gewisse Zeit durch das interne Bewegungsmodell bestimmt. Handelt es sich bei dem internen Modell wieder um ein *AGbMM*, so betritt der mobile Knoten rekursiv in diesem Modell für eine gewisse Zeit ein Cluster. Das *Random Waypoint Mobility Model* dient hierbei als Rekursionsanker für die Verschachtelung mehrerer Bewegungsmodelle ineinander. Unabhängig davon, in wie vielen rekursiven Modellen sich ein mobiler Knoten befindet, sobald die Aufenthaltsdauer in einem hierarchisch höheren Cluster abläuft, übernimmt dieses Cluster die Kontrolle über den mobilen Knoten. Der Knoten wird dann zu einem zufälligen Ausgang (Verbindungsendpunkt) des Clusters bewegt. Weil das interne Modell nicht konvex sein muss, kann nicht, wie beim *Random Waypoint Mobility Model*, immer der direkte Weg zum Ausgang gewählt werden. Ist das interne Bewegungsmodell ein *AGbMM*, so verläuft der kürzeste Weg zum Ausgang gegebenenfalls über mehrere Cluster und Kanten des Graphen. Dieser Weg kann wiederum die Berechnung des kürzesten Weges in weiteren Modell erfordern. Das heißt, jedes Bewegungsmodell muss einen Algorithmus zur Berechnung des kürzesten Weges zwischen zwei Punkten innerhalb des Modells zur Verfügung stellen. Beim *Random Waypoint Mobility Model* ist das die direkte Strecke, die zwei Punkte miteinander verbindet. Beim *AGbMM* kann zur Berechnung des kürzesten Weges der Dijkstra Algorithmus verwendet werden.

Um den Unterschied des erweiterten *AGbMM* zum Basismodell deutlich zu machen, wird das Modell eines Fachbereiches einer Universität betrachtet. Der Fachbereich besteht aus einem Institutgebäude, einer Bibliothek und einem Weg der diese Gebäude in beide Richtungen miteinander verbindet. Das Institut hat eine Aufenthaltsdauer von 10 Minuten bis zu 4 Stunden. Dieses Szenario lässt sich sowohl mit dem Basismodell als auch mit rekursiven Bewegungsmodellen konstruieren. Wird jetzt das Institutgebäude genauer betrachtet, z.B. mit einem Hörsaal und mehreren Seminarräumen, so lässt sich das auf zwei Arten umsetzen. Für beide Herangehensweisen müssen die einzelnen Räume mit den entsprechenden Zeiten in das vorhandene Modell integriert werden. Beim Basismodell muss das Cluster mit dem Institutgebäude durch mehrere einzelne Cluster ersetzt werden. Diese Modelle repräsentieren dann die einzelnen Räume des Instituts. Im Gegensatz dazu, wird im erweiterten Modell vom Cluster des Institutgebäudes nicht mehr auf ein einzelnes *Random Waypoint Mobility Model* verwiesen, sondern auf ein *AGbMM*. Dieses *AGbMM* stellt die einzelnen Räume des Instituts dar. Mit letzterem Ansatz läuft die Bewegung eines mobilen Knotens wie folgt ab: Der Knoten betritt das Institutgebäude und kann sich in diesem entsprechend der Vorgaben des *AGbMM* frei bewegen. Sobald die Aufenthaltsdauer des

## Beschreibung des Area Graph-based Mobility Models

---

Knotens innerhalb des Instituts abläuft, verlässt er das Institutgebäude in Richtung Bibliothek. Demgegenüber gibt es im Basismodell keine Aufenthaltsdauer für das gesamte Institutgebäude. Ein mobiler Knoten kann sich also theoretisch unbegrenzt im Institutgebäude bewegen oder es bereits nach Ablauf der Aufenthaltsdauer im ersten Raum wieder verlassen.

Ein weiterer Vorteil verschachtelter Bewegungsmodelle ist die Wiederverwendbarkeit dieser. So lässt sich z.B. ein entworfenes Modell in jedem Cluster eines *AGbMM* erneut verwenden, es müssen hierzu lediglich die Verbindungsendpunkte für das übergeordnete Modell festgelegt werden. Durch diese Punkte ist es möglich, vom übergeordneten Modell in das interne Modell zu gelangen und umgekehrt. So kann eine Universität zuerst auf oberster Ebene mit Fachbereichen und deren Verbindungen modelliert werden. Wird dann zu einem späteren Zeitpunkt ein detaillierteres Modell benötigt, können Gebäude und Wege für die einzelnen Fachbereiche entworfen werden. Ist das weiterhin zu ungenau, können auf einer tieferen Ebene die Räume der einzelnen Gebäude beschrieben werden. In gleicher Weise ist es nun möglich, das gesamte Modell der Universität in ein komplexeres Szenario zu integrieren.

Durch die hierarchische Anordnung der Modelle entstehen verschiedene Betrachtungsebenen. So lässt sich ein komplexes Modell einer Universität (mit modellierten Gebäuden für jeden Fachbereich) zuerst auf der Ebene der einzelnen Fachbereiche und deren Verbindungen betrachten, um einen Überblick über die Bewegung auf dem gesamten Universitätsgelände zu erhalten, ohne die Bewegung in den einzelnen Gebäuden zu berücksichtigen. Auf der nächsten Ebene ließen sich, wenn notwendig die detaillierten Bewegungen der Studenten innerhalb der einzelnen Fachbereiche betrachten.

Eine weitere Änderung am Basismodell ist die Gewichtung der Kanten des Graphen. Im Basismodell ist das Gewicht einer Kante eine Zahl zwischen 0 und 1, welche die Wahrscheinlichkeit repräsentiert, dass diese Kante beim Verlassen des dazugehörigen Clusters genutzt wird. Das bedeutet, im Basismodell muss die Summe der Gewichte aller ausgehenden Kanten eines Clusters immer 1 sein. Das führt dazu, dass beim Einfügen einer weiteren Kante mit einem Gewicht ungleich 0, auch das Gewicht aller anderen Kanten des Clusters angepasst werden muss. Um diese Anpassung zu vermeiden, ist das Gewicht einer Kante im erweiterten Modell eine positive reelle Zahl inklusive 0. Die Wahrscheinlichkeit, dass diese Kante beim Verlassen des Clusters genutzt wird, entspricht nun dem Verhältnis des Gewichtes dieser Kante zur Summe der Gewichte aller ausgehenden Kanten des Clusters. Das heißt, sollen alle Kanten eines Clusters mit gleicher Häufigkeit benutzt werden, muss das Gewicht aller Kanten gleich sein. Analog folgt aus einem doppelt so hohem Gewicht eine doppelt so hohe Nutzung dieser Kante beim Verlassen des Clusters.

## Beschreibung des Area Graph-based Mobility Models

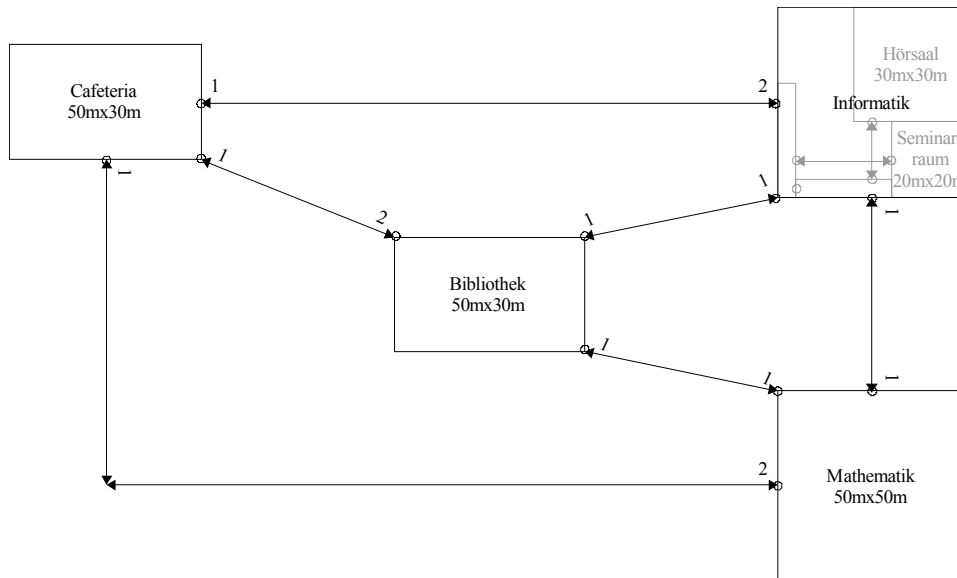


Abbildung 10: Erweiterungen des Area Graph-based Mobility Model

Abbildung 10 zeigt ein erweitertes *AGbMM* für den Fachbereich einer Universität. Der Graph besteht auf der obersten Ebene aus vier Knoten: der Cafeteria, dem Informatikgebäude, der Bibliothek und dem Mathematikgebäude. Alle Gebäude sind direkt miteinander verbunden, d.h. jedes Gebäude ist direkt von jedem anderen aus zu erreichen. Demzufolge hat der Graph insgesamt 12 Kanten. Die Wahrscheinlichkeit, dass ein Knoten die Bibliothek in Richtung Cafeteria verlässt ist 50% und somit doppelt so hoch wie die Wahrscheinlichkeit, in Richtung Informatik oder Mathematik zu gehen, sie beträgt jeweils 25%. Analog verhält es sich beim Informatik- und Mathematikgebäude, hier ist die Gewichtung der Kanten zur Cafeteria ebenfalls doppelt so hoch wie zu den anderen zwei Gebäuden. Von der Cafeteria hingegen, sind alle Gebäude mit gleicher Wahrscheinlichkeit (jeweils ein Drittel) zu erreichen. Die Cafeteria, die Bibliothek und das Mathematikgebäude werden jedes Mal durch ein *Random Waypoint Mobility Models* beschrieben. Wohingegen das Informatikgebäude durch ein weiteres *AGbMM* definiert wird. Dieses besteht aus einem Hörsaal, einem Seminarraum und zwei Fluren. Die Räume sind alle direkt miteinander verbunden und haben eine Gewichtung von 1. Sie wurden um die Übersichtlichkeit zu wahren nicht eingezeichnet.

## 4 Implementierung

Im Rahmen dieser Arbeit werden verschiedene Simulationen mit dem *Area Graph-based Mobility Model (AGbMM)* durchgeführt. Die Berechnungen der mittleren Knotenanzahl im Kapitel 5 sollen verifiziert und Bewegungsdateien zur Simulation der Nachrichtenverteilung in Kapitel 6 erstellt werden. Es muss eine Syntax entworfen werden, um ein *AGbMM* beschreiben zu können. Zusätzlich wird ein Programm benötigt, um entworfenen Modelle einlesen und die Bewegung der Knoten simulieren zu können. Mit diesem Simulator kann dann die durchschnittliche Knotenanzahl ermittelt und die Ergebnisse der Analyse aus Kapitel 5 verifiziert werden. Des Weiteren wird ein Programm zur Simulation verschiedener Verteilungsprotokolle benötigt, um zum einen die Vorhersagen aus Kapitel 6 zu verifizieren und zum anderen Szenarien mit Pull Protokollen in Kapitel 7 simulieren zu können. Ein Programm, welches diese Aufgaben erfüllt und diesbezüglich zur Simulation von mobilen Ad-hoc Netzen verwendet werden kann, ist der Netzwerk Simulator 2 (ns2, <http://www.isi.edu/nsnam/ns/>). In ns2 ist standardmäßig nur das *Random Waypoint Mobility Model* zur Bewegung der mobilen Knoten integriert. Um andere Bewegungsmodelle in ns2 integrieren zu können, besteht die Möglichkeit Bewegungsdateien in ns2 einzulesen. In einer solchen Datei wird die Bewegung einer bestimmten Anzahl an Knoten für einen festgelegten Zeitraum beschrieben. Demzufolge wird ein Programm benötigt, welches modellierte Szenarien einlesen und simulieren kann und die Bewegung der Knoten in einer für ns2 kompatiblen Bewegungsdatei ausgibt. Im Folgenden werden die drei umzusetzenden Aufgaben und deren Implementierungen genauer beschrieben.

### 4.1 Sprache zur Beschreibung des Area Graph-based Mobility Models

In Anbetracht der steigenden Popularität und der einfachen und lesbaren Struktur wurde XML (eXtensible Markup Language) als Sprache zur Beschreibung eines *AGbMM* gewählt. Es existieren viele Werkzeuge für verschiedene Programmiersprachen zum Einlesen und Verarbeiten von XML-Dateien. Darüber hinaus besteht durch XML die Möglichkeit diese Form der Beschreibung als Zwischendarstellung eines *AGbMM* zu verwenden. So können Programme entwickelt werden um *AGbMM* mit Hilfe grafischer Benutzeroberflächen zu modellieren und diese Modelle in eine XML-Datei zu exportieren. Diese Datei kann anschließend von anderen Anwendungen eingelesen und weiterverarbeitet werden.

Folgende Tags ermöglichen die Beschreibung eines *AGbMM* innerhalb einer XML-Datei. Bei der Angabe von Entfernungen, Zeiten oder Geschwindigkeiten können dabei eine Vielzahl von gängigen Maßeinheiten verwendet werden.



## Implementierung

<i>Tag</i>	<i>Beschreibung</i>	<i>Attribute</i>
<b>mobilitymodel</b>	<p>Definiert ein Bewegungsmodell. Je nach Art des Modells mit verschiedenen vielen Attributen oder Unterelementen.</p> <p>Ein <i>Random Waypoint Mobility Model (RWMM)</i> kann keine Unterelemente enthalten.</p> <p>Ein <i>AGbMM</i> kann Knoten (node) oder Kanten (connection) als Unterelemente enthalten.</p>	<p><b>name</b> = Name des Modells</p> <p><b>type</b> = Art des Modells, z.B. <i>AGbMM</i> oder <i>RWMM</i></p> <p><b>width, height</b> = Breite und Höhe des Modells (nur <i>RWMM</i>)</p> <p><b>minSpeed, maxSpeed</b> = minimale bzw. maximale Geschwindigkeit im Modell (nur <i>RWMM</i>)</p> <p><b>minPauseTime, maxPauseTime</b> = minimale bzw. maximale Wartezeit zwischen den Bewegungen (nur <i>RWMM</i>)</p>
<b>node</b>	<p>Definiert einen Knoten innerhalb eines <i>AGbMM</i> (mobilitymodel). Dieser Knoten muss genau ein Bewegungsmodell (mobilitymodel) enthalten. Zusätzlich kann er mehrere Endpunkte (endpoint) enthalten.</p>	<p><b>x, y</b> = X-Position bzw. Y-Position jeweils relativ zum übergeordneten Modell</p> <p><b>minTime, maxTime</b> = minimale bzw. maximale Aufenthaltsdauer im Knoten und somit im internen Modell</p>
<b>endpoint</b>	<p>Definiert einen Verbindungsendpunkt innerhalb eines Knotens (node).</p>	<p><b>name</b> = Name des Endpunktes (wird für die Kanten verwendet)</p> <p><b>x, y</b> = X-Position bzw. Y-Position jeweils relativ zum übergeordneten Knoten</p>
<b>connection</b>	<p>Definiert eine Kante innerhalb eines <i>AGbMM</i> (mobilitymodel).</p>	<p><b>from, to</b> = Start- bzw. Endpunkt der Kante. Angabe des Namen eines definierten Endpunktes (endpoint)</p> <p><b>minSpeed, maxSpeed</b> = minimale bzw. maximale Geschwindigkeit auf der Kante</p>

Ein Beispiel für die Beschreibung eines *AGbMM* mittels XML ist in Abbildung 11 dargestellt. Es ist die XML-Beschreibung des in Abbildung 7 beschriebenen *AGbMM*.

```

1<mobilitymodel type="Area Graph-based Mobility Model">
2  <node x="-10m" y="0m" minTime="0s" maxTime="60s">
3    <mobilitymodel
4      name="left"
5      type="Random Waypoint Mobility Model"
6      width="10m"
7      height="10m"
8      minSpeed="1m/s"
9      maxSpeed="1m/s"
10     minPauseTime="0s"
11     maxPauseTime="10s"/>
12    <endpoint name="endpoint1" x="5m" y="0m" />
13  </node>
14  <node x="10m" y="0m" minTime="0s" maxTime="60s">
15    <mobilitymodel
16      name="right"
17      type="Random Waypoint Mobility Model"
18      width="10m"
19      height="10m"
20      minSpeed="1m/s"
21      maxSpeed="1m/s"
22      minPauseTime="0s"
23      maxPauseTime="10s"/>
24    <endpoint name="endpoint2" x="-5m" y="0m" />
25  </node>
26  <connection from="endpoint1" to="endpoint2" minSpeed="1m/s" maxSpeed="1m/s" />
27  <connection from="endpoint2" to="endpoint1" minSpeed="1m/s" maxSpeed="1m/s" />
28</mobilitymodel>

```

Abbildung 11: Beschreibung eines Area Graph-based Mobility Models mit XML

Das umschließende Tag *mobilitymodel* (Zeile 1-28) stellt das übergeordnete *AGbMM* dar (zu erkennen am Attribut *type*) und enthält die entsprechenden Unterelemente, die das Modell beschreiben. In Zeile 2 wird dann der linke Knoten des Graphen definiert, dargestellt durch das Tag *node*. Es erhält als Attribute die Position des Clusters im übergeordneten Modell und die minimale und maximale Aufenthaltsdauer eines mobilen Knotens innerhalb des internen Clusters. Dieses wird durch ein Bewegungsmodell in Zeile 3 und einen Verbindungsendpunkt in Zeile 12 beschrieben. Bei dem internen Bewegungsmodell des Clusters handelt es sich um ein *Random Waypoint Mobility Model*, welches durch das Attribut *type* in Zeile 5 definiert wird. Außerdem werden noch Attribute wie Größe, Geschwindigkeit und Wartezeit für das interne Modell in den Zeilen 6-11 festgelegt. Der Verbindungsendpunkt des linken Clusters wird durch das Tag *endpoint* in Zeile 12 definiert. Er erhält einen Namen, damit er später bei den Kanten des Graphen wiederverwendet werden kann. Des Weiteren wird die Position des Endpunktes relativ zum internen Bewegungsmodell angegeben. Analog ist in den Zeilen 13-25 der rechte Knoten des Graphen definiert. Zuletzt werden durch die Tags *connection* noch die zwei Kanten des *AGbMM* beschrieben. Die Attribute geben an, zwischen welchen zwei Verbindungsendpunkten diese Kante verläuft und wie schnell sich die mobilen Knoten auf der Kante bewegen können.

## 4.2 Area Graph-based Mobility Model Simulator

Es wird ein Programm benötigt, um die Bewegung mobiler Knoten entsprechend dem *AGbMM* zu simulieren. Mit diesem Programm soll sowohl die Verteilung der mobilen Knoten innerhalb des *AGbMM* ermittelt, als auch die Bewegungen der Knoten in eine Datei

## Implementierung

---

exportiert werden, um diese in ns2 simulieren zu können. Als Programmiersprache wurde Java gewählt, weil diese frei erhältlich ist und plattformübergreifend eingesetzt werden kann. Zum Einlesen der Bewegungsmodelle im XML-Format wird der „*Xerces Java Parser*“ benutzt. Darüber hinaus wurde die *Eclipse* Plattform in der Version 2.1 als Entwicklungsumgebung verwendet.

Die zentralen Objekte der Simulation sind die mobilen Knoten. Jeder Knoten kennt das aktuelle Bewegungsmodell in dem er sich momentan befindet und seinen aktuellen Status. Der Status repräsentiert dabei die Aktion, die ein Knoten gegenwärtig ausführt. Die verschiedenen Status sind: Gehe zu einer Position mit einer zufälligen Geschwindigkeit, warte an einer Position für ein zufällige Zeit oder verlasse ein übergeordnetes Modell und gehe zu einem anderen Cluster. Letzterer Status kann sich hierbei aus mehreren Bewegungen zusammensetzen. Zusätzlich weiß jeder Knoten, wann seine aktuelle Aktion endet und somit die nächste Aktion beginnt. Wenn sich ein mobiler Knoten zu einer Position bewegt, so ändert sich seine Aktion genau in dem Moment, wenn der Knoten das Ziel erreicht. Demnach wird die Bewegung eines mobilen Knotens durch diskrete Ereignisse beschrieben. Das heißt, jeder Knoten hat jederzeit einem konkreten Status. Ausgehend von diesem Status, wird zu einem festgelegten Zeitpunkt unter Verwendung definierter Zufallswerte der neue Status berechnet. Zur Simulation der Bewegung eines einzelnen mobilen Knotens können die Zustandsänderungen nacheinander in chronologischer Reihenfolge durchgeführt werden. Bei der Simulation mehrerer Knoten ist zu beachten, dass die Zustandsänderungen in der richtigen Abfolge ausgeführt werden. Das bedeutet, es muss zu jedem Zeitpunkt der Simulation bekannt sein, welcher mobile Knoten, entsprechend der aktuellen Simulationszeit, als nächstes seine Aktion ändert. Es wird demnach eine Datenstruktur benötigt, die eine Menge von Ereignissen verwaltet und aus dieser Menge das Ereignis mit der kleinsten Zeit extrahieren bzw. ein neues Ereignisse hinzufügen kann. Als Datenstruktur wurde hierzu ein *Binomial Heap* verwendet, da dieser eine deutlich bessere Performance aufweist als ein gewöhnlicher Heap.

Der Ablauf der Simulation gestaltet sich wie folgt: Zu Beginn werden einmalig alle Knoten mit dem Zeitpunkt ihrer ersten Aktionsänderung in den *Binomial Heap* eingefügt. Anschließend wird das erste Ereignis des Heaps extrahiert. Dieses Ereignis verweist auf den mobilen Knoten dessen Aktion zu diesem Zeitpunkt abgelaufen ist. Nachdem die nächste Aktion des mobilen Knotens ermittelt ist, wird der Knoten, entsprechend des Zeitpunktes der nächsten Aktionsänderung, in den Heap eingefügt. Im Anschluss extrahiert man erneut das erste Element des Heaps und der Vorgang beginnt von vorn.

Durch die diskreten Ereignisse werden die Positionen der mobilen Knoten nicht fortlaufend aktualisiert. Das heißt, während der Simulation wird die Position der Knoten nur bei Aktionsänderungen neu ermittelt. Um trotzdem zu jedem beliebigen Zeitpunkt der Simulation die aktuelle Verteilung der Knoten visualisieren zu können, kann die Position eines Knotens jederzeit neu berechnet werden. Dazu wird anhand der momentanen

## Implementierung

Simulationszeit und der aktuellen Aktion eine approximative Position für den Knoten bestimmt. Vor jedem neuen Bild des Simulators werden die Positionen aller Knoten neu berechnet.

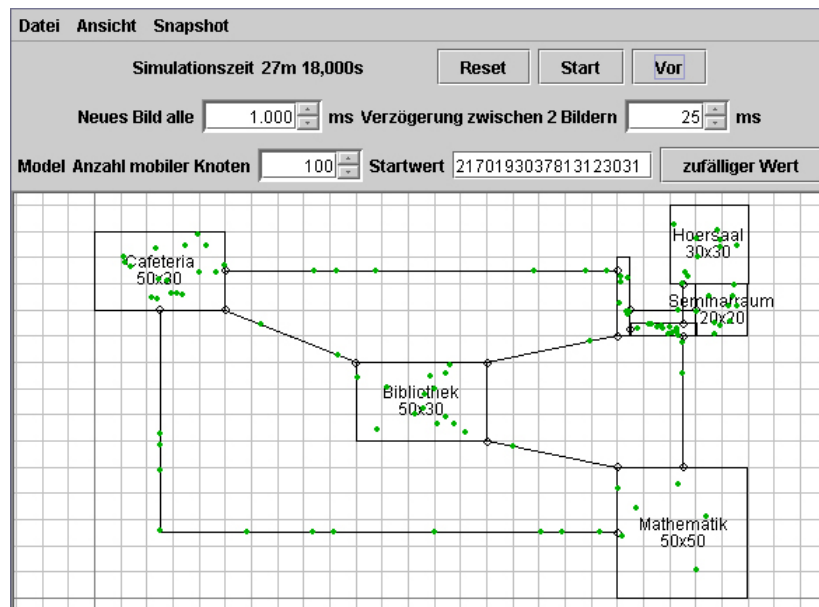


Abbildung 12: Java-Programm zur Simulation des Area Graph-based Mobility Models

Die grafische Oberfläche des Simulator ist in Abbildung 12 dargestellt. Mit ihr lassen sich Bewegungsmodelle in Form einer XML-Datei einlesen und darstellen. Es lässt sich die Anzahl der mobilen Knoten einstellen und der Startwert des Zufallsgenerators festlegen. Letzteres ist sinnvoll, um identische Simulationen zu einem späteren Zeitpunkt erneut durchführen zu können. Denn bei identischen Bewegungsmodellen, gleicher Knotenanzahl und gleichem Startwert bleibt die Bewegung der Knoten unverändert. Zusätzlich können am Hauptfenster verschiedene Parameter der Simulation eingestellt werden.

Neben dem Hauptfenster mit der grafischen Darstellung des Modells, gibt es weitere Fenster mit anderen Informationen und Einstellungen, wie z.B. der Anzahl mobiler Knoten, Informationen zu den einzelnen Knoten, Informationen zum Bewegungsmodell und ein Fenster mit verschiedenen grafischen Einstellungen. Das Fenster mit der Anzahl der mobilen Knoten stellt in den Zeilen einer Tabelle die einzelnen Elemente des *AGbMM* dar. Die Spalten zeigen unter anderem die absolute und durchschnittliche Anzahl mobiler Knoten für das jeweilige Element an. Dieses Fenster dient zur Überprüfung der analytischen Ergebnisse aus Kapitel 5. Im Informationsfenster zu den mobilen Knoten lässt sich neben der derzeitigen Position auch die aktuell ausgeführte Aktion jedes Knotens überwachen.

Im Fenster mit den grafischen Einstellungen können verschiedene Attribute zur Darstellung des Modells geändert werden. Man kann die Farbe des Modells, der Knoten, des

Hintergrunds und der Schrift anpassen oder die Schriftart und Schriftgröße der Beschriftungen ändern. Zusätzlich kann man auswählen, welche Eigenschaften der Elemente dargestellt werden. Mit Hilfe einer Schnappschuss Funktion lässt sich die gewünschte Darstellung eines Modells anschließend exportieren. Neben den eigentlichen mobilen Knoten können auch deren Wege oder die mittlere Verteilung der Knoten innerhalb eines Cluster visualisiert werden.

In Abbildung 13 ist ein aufgenommener Schnappschuss des *AGbMM* Simulator dargestellt. Dieser zeigt, neben dem Modell und den mobilen Knoten, zusätzlich die mittlere Knotendichte für verschiedene Bereiche innerhalb eines Clusters. Außerdem sind die einzelnen Elemente mit verschiedenen Eigenschaften und der aktuellen Knotenanzahl beschriftet.

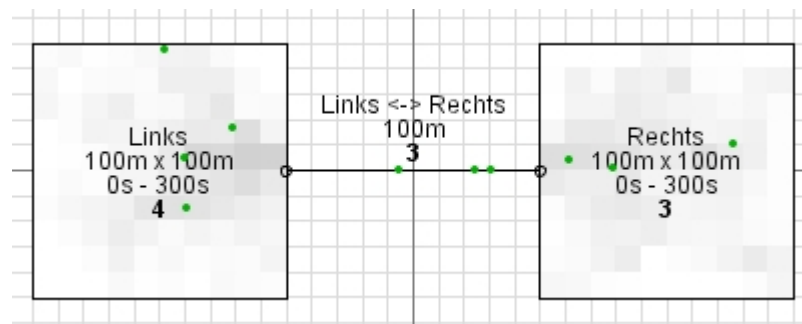


Abbildung 13: Schnappschuss eines Modells mit dem *AGbMM* Simulator

### 4.3 Generierung der Bewegungsdateien für den Netzwerksimulator 2

Um die Auswirkung des *AGbMM* auf die Effizienz von Verteilungsprotokollen zu testen, sollen verschiedene heterogene Szenarien modelliert und in ns2 simuliert werden. Da die Bewegung der mobilen Knoten entsprechend dem *AGbMM* bereits vollständig durch den Simulator umgesetzt ist, müssen diese Daten nur in einem ns2 kompatiblen Format ausgegeben werden.

In ns2 wird die Bewegung eines mobilen Knotens folgendermaßen beschrieben: Jeder Knoten hat eine definierte Startposition. Anschließend, zu bestimmten Zeitpunkten der Simulation, bewegt sich ein Knoten direkt (d.h. auf einer Geraden) zu einer neuen Position, das geschieht mit einer vorgeschriebenen Geschwindigkeit. Nach Erreichen des Ziels wartet der Knoten an dieser Position bis seine nächste Bewegung beginnt. Das heißt, ns2 benötigt in einer Bewegungsdatei die Startpositionen von jedem Knoten und die gesamten Bewegungsänderungen aller Knoten innerhalb des zu simulierenden Zeitraums. Hierfür wurden im Simulator verschiedene Schnittstellen definiert, um verschiedene Ereignisse der mobilen Knoten überwachen und ausgeben zu können. Zum Beispiel kann jede

## Implementierung

---

Bewegungsänderung eines Knotens überwacht werden. Eine beliebige Java-Klasse kann diese Schnittstelle implementieren und sich anschließend als Beobachter beim mobilen Knoten registrieren. Sobald der Knoten die Bewegung zu einer neuen Position beginnt, bekommt der Beobachter dies mitgeteilt. Ihm wird dazu der Zeitpunkt des Ereignisses und der Knoten, der das Ereignis ausgelöst hat, übermittelt. Mittels des mobilen Knotens kann nun direkt die Zielposition der Bewegung und die dabei verwendete Geschwindigkeit ermittelt werden. Diese Daten kann der Beobachter in jeglicher Form weiterverarbeiten, z.B. die Bewegung in eine Datei exportieren oder in einem anderen Programm visualisieren. Im Rahmen dieser Arbeit wurde der Export der Bewegung in ns2 Bewegungsdateien implementiert.

Zusätzlich sind weitere Schnittstellen im *AGbMM* Simulator definiert, weil in folgenden oder weiterführenden Arbeiten mit dem *AGbMM* gegebenenfalls andere Netzwerk-Simulatoren oder anderen Programme verwendet werden. Durch diese zusätzlichen Schnittstellen können mehr Eigenschaften des Modells und der mobilen Knoten überwacht werden. Diese Daten lassen sich ebenfalls exportieren und können in anderen Programmen weiterverarbeitet oder analysiert werden. So kann neben der Änderung der Bewegung eines Knotens auch seine generelle Bewegung überwacht werden. Das heißt, wann betritt oder verlässt dieser Knoten ein Cluster und wann beginnt er zu laufen oder zu warten. Neben den mobilen Knoten können auch Elemente des Bewegungsmodells überwacht werden, zum Beispiel eine Kante des Graphen und wann diese von einem Knoten betreten oder verlassen wird.

## 5 Verteilung der mobilen Knoten

In diesem Kapitel wird die Verteilung der mobilen Knoten in einem *Area Graph-based Mobility Models (AGbMM)* analysiert. Das bedeutet, es wird untersucht wie sich die mobilen Knoten auf die einzelnen Elemente des *AGbMM* verteilen. Ziel dieses Kapitels ist: bei Vorgabe eines *AGbMM* mit dem Graphen  $G=(V,E)$  und einer Gesamtzahl mobiler Knoten  $N$ , für jedes Element  $a \in V \cup E$ , die durchschnittliche Anzahl von mobilen Knoten  $n(a)$  in  $a$  berechnen zu können.

Bei  $n(a)$  handelt es sich um einen Erwartungswert. Auf diesen Wert stabilisiert sich im Laufe der Simulation die durchschnittliche Knotenanzahl eines Elementes  $a$  und erreicht den exakten Wert  $n(a)$  theoretisch nach unendlicher Simulationszeit. Der Durchschnittswert  $n(a)$  ermöglicht also keine Vorhersagen über die konkrete Verteilung der mobilen Knoten zu einem bestimmten Zeitpunkt der Simulation. Umgekehrt lässt sich auch aus der momentanen Verteilung der Knoten, zu einem bestimmten Zeitpunkt der Simulation, kein Rückschluss auf die durchschnittliche Anzahl mobiler Knoten  $n(a)$  ziehen. Betrachtet man zum Beispiel ein Modell, bei dem sich die mobilen Knoten periodisch durch die einzelnen Cluster bewegen, so können sich zu einem bestimmten Zeitpunkt der Simulation alle mobilen Knoten in einem Cluster des Graphen befinden. Entsprechend der durchschnittlichen Verteilung befinden sich aber im Mittel in allen Clustern gleich viele mobile Knoten.

Um die Verteilung der Knoten innerhalb eines *AGbMM* zu berechnen, muss zuerst festgestellt werden, welche Attribute des Modells Auswirkung auf die Verteilung haben. Diese Merkmale können dann genauer untersucht und in Formeln übertragen werden. Erste Simulationen und Überlegungen zeigten aber, dass praktisch alle Merkmale wie Aufenthaltsdauer, Wartezeit, Geschwindigkeit, Gewichtung der Kanten und auch die Topologie des Graphen Auswirkungen auf die Verteilung haben. All diese Eigenschaften des *AGbMM* lassen sich, entsprechend ihrer Auswirkungen bei der Verteilung, rudimentär in zwei Gruppen einteilen. Die erste Gruppe sind die Eigenschaften die Auswirkung auf die Verweildauer eines mobilen Knotens in einem Element haben, dazu gehören die Aufenthaltsdauer, die Wartezeit, die Geschwindigkeit und die Gewichtung der Kanten. Die zweite Gruppe sind die Merkmale, die Auswirkung auf den Grad der Verbindung zwischen den einzelnen Elementen haben. Zu diesen Merkmalen gehört die Topologie des Graphen und die Gewichtung der Kanten. Das heißt, das einzige Attribut des *AGbMM* welches Auswirkungen auf beide Gruppen hat ist die Gewichtung der Kanten.

In den folgenden Abschnitten werden beide Gruppen mit den jeweiligen Merkmalen genauer untersucht. Dazu werden verschiedene exemplarische Szenarien entworfen, um die Auswirkungen der entsprechenden Eigenschaften genauer untersuchen und analysieren zu können. Diese Ergebnisse dienen anschließend als Grundlage für die Berechnungen der

mittleren Knotenanzahl. Dabei werden Formeln aus den einzelnen Attributen des *AGbMM* abgeleitet und durch Anwendung eines Bottom-Up-Verfahrens zu einer geschlossenen Formel zusammengefügt. Im letzten Abschnitt dieses Kapitels wird dann die Anzahl mobiler Knoten für einige ausgewählte Szenarien berechnet und mit entsprechenden Simulationsergebnissen verglichen.

### System Modell

Für die Analyse der Knotenverteilung wird das Basismodell des *AGbMM* ohne Erweiterungen aus Abschnitt 3.1 verwendet. Das heißt, es werden keine rekursiven Aufrufe mehrerer *AGbMM* untersucht. Des Weiteren sind alle Berechnungen in diesem Kapitel nur für Modelle anwendbar bei denen der Graph des *AGbMM* zusammenhängend ist, d.h. alle Cluster und Kanten von jedem Element des Modells erreichbar sind. Zur Vereinfachung bezeichne für den Rest des Kapitels  $G=(V,E)$  den Graphen eines *AGbMM*. Weiterhin seien  $v \in V$  ein Knoten und  $e \in E$  eine Kante des Graphen  $G$ . Und  $A=V \cup E$  sei die Menge der Knoten und Kanten des Graphen  $G$  und  $a$  sei ein Element der Menge  $A$ .

## 5.1 Analyse der Verweildauer

Zuerst werden die Eigenschaften untersucht, die Auswirkungen auf die Verweildauer eines Knotens innerhalb eines Elementes  $a$  haben. Dazu sei ein einfaches Modell wie in Abbildung 14 mit den zwei Clustern  $V_1$  und  $V_2$  und den zwei gerichteten Kanten  $(V_1, V_2)$  und  $(V_2, V_1)$  gegeben. Die Cluster seien 10m mal 10m groß und die Kanten jeweils 10m lang. Die Geschwindigkeit der mobilen Knoten sei sowohl in den Clustern als auch auf den Kanten genau 1m/s. Die Pause zwischen den Bewegungen innerhalb der Cluster beträgt 0-10 Sekunden. Weil beide Kanten die einzigen inzidenten Kanten der Cluster sind, haben sie jeweils ein Gewicht von 1.

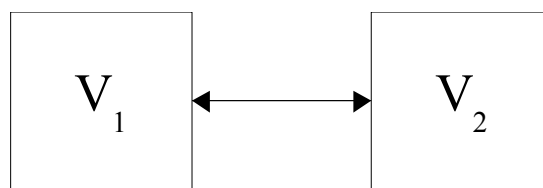


Abbildung 14: Area Graph-based Mobility Model zur Analyse der Aufenthaltsdauer

Tabelle 1 zeigt die Ergebnisse verschiedener Testläufe mit unterschiedlichen Aufenthaltsdauern  $t_{min}$  und  $t_{max}$  der Cluster  $V_1$  und  $V_2$ . Hierbei wurden jeweils die Bewegungen von 100 mobilen Knoten simuliert und dabei eine Stunde lang die mittlere Knotenanzahl der Elemente des *AGbMM* gemessen. Für jede Zeile wurden 20 zufällige Simulationen durchgeführt und daraus der Mittelwert der Knotenanzahl für die einzelnen Elemente des Graphen berechnet. Die mobilen Knoten wurden zu Beginn der Simulation



## Verteilung der mobilen Knoten

entsprechend der Aufenthaltsdauer auf die zwei Cluster aufgeteilt und darin gleichverteilt platziert. Das heißt, zum Zeitpunkt Null befanden sich keine Knoten auf den Kanten des Modells. Weil sich die Knoten, durch die initiale Positionierung, erst nach einer gewissen Simulationszeit im gesamten Modell verteilt haben, wurde mit der Messung der mittleren Knotenanzahl erst nach 200 Sekunden begonnen.

Tabelle 1: Durchschnittliche Anzahl mobiler Knoten für verschiedene Aufenthaltsdauern

$t_{\min}(V_1)-t_{\max}(V_1)$ in Sekunden	$t_{\min}(V_2)-t_{\max}(V_2)$ in Sekunden	$V_1$	$V_2$	$(V_1, V_2)$	$(V_2, V_1)$
0-60	0-60	39,1947	39,1826	10,8311	10,7929
0-120	0-120	43,3288	43,5294	6,5897	6,5562
0-60	0-120	29,5280	54,0985	8,1879	8,1924
0-60	0-180	24,0782	62,7694	6,6483	6,5074
0-120	0-240	31,1915	59,3076	4,7022	4,7959

Die Tabelle bestätigt intuitive Annahmen, so folgt aus gleicher Aufenthaltsdauer eine gleiche durchschnittliche Knotenanzahl, siehe Zeile 1 und 2. Die Unterschiede der mittleren Knotenanzahl von  $V_1$  und  $V_2$  beträgt jeweils weniger als 0,3 was bei einer Gesamtzahl von 100 Knoten einer Abweichung von weniger als 0,3% entspricht. Analog folgt aus einer längeren Aufenthaltsdauer eine höhere durchschnittliche Knotenanzahl, siehe Zeile 3, 4 und 5. Hierbei befindet sich im Cluster mit höherer Aufenthaltsdauer auch immer die größere mittlere Anzahl an Knoten.

Interessant und für spätere Berechnungen von wesentlicher Bedeutung ist die Feststellung, dass sich bei doppelter Aufenthaltsdauer die durchschnittliche Knotenanzahl nicht gleichfalls verdoppelt. Nachdem die Aufenthaltsdauer eines mobilen Knotens innerhalb eines Clusters abgelaufen ist, geht der Knoten zu einem entsprechenden Verbindungsendpunkt um das Cluster zu verlassen. Das heißt, nach Ablauf der Aufenthaltsdauer benötigt ein mobiler Knoten eine gewisse Zeit um zu einer inzidenten Kante zu laufen. Während dieser Zeit befindet sich der Knoten weiterhin innerhalb des Clusters. Demnach ist die gesamte mittlere Verweildauer eines mobilen Knotens in einem Cluster größer als die mittlere Aufenthaltsdauer. Die durchschnittliche Verweildauer  $t$  eines mobilen Knotens innerhalb eines Clusters kann somit wie folgt berechnet werden:  $t=t_s+t_l$ . Hierbei ist  $t_s$  der mittlere Aufenthaltsdauer innerhalb des Clusters und  $t_l$  die durchschnittliche Zeit, die zum Verlassen des Clusters benötigt wird. Wie man in Zeile 3 und 5 von Tabelle 1 erkennt, ist der Einfluss von  $t_l$  auf die durchschnittliche Knotenanzahl

abhängig von dem Verhältnis zwischen  $t_l$  und  $t_s$ . So ist in Zeile 3 bei einer maximalen Aufenthaltsdauer von 60 Sekunden in  $V_1$  und 120 Sekunden in  $V_2$  die durchschnittliche Knotenanzahl in  $V_2$  mit 54,0985 nur 1,83 mal so groß wie in  $V_1$  mit 29,5280. Im Gegensatz dazu ist bei einer maximalen Aufenthaltsdauer von 120 Sekunden in  $V_1$  und 240 Sekunden in  $V_2$  die Knotenanzahl von  $V_2$  bereits 1.90 mal so groß wie die von  $V_1$ , weil hierbei der Einfluss des Weges zum Ausgang weniger stark ist als im Szenario der Zeile 3. Die mittlere Knotenanzahl eines Elementes eines *AGbMM* ist folglich von der mittleren Verweildauer  $t$  eines mobilen Knotens innerhalb des Elementes abhängig. Dabei setzt sich  $t$  aus der mittleren Aufenthaltsdauer  $t_s$  und der durchschnittlichen Zeit zum Verlassen  $t_l$  zusammen. Nachfolgend werden diese Eigenschaften für die Cluster und Kanten eines *AGbMM* genauer untersucht.

### Mittlere Aufenthaltsdauer

In diesem Abschnitt wird analysiert, wie lange sich ein mobiler Knoten in einem Element des *AGbMM* aufhält, bevor er dieses wieder verlässt. Dazu werden die Cluster und Kanten separat betrachtet. Die Kanten eines *AGbMM* haben eine mittlere Aufenthaltsdauer  $t_s$  von 0 Sekunden, weil ein mobiler Knoten eine Kante sofort nach betreten in Richtung des anderen Endpunktes verlässt. Die mittlere Aufenthaltsdauer  $t_s$  eines Clusters ist die Zeit, nach welcher durchschnittlich die Aufenthaltsdauer eines mobilen Knotens abläuft und der Knoten beginnt zum Ausgang des Clusters zu laufen.. Das heißt,  $t_s$  entspricht dem Erwartungswert der Aufenthaltsdauer des entsprechenden Clusters. Im Rahmen dieser Arbeit wird die Aufenthaltsdauer eines mobilen Knotens innerhalb eines Clusters ausschließlich gleichverteilt aus dem Intervall  $[t_{min}, t_{max}]$  bestimmt. Wobei  $t_{min}$  und  $t_{max}$  für jedes Cluster einzeln definiert werden kann. Es ist aber auch möglich, zur Festlegung der Aufenthaltsdauer andere Wahrscheinlichkeitsverteilungen zu verwenden. Entscheidend für die Berechnung der Knotenverteilung innerhalb eines *AGbMM* ist, dass für die Aufenthaltsdauer innerhalb eines Cluster ein Erwartungswert existiert. Für eine Gleichverteilung auf dem Intervall  $[t_{min}, t_{max}]$  kann der Erwartungswert wie folgt berechnet werden:

$$\frac{t_{min} + t_{max}}{2}$$

Demnach kann die mittlere Aufenthaltsdauer  $t_s(v)$  eines Clusters  $v$  und  $t_s(e)$  einer Kante  $e$  wie folgt definiert werden:

$$t_s(v) = \frac{t_{min}(v) + t_{max}(v)}{2}, \forall v \in V$$
$$t_s(e) = 0, \forall e \in E$$

### Mittlere Zeit zum Verlassen eines Elementes

Nachdem die Aufenthaltsdauer innerhalb eines Clusters abgelaufen ist oder eine Kante betreten wird, verlässt ein mobiler Knoten das entsprechende Element und geht umgehend zu einem Ausgang. In diesem Abschnitt wird untersucht wie viel Zeit ein mobiler Knoten im Mittel benötigt um ein Cluster oder eine Kante eines *AGbMM* zu verlassen.

Um von einem zum anderen Ende einer Kante zu gehen, muss ein mobiler Knoten über die gesamte Kante und somit über die gesamte Länge laufen. Dazu wählt der Knoten beim Betreten der Kante eine zufällige gleichverteilte Geschwindigkeit aus dem Intervall  $[v_{min}, v_{max}]$ . Das bedeutet, die Zeit zum Überqueren einer Kante berechnet sich aus Länge der Kante durch die entsprechende Geschwindigkeit. Zur Berechnung der mittleren Zeit muss demzufolge der Erwartungswert der Geschwindigkeit ermittelt werden. Bei einer Gleichverteilung auf dem Intervall  $[v_{min}, v_{max}]$  ist der Erwartungswert der Geschwindigkeit gleich  $(v_{min} + v_{max})/2$ . Demzufolge kann für eine Kante  $e$  mit der Länge  $l(e)$  und einer minimalen und maximalen Geschwindigkeit  $v_{min}$  und  $v_{max}$  die mittlere Zeit zum Verlassen  $t_l(e)$  wie folgt berechnet werden:

$$t_l(e) = \frac{2 \cdot l(e)}{v_{min}(e) + v_{max}(e)}, \forall e \in E$$

Zur Bestimmung der Geschwindigkeit auf einer Kante kann auch eine andere Wahrscheinlichkeitsverteilung verwendet werden, wenn für diese Verteilung ein Erwartungswert existiert.

Die Berechnung der mittleren Zeit zum Verlassen eines Clusters ist weit aus komplexer und von deutlich mehr Attributen des *AGbMM* abhängig. Grundlegend lässt sich  $t_l(v)$  für ein Cluster  $v$  wie folgt berechnen:

$$t_l(v) = \frac{d(v)}{(v_{min}(v) + v_{max}(v))/2}, \forall v \in V$$

Hierbei sind  $v_{min}$  und  $v_{max}$  die minimale und maximale Geschwindigkeit innerhalb eines Clusters und  $d$  die durchschnittliche Länge des Weges zum Ausgang nach Ablauf der Aufenthaltsdauer. Die durchschnittliche Länge  $d(v)$  hängt dabei sowohl von der Aufenthaltsdauer, der Fläche, der Geschwindigkeit, der Wartezeit und der Anzahl und Gewichtung der Ausgänge eines Clusters  $v$  ab. So sind Szenarien denkbar, bei denen die Aufenthaltsdauer innerhalb eines Clusters so gering ist, dass die mobilen Knoten nach wenigen Schritten das Cluster wieder verlassen. Solch ein Szenario führt dazu, dass die Dichte mobiler Knoten um die Ein- und Ausgänge deutlich höher ist als in die anderen Bereichen des Clusters. Demzufolge wäre  $d(v)$  kleiner als bei einem Szenario mit einer längeren Aufenthaltsdauer. Weil die Berechnung eines exakten Wertes für  $d$  überaus

komplex ist, werden im Folgenden verschiedene Ansätze zur Approximation von  $d$  vorgestellt.

### **Mittlere Länge des Weges zum Ausgang innerhalb eines Clusters**

Zunächst wird ein Verfahren beschrieben, um eine sehr genaue Approximation der mittleren Länge des Weges zum Ausgang  $d$  zu berechnen. Weil diese Berechnung außerhalb des Rahmens der Diplomarbeit liegt, werden im Anschluss einfache zu berechnende Approximationen vorgestellt.

Davon ausgehend, dass die Aufenthaltsdauer eines Clusters so hoch ist, dass die meisten Knoten wenigstens zu einer Position laufen, ist folgende Vorgehensweise eine sehr gute Annäherung zur Berechnung von  $d(v)$ . Zunächst wird das Verhältnis zwischen der durchschnittlichen Wartezeit und der durchschnittlichen Bewegungszeit der Knoten innerhalb eines Clusters  $v$  berechnet. Die durchschnittliche Wartezeit ist dabei der Mittelwert von minimaler und maximaler Wartezeit und die durchschnittliche Bewegungszeit ist die durchschnittliche Länge einer Strecke innerhalb des Clusters, geteilt durch die mittlere Geschwindigkeit. Das heißt, dieses Verhältnis zwischen mittlerer Wartezeit und mittlerer Bewegungszeit gibt an, wie viele der Knoten sich nach Ablauf der Aufenthaltsdauer in Bewegung befinden und wie viele der Knoten warten. Für beide Fälle (Knoten wartet oder Knoten ist in Bewegung) muss die durchschnittliche Entfernung zu den Ausgängen des Clusters berechnet werden. Für den Fall, dass ein mobiler Knoten nach Ablauf der Aufenthaltsdauer wartet, sind alle Positionen innerhalb des Clusters gleich wahrscheinlich. Die mittlere Entfernung lässt sich somit mittels Integralrechnung über alle Positionen des Clusters berechnen. Ist die mittlere Entfernung zu allen Ausgängen berechnet, müssen die Gewichtungen der einzelnen Kanten berücksichtigt werden. Für den Fall, dass die Aufenthaltsdauer eines Knotens während der Bewegung abläuft, muss die durchschnittliche Entfernung zu den einzelnen Ausgängen berechnet werden. Dazu muss der Mittelwert aller möglichen Wege und deren Wahrscheinlichkeiten berücksichtigt werden. Anschließend kann dann mit Hilfe der Gewichtung der einzelnen Kanten die mittlere Entfernung zu einem beliebigen Ausgang berechnet werden. Zuletzt wird der Mittelwert der Entfernung im wartenden und im bewegten Zustand mittels dem Verhältnisses von Wartezeit und Bewegungszeit berechnet. Das Ergebnis wäre eine sehr gute Approximation für die durchschnittliche Länge des Weges zu einem zufälligen Ausgang dieses Clusters.

Weil der Schwerpunkt der Arbeit auf der Nachrichtenverteilung im *AGbMM* liegt, soll an dieser Stelle das zuvor beschriebene Verfahren nicht genauer untersucht werden. Vielmehr könnte dieses Verfahren in folgenden Arbeiten zum *AGbMM* untersucht werden, wenn diese sich detaillierter mit der Berechnung der mittleren Knotenanzahl beschäftigen. Des Weiteren haben verschiedene Simulationen gezeigt, dass sich mit einer Approximation von  $d$  bereits sehr genaue Ergebnisse berechnen lassen. Somit wird im Folgenden eine Formel

## Verteilung der mobilen Knoten

---

zur Berechnung eines approximativen Wertes  $\tilde{d}$  von  $d$  vorgestellt. Dieser Wert  $\tilde{d}$  lässt sich im Vergleich zu  $d$  einfacher berechnen und ist für die meisten praktischen Szenarien eine ausreichende Näherung für  $d$ . Zusätzlich zu  $\tilde{d}$  wird noch eine obere Schranke  $\bar{d}$  und eine untere Schranke  $\underline{d}$  zur Berechnung des mittleren Weges zum Ausgang vorgestellt. Mit diesen Schranken kann bei späteren Berechnungen die durchschnittliche Knotenanzahl eingegrenzt werden.

Wie im Kapitel zu den einfachen Bewegungsmodellen beschrieben, ist die Dichte der Knoten im Mittelpunkt eines *Random Waypoint Mobility Models* höher als an den Rändern des Modells. Weil die Bewegung der Knoten beim *AGbMM* innerhalb der Cluster ebenfalls durch ein *Random Waypoint Mobility Models* beschrieben wird, ist die Konzentration der Knoten innerhalb eines Clusters im Zentrum in gleicher Weise höher als an den Rändern der Cluster. Sind die Parameter eines Clusters  $\nu$  so gewählt, dass sich die mobilen Knoten nach Betreten des Clusters im Mittel weit genug von den Eingängen entfernen, kann  $\tilde{d}_\nu$  für eine ausgehende Kante  $e$  des Clusters  $\nu$  wie folgt definiert werden:

$$\tilde{d}_\nu(e) = \sqrt{|v_x - e_x|^2 + |v_y - e_y|^2}$$

Hierbei sind  $v_x$  und  $v_y$  die x und y Koordinaten des Mittelpunktes des Clusters  $\nu$  und  $e_x$  und  $e_y$  die x und y Koordinaten des Startpunktes der Kante  $e$ . Das heißt,  $\tilde{d}_\nu(e)$  entspricht dem euklidischen Abstand vom Mittelpunkt des Clusters  $\nu$  zum Startpunkt der Kante  $e$ . Wenn ein Cluster  $\nu$  mehr als eine inzidente Kante hat, berechnet sich  $\tilde{d}$  für dieses Cluster als Mittelwert aus allen Strecken zu den einzelnen Ausgängen des Clusters. Wobei die Wege zu den einzelnen Ausgängen entsprechend der dazugehörige Kanten gewichtet werden.

$$\tilde{d}(\nu) = \sum_{e \in E_\nu} (w(e) \cdot \tilde{d}_\nu(e)) = \sum_{e \in E_\nu} (w(e) \cdot \sqrt{|v_x - e_x|^2 + |v_y - e_y|^2})$$

Dabei ist  $E_\nu$  die Menge aller ausgehenden Kanten von  $\nu$  und  $w(e)$  die Gewichtung der Kante  $e$ .

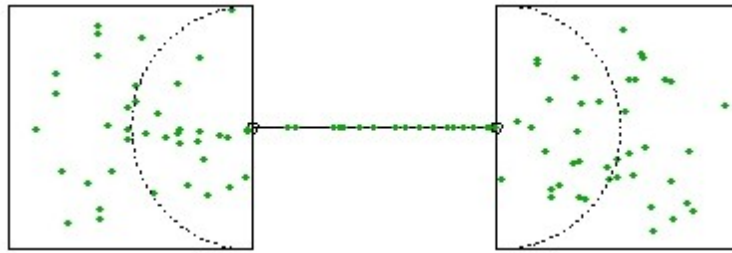


Abbildung 15: Approximation des mittleren Weges zum Ausgang

In Abbildung 15 ist der Wert von  $\tilde{d}$  für das Modell aus Abbildung 14 veranschaulicht, dazu sind um die Ausgänge der Cluster Halbkreise mit einem Radius  $\tilde{d}$  eingezeichnet. Man sieht, dass diese Approximation faktisch eine untere Schranke des genauen Wertes von  $d$  sein muss. Unter der Annahme, dass alle Knoten innerhalb des Clusters symmetrisch verteilt sind, befinden sich durchschnittlich mehr mobile Knoten außerhalb als innerhalb des Halbkreises. Somit haben mehr mobile Knoten einen längere Weg zum Ausgang als der Wert  $\tilde{d}$  angibt. Weil sich aber ungeachtet einer hohen Aufenthaltsdauern immer eine gewisse Anzahl mobiler Knoten auf dem Weg zum Ausgang bzw. vom Eingang befindet, ist in nahezu allen Modellen die Konzentration der mobilen Knoten im Bereich der Ein- und Ausgänge höher als im Rest des Clusters. Somit sollten sich im Mittel die gleiche Anzahl Knoten innerhalb und außerhalb des Kreises befinden. Aus  $\tilde{d}$  lässt sich nun  $\tilde{t}_l$ , als Approximation für die durchschnittliche Zeit zum Verlassen eines Clusters, wie folgt definieren:

$$\tilde{t}_l(v) = \frac{\tilde{d}(v)}{(v_{min}(v) + v_{max}(v))/2} = \frac{2 \cdot \sum_{e \in E_v} (w(e) \cdot \sqrt{|v_x - e_x|^2 + |v_y - e_y|^2})}{v_{min}(v) + v_{max}(v)}$$

Für den weiteren Verlauf der Arbeit wird außerdem mit einer unteren und einer oberen Schranke für den durchschnittlichen Weg zum Ausgang gearbeitet. Zum einen vereinfachen sie die Berechnungen der durchschnittlichen Knotenanzahl und zum anderen ermöglichen sie eine Abschätzung des exakten Wertes innerhalb eines bestimmten Bereiches. Als untere Schranke  $\underline{d}$  wird eine Entfernung von 0m zum Ausgang angenommen, d.h. die Knoten würden sich direkt nach Ablauf der Aufenthaltsdauer am Ausgang befinden. Daraus folgt, dass für die untere Schranke  $\underline{t}_l$  von  $t_l$  gilt:

$$\underline{t}_l(v) = \frac{0}{(v_{min}(v) + v_{max}(v))/2} = 0$$

Als obere Schranke  $\bar{d}$  wird die maximale Entfernung zweier Punkte innerhalb eines Clusters definiert:  $\bar{d}(v) = \sqrt{width(v)^2 + height(v)^2}$ . Hierbei entsprechen  $width(v)$  und

## Verteilung der mobilen Knoten

---

$height(v)$  der Länge und Breite des jeweiligen Clusters.  $\bar{d}$  entspricht somit der Entfernung zwischen der linken oberen und rechten unteren Koordinate eines Clusters. Als obere Schranke  $\bar{t}_l$  von  $t_l$  ergibt sich somit:

$$\bar{t}_l = \frac{2 \cdot \sqrt{width(v)^2 + height(v)^2}}{(v_{min}(v) + v_{max}(v))}$$

Mit Hilfe der mittleren Aufenthaltsdauer  $t_s$  und der durchschnittlichen Zeit zum Verlassen  $t_l$  kann jetzt die mittlere Verweildauer  $t$  eines mobilen Knotens berechnet werden.

### Berechnung der mittlere Verweildauer

Die mittlere Verweildauer eines mobilen Knotens innerhalb eines Elementes des *AGbMM* ist die Summe aus mittlerer Aufenthaltsdauer  $t_s$  und durchschnittlicher Zeit zum Verlassen  $t_l$ . Demnach lassen sich aus  $\tilde{t}_l$ ,  $\underline{t}_l$ ,  $\bar{t}_l$  die Approximationen  $\tilde{t}$ ,  $\underline{t}$  und  $\bar{t}$  für die mittlere Verweildauer eines mobilen Knotens wie folgt definieren:

$$\tilde{t}(v) = t_s(v) + \tilde{t}_l(v) = \frac{t_{min}(v) + t_{max}(v)}{2} + \frac{2 \cdot \sum_{e \in E_v} (w(e) \cdot \sqrt{|v_x - e_x|^2 + |v_y - e_y|^2})}{v_{min}(v) + v_{max}(v)}$$

$$\underline{t}(v) = t_s(v) + \underline{t}_l(v) = \frac{t_{min}(v) + t_{max}(v)}{2}$$

$$\bar{t}(v) = t_s(v) + \bar{t}_l(v) = \frac{t_{min}(v) + t_{max}(v)}{2} + \frac{2 \cdot \sqrt{width(v)^2 + height(v)^2}}{(v_{min}(v) + v_{max}(v))}$$

Mit Hilfe der mittleren Verweildauer eines mobilen Knotens innerhalb eines Elementes des *AGbMM* kann nun für ein Modell wie in Abbildung 14 die Verteilung der mobilen Knoten auf die einzelnen Elemente berechnet werden.

### Berechnung der mittlere Knotenanzahl

Betrachtet man die Bewegung eines mobilen Knotens durch das *AGbMM* von Abbildung 14, so verläuft diese wie folgt: Der Knoten befindet sich zu Beginn in einem zufällig gewählten Cluster  $v$ , wobei  $v \in \{V_1, V_2\}$ . Nach Ablauf der Aufenthaltsdauer in  $v$  verlässt der Knoten das Cluster und geht in Richtung  $w$  mit  $w \in \{V_1, V_2\}$ ,  $w \neq v$ , dabei nutzt der Knoten die Kante  $(v, w)$ . Nachdem der mobile Knoten das Cluster  $w$  betreten hat, erhält er für dieses Cluster eine zufällige, gleichverteilte Aufenthaltsdauer. Sobald diese Aufenthaltsdauer abgelaufen ist, verlässt der Knoten das Cluster  $w$  und geht über die Kante  $(w, v)$  zum Cluster  $v$ . Anschließend beginnt der Vorgang in Cluster  $v$  von vorn. Das bedeutet, der mobile Knoten betritt alle Elemente des *AGbMM* einmal. Die einzigen

## Verteilung der mobilen Knoten

---

zufälligen Faktoren bei der Bewegung des Knotens sind die Aufenthaltsdauern, Geschwindigkeiten und Positionen innerhalb der Cluster und die Geschwindigkeiten auf den Kanten. Der mobile Knoten hat keine Wahl, in welcher Reihenfolge er die Elemente des Graphen betritt. Die Reihenfolge der Elemente ist immer  $v, (v,w), w, (w,v), v$  usw., wobei  $v, w \in (V_1, V_2)$  mit  $v \neq w$  (Szenarien, bei denen sich die mobilen Knoten auf verschiedenen Wegen durch den Graphen bewegen können werden in Abschnitt 5.2 untersucht). Daraus folgt, dass ein Knoten für einen Rundgang durch das Modelle im Mittel

$T = \sum_{a \in A} t(a)$  Zeit benötigt. Berechnet man nun das Verhältnis der durchschnittlichen

Verweildauer eines Elements  $t(a)$  zu  $T$ , erhält man die mittlere Wahrscheinlichkeit, dass sich ein Knoten im Element  $a$  aufhält. Demzufolge befindet sich zu  $100 \cdot t(a)/T$  Prozent ein mobiler Knoten im Element  $a$  des *AGbMM* der Abbildung 14. Daraus lässt sich nun, bei einer Gesamtzahl von  $N$  mobilen Knoten im Modell, die durchschnittliche Knotenanzahl  $n(a)$  für ein Element  $a \in A$  wie folgt bestimmen:

$$n(a) = N \frac{t(a)}{T} = N \frac{t(a)}{\sum_{x \in A} t(x)}$$

Das heißt, die mittlere Knotenanzahl eines Elements  $a$  berechnet sich aus dem Anteil der Verweildauer des Elements zur Gesamtdauer eines Rundgangs multipliziert mit der Gesamtzahl mobiler Knoten im Modell. Auf analoge Art und Weise lassen sich nun  $\tilde{n}$ ,  $\underline{n}$  und  $\bar{n}$  mittels  $\tilde{t}$ ,  $\underline{t}$  und  $\bar{t}$  definieren.

Als Beispiel soll jetzt  $\tilde{n}(V_1)$  für das Modell der Abbildung 14 berechnet werden. Dazu seien, wie in Zeile 1 der Tabelle 1, die minimale bzw. maximale Aufenthaltsdauer von  $V_1$  und  $V_2$  aus dem Intervall von 0 bis 60 Sekunden. Außerdem seien die Cluster  $V_1$  und  $V_2$  10m lang und 10m breit, die Kanten  $(V_1, V_2)$  und  $(V_2, V_1)$  jeweils 10m lang und die Geschwindigkeit in den Clustern und auf den Kanten sei genau 1m/s.  $\tilde{n}(V_1)$  wird nun wie folgt berechnet:

$$\tilde{n}(V_1) = N \frac{\tilde{t}(V_1)}{\sum_{x \in A} \tilde{t}(x)}$$

Das heißt, es muss vorerst  $\tilde{t}(a)$  für alle Elemente aus  $a \in A = \{V_1, V_2, (V_1, V_1), (V_2, V_1)\}$  berechnet werden. Für  $\tilde{t}(V_1)$  funktioniert dies wie folgt:



## Verteilung der mobilen Knoten

---

$$\begin{aligned}
 \tilde{t}(V_1) &= \frac{t_{\min}(V_1)+t_{\max}(V_1)}{2} + \frac{2 \cdot \tilde{d}}{v_{\min}(V_1)+v_{\max}(V_1)} \\
 &= \frac{0\text{s}+60\text{s}}{2} + \frac{2 \cdot 5\text{m}}{2 \frac{\text{m}}{\text{s}}} \\
 &= 35\text{s}
 \end{aligned}$$

Analog lässt sich  $\tilde{t}(V_2)=35\text{s}$  ausrechnen. Für die Kante  $(V_1, V_2)$  gilt:

$$\begin{aligned}
 \tilde{t}((V_1, V_2)) &= \frac{l((V_1, V_2))}{(v_{\min}((V_1, V_2))+v_{\max}((V_1, V_2)))/2} \\
 &= \frac{10\text{m}}{1\text{m/s}} \\
 &= 10\text{s}
 \end{aligned}$$

Ebenso ist  $\tilde{t}((V_2, V_1))=10\text{s}$ . Insgesamt folgt daraus für  $\sum_{a \in A} \tilde{t}(a)$ :

$$\begin{aligned}
 \sum_{a \in A} \tilde{t}(a) &= \tilde{t}(V_1) + \tilde{t}(V_2) + \tilde{t}((V_1, V_2)) + \tilde{t}((V_2, V_1)) \\
 &= 35\text{s} + 35\text{s} + 10\text{s} + 10\text{s} \\
 &= 90\text{s}
 \end{aligned}$$

Setzt man die Werte für  $\tilde{t}(V_1)$  und  $\sum_{a \in A} \tilde{t}(a)$  in  $\tilde{n}(V_1)$  ein ergibt sich:

$$\tilde{n}(V_1) = N \frac{35\text{s}}{90\text{s}} = \frac{7}{18} N$$

Das bedeutet, im Mittel befinden sich  $7/18$  oder 38,8% aller Knoten in Cluster  $V_1$ . Verglichen mit dem Wert der Simulation der Tabelle 2 entspricht dies einer Abweichung von weniger als 2 Prozent.

In Tabelle 2 sind neben  $\tilde{n}(V_1)$ , auch  $\underline{n}(V_1)$  und  $\bar{n}(V_1)$  für das Cluster  $V_1$  des *AGbMM* aus Abbildung 14 angegeben. Insgesamt wurde die Bewegung von 100 mobilen Knoten simuliert und die Parameter und Aufenthaltsdauern des Modells entsprechend denen der Tabelle 1 gewählt. Zusätzlich ist in der 3. Spalte der Tabelle 2 noch einmal der Messwert der durchschnittlichen Knotenanzahl aus Tabelle 1 angegeben. Somit lassen sich die berechneten Ergebnisse direkt mit dem Messwert aus Tabelle 1 vergleichen.

## Verteilung der mobilen Knoten

Tabelle 2: Vergleich der Approximationen mit dem Messwert der Knotenverteilung eines Area Graph based Mobility Models

Aufenthaltsdauer in $V_1$ (in Sekunden)	Aufenthaltsdauer in $V_2$ (in Sekunden)	Mittlere Knotenanzahl in $V_1$			
		Messwert $n(V_1)$	Approximation $\tilde{n}(V_1)$	Untere Schranke $\underline{n}(V_1)$	Obere Schranke $\bar{n}(V_1)$
0-60	0-60	39,1947	38,8889	37,5000	40,7650
0-120	0-120	43,3288	43,3333	42,8571	44,0577
0-60	0-120	29,5280	29,4884	27,2727	31,9213
0-60	0-180	24,0782	23,3333	21,4286	26,2307
0-120	0-240	31,1915	30,9524	30,0000	32,4780

In Tabelle 2 ist zu erkennen, dass der gemessene Wert für  $V_1$  immer innerhalb der unteren und oberen Schranke  $\underline{n}$  und  $\bar{n}$  liegt. Die Approximation  $\tilde{n}$  hat im Mittel eine Abweichung von weniger als einem Prozent gegenüber dem Messwert des Clusters  $V_1$ . Durch den kleineren Einfluss des Hinausgehens in Zeile 2 gegenüber der Zeile 1 ist die Differenz zwischen  $\underline{n}$  und  $\bar{n}$  von 3,265 in Zeile 1 auf 1,2006 in Zeile 2 gesunken. Das bedeutet, je geringer der Einfluss der Zeit zum Verlassen eines Clusters, um so präziser kann der exakte Wert der mittleren Knotenanzahl vorausgesagt werden. Vergleicht man im Gegensatz dazu Zeile 1 mit Zeile 3 oder Zeile 4, zeigt sich, dass bei steigender Aufenthaltsdauer eines anderen Clusters (hierbei  $V_2$ ) die Schranken ungenauer werden. Die Differenz von  $\underline{n}$  und  $\bar{n}$  liegt in Zeile 1 bei 3,265, in Zeile 3 bereits bei 4,6486 und in Zeile 4 sogar bei 4,8021. Für Zeile 4 ist auch die Diskrepanz zwischen  $\tilde{n}$  und dem Messwert mit rund 3,2% am größten. Demnach kann für eine *AGbMM* wie in Abbildung 14 die mittlere Knotenanzahl der Elemente mit einer Abweichung von weniger als 5% vorhergesagt werden.

## 5.2 Analyse des Verbindungsgrades

In diesem Abschnitt werden die Eigenschaften des *AGbMM* genauer untersucht, die Auswirkungen auf den Grad der Verbindung zwischen den einzelnen Elementen haben. Diese Eigenschaften sind: die Topologie des Graphen und die Gewichtung der Kanten. Durch die Untersuchungen in diesem Abschnitt lassen sich nicht, wie im vorherigen Abschnitt, konkrete Verteilungen der Knoten berechnen, vielmehr soll das Verhältnis der Knotenanzahl einzelner Elemente zueinander analysiert und daraus eine allgemeine Formel

zur Berechnung der mittleren Knotenanzahl hergeleitet werden. Dazu wird als erstes die Topologie des Graphen anhand ausgewählter Beispiele untersucht.

### Auswirkung der Topologie des Graphen

Zur Analyse der Verweildauer in Abschnitt 5.1 wurde ein einfaches Modell mit zwei Knoten und zwei Kanten verwendet (siehe Abbildung 14). Dieses Modell wurde gewählt, weil es von der Struktur das einfachste, symmetrische *AGbMM* ist, welches sowohl aus Knoten als auch aus Kanten besteht. Durch diese Anordnung und die damit verbundene gleichmäßige Verteilung der mobilen Knoten auf die beiden Cluster, konnte die Auswirkung der Verweildauer auf die Knotenverteilung direkt untersucht werden. Wird diesem Modell ein weiterer Knoten und zwei zusätzliche Kanten hinzugefügt, wie in Abbildung 16 dargestellt, ändert sich die Verteilung der mobilen Knoten grundlegend. Das Modell ist weiterhin symmetrisch. Die Clustern  $V_1$  und  $V_3$  können jeweils über eine Kante nur in Richtung  $V_2$  verlassen werden. Im Gegensatz dazu kann  $V_2$  über zwei unterschiedliche Kanten verlassen werden, eine in Richtung  $V_1$  und eine in Richtung  $V_3$ . Die Wahrscheinlichkeit, dass  $(V_2, V_1)$  bzw.  $(V_2, V_3)$  beim Verlassen von  $V_2$  benutzt wird, sei bei diesen Betrachtungen gleich wahrscheinlich. Das bedeutet, 50% aller Knoten verlassen  $V_2$  nach  $V_1$  und 50% nach  $V_3$ . Betrachtet man nun die Bewegung eines einzelnen Knotens durch das Modell und dabei insbesondere die Bewegung durch die einzelnen Cluster, so lässt sich Folgendes feststellen: Von  $V_1$  und  $V_3$  kann der Knoten jeweils nur nach  $V_2$  gehen. Ausgehend von  $V_2$  kann der Knoten zu 50% nach  $V_1$  und zu 50% nach  $V_3$  gehen. Anschließend muss der Knoten dann wieder  $V_1$  bzw.  $V_3$  in Richtung  $V_2$  verlassen. Das bedeutet also, im Mittel befindet sich ein Knoten doppelt so oft im Cluster  $V_2$  wie in den Clustern  $V_1$  und  $V_3$ . Nimmt man nun an, dass die durchschnittliche Verweildauer in den Clustern  $V_1$ ,  $V_2$  und  $V_3$  gleich groß ist, so folgt daraus, dass im Mittel doppelt so viele Knoten in  $V_2$  sind wie in  $V_1$  und  $V_3$ .

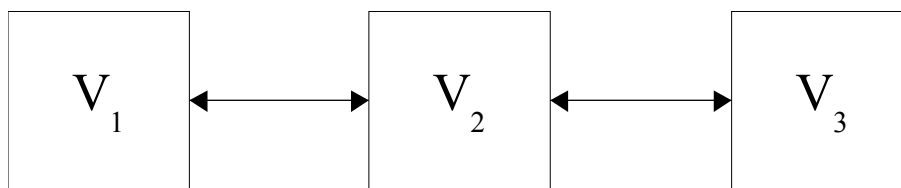


Abbildung 16: Area Graph-based Mobility Model zur Analyse des Verbindungsgrades

In gleicher Weise lässt sich die Bewegung eines Knotens auf den Kanten des Graphen betrachten. Von  $V_1$  muss der Knoten über die Kante  $(V_1, V_2)$  gehen. Genauso muss der Knoten das Cluster  $V_3$  über die Kante  $(V_3, V_2)$  verlassen. Das heißt, ein Knoten befindet sich in etwa gleich häufig im Cluster  $V_1$  und  $V_3$  wie auf den Kanten  $(V_1, V_2)$  und  $(V_3, V_2)$ . Ausgehend von  $V_2$  geht ein Knoten zu 50% zum Cluster  $V_1$  und muss dazu die Kante  $(V_2, V_1)$  nutzen und die weiteren 50% geht er zum Cluster  $V_3$  und benutzt dabei die Kante

## Verteilung der mobilen Knoten

---

$(V_2, V_3)$ . Demzufolge werden die Kanten  $(V_2, V_1)$  und  $(V_2, V_3)$  halb so oft betreten wie das Cluster  $V_2$ , und somit genauso oft wie die Kanten  $(V_1, V_2)$  und  $(V_3, V_2)$ . Insgesamt folgt daraus, dass ein Knoten das Cluster  $V_2$  durchschnittlich zweimal so oft nutzt wie alle anderen Elemente (Knoten und Kanten) des Graphen. Gilt zusätzlich, dass alle Cluster die gleiche mittlere Verweildauer haben, so ist die mittlere Knotenanzahl in  $V_2$  doppelt so groß wie in  $V_1$  und  $V_3$ . Wenn alle Kanten die gleiche durchschnittliche Verweildauer haben, so folgt daraus, dass die mittlere Knotenanzahl auf allen Kanten gleich groß ist.

In Tabelle 3 ist die Knotenverteilung des Modells von Abbildung 16 für verschiedene Knotenanzahlen abgebildet. Dazu wurden 20 verschiedene Simulationsläufe mit jeder Knotenanzahl  $N$  durchgeführt und anschließend der Mittelwert der einzelnen Läufe bestimmt. Um die theoretischen Annahmen überprüfen zu können, müssen zwei Eigenschaften berücksichtigt werden. Zum einen müssen alle ausgehenden Kanten eines Clusters gleich wahrscheinlich sein, was sich durch entsprechende Wahl der Kantengewichte realisieren lässt. Zum anderen müssen die Verweildauern der Cluster und die der Kanten jeweils gleich groß sein. Dazu erhalten alle Clustern die gleichen Parameter. Diese sind: eine Fläche von 10m mal 10m, eine Aufenthaltsdauer von 0 bis 60 Sekunden, eine Geschwindigkeit der Knoten von 1m/s und eine Wartezeit von 0 bis 10 Sekunden. In gleicher Weise erhalten alle Kanten die gleichen Parameter um eine identische Verweildauer zu garantieren. Die Kanten sind dazu jeweils 10m lang und die mobilen Knoten haben auf den Kanten eine Geschwindigkeit von genau 1m/s. Bei allen Simulationen wurde mit der Messung der mittleren Knotenanzahl nach 300 Sekunden begonnen, um eine Abweichungen durch die initiale Positionierung der Knoten zu vermeiden. Anschließend wurde für 3600 Sekunden die durchschnittlichen Knotenanzahlen gemessen.

*Tabelle 3: Messwerte zur Analyse des Verbindungsgrades*

$N$	$V_1$	$V_2$	$V_3$	$(V_1, V_2)$	$(V_2, V_1)$	$(V_2, V_3)$	$(V_3, V_2)$
100	19,4535	39,1230	19,7733	5,4257	5,3844	5,4310	5,3976
300	58,7940	117,2112	58,9103	16,2551	16,1865	16,3692	16,2733
500	97,8502	195,4139	98,3185	27,0272	27,1654	27,0969	27,1246

Entsprechend der theoretischen Vorhersagen ist für alle  $N$  die durchschnittliche Knotenanzahl von  $V_1$  und  $V_3$  in etwa gleich groß. Der Messwert für  $V_2$  ist in allen Zeilen ungefähr doppelt so groß wie die Werte von  $V_1$  und  $V_3$  und wie vermutet, ist die durchschnittliche Knotenanzahl auf den Kanten annähernd identisch.

## Verteilung der mobilen Knoten

Weil alle Elemente bis auf  $V_2$  im Mittel gleich oft betreten werden, lässt sich aus dem Verhältnis der durchschnittlichen Knotenanzahl zweier Elemente (ausgeschlossen  $V_2$ ) auf das Verhältnis der Verweildauern dieser Elemente zueinander schließen. So ist das Verhältnis von  $V_1$  und  $V_2$  in allen Zeilen ungefähr 1, demzufolge muss die Verweildauer gleich sein. Genau das sollte auch durch die Wahl der Parameter erreicht werden. Das Gleiche gilt für das Verhältnis der Kanten untereinander. Weil die Verweildauern der Kanten gleich sind, ist das Verhältnis der Knotenanzahlen jeweils 1. Interessant ist ein Vergleich des Verhältnisses der mittleren Knotenanzahl von  $V_1$  und  $(V_1, V_2)$ . Der Wert liegt in allen Zeilen zwischen 3,5 und 3,7. Das bedeutet, die Verweildauer eines Knotens in  $V_1$  ist im Schnitt 3,6 mal so groß wie die von  $(V_1, V_2)$ , weil im Mittel beide Elemente mit gleicher Häufigkeit betreten werden.

Als weitere Beispiele zur Analyse des Verbindungsgrades werden die Modelle aus Abbildung 17 betrachtet. Beide Modelle bestehen jeweils aus vier Clustern ( $V_1$ ,  $V_2$ ,  $V_3$  und  $V_4$ ), die in quadratischer Form angeordnet sind. In Modell A sind jeweils zwei benachbarte Cluster  $v$  und  $w$  durch die Kanten  $(v,w)$  und  $(w,v)$  in beide Richtungen miteinander verbunden. Das Modell A hat somit insgesamt acht Kanten. Beim Modell B hingegen sind die Cluster  $V_3$  und  $V_4$  nicht direkt miteinander verbunden, so dass das Modell nur über sechs Kanten verfügt.

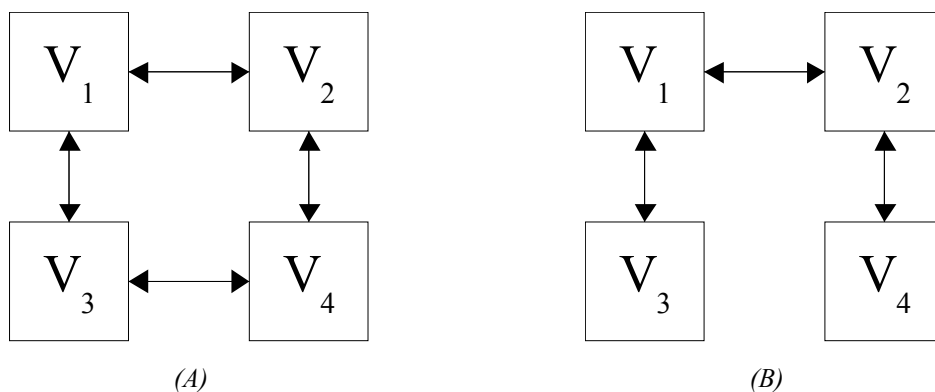


Abbildung 17: Auswirkung verschiedener Topologien auf die Verteilung der Knoten innerhalb eines AGbMM

Obwohl sich die beiden Modelle nur durch die Verbindung zweier Cluster unterscheiden, verläuft die Bewegung eines mobilen Knotens in den Modellen sehr unterschiedlich. Bewegt sich ein mobiler Knoten in Modell A, so hat er ausgehend von einem Cluster immer zwei mögliche Wege um dieses zu verlassen. Weil die Knoten und Kanten des Graphen zusätzlich einen Kreis bilden, gibt es immer zwei mögliche Wege zwischen zwei verschiedenen Knoten des Graphen. Zum Beispiel kann ein Knoten von  $V_1$  nach  $V_4$  gelangen, indem er über Cluster  $V_2$  oder über Cluster  $V_3$  geht. Wenn also beim Verlassen eines Clusters alle Kanten gleich wahrscheinlich sind, folgt daraus, dass im Mittel alle Cluster mit gleicher Häufigkeit betreten werden. Weil zusätzlich alle Cluster zwei Ausgänge

## Verteilung der mobilen Knoten

---

haben, ergibt sich daraus, dass die Kanten durchschnittlich halb so oft betreten werden wie die Knoten des Graphen.

Wenn man das Modell B als einen ungerichteten Graphen betrachtet, so ist dieser azyklisch und demzufolge ein Baum. Die mobilen Knoten im Modell B können auf nur einem Weg von einem Knoten des Graphen zu einem anderen gelangen. Von  $V_3$  bzw.  $V_4$  kann ein mobiler Knoten nur nach  $V_1$  bzw.  $V_2$  gehen und das immer nur über die inzidente Kante. Wenn man zusätzlich von gleichverteilten Kantengewichten ausgeht, geht die Hälfte aller Knoten von  $V_1$  nach  $V_2$  und die andere Hälfte nach  $V_3$ . Analog gehen 50% der Knoten von  $V_2$  nach  $V_1$  und 50% nach  $V_4$ . Ein Knoten ist demzufolge doppelt so oft in den Clustern  $V_1$  und  $V_2$  wie in den Clustern  $V_3$  und  $V_4$ .

In Tabelle 4 sind die Messwerte der einzelnen Cluster  $V_1$ ,  $V_2$ ,  $V_3$  und  $V_4$  der Modelle A und B aufgezeigt. Es wurden dazu 20 verschiedene Simulationsläufe mit insgesamt 400 mobilen Knoten und einer Simulationszeit von einer Stunde durchgeführt. Weil hierbei nur die Auswirkung der Topologie des Graphen untersucht werden soll, sind die Gewichte aller ausgehenden Kanten eines Clusters gleichverteilt.

Tabelle 4: Mittlere Knotenanzahl der Elemente der Modelle A und B

Modell	Messwert für $V_1$	Messwert für $V_2$	Messwert für $V_3$	Messwert für $V_4$
A	79,6422	79,3530	79,0461	79,7970
B	106,1196	105,6711	52,1998	53,3376

Wie erwartet, ist die Anzahl mobiler Knoten im Modell A in allen Clustern gleich groß. Ebenso ist im Modell B, entsprechend der Vorhersage, die mittlere Anzahl mobiler Knoten in den Clustern  $V_1$  und  $V_2$  in etwa doppelt so groß wie in den Clustern  $V_3$  und  $V_4$ . Analog kann für ein beliebiges Modell die Bewegung eines mobilen Knotens analysiert werden und daraus die mittlere Verteilung auf die einzelnen Elemente vorhergesagt werden.

### Gewichtung entsprechend der Eingänge eines Elementes

Ein genauerer Vergleich der Modelle aus Abbildung 17 zeigt, dass dem Modell B gegenüber Modell A die Kanten  $(V_3, V_4)$  und  $(V_4, V_3)$  fehlen. Das bedeutet, die Cluster  $V_3$  und  $V_4$  haben jeweils einen Ausgang und einen Eingang weniger als die Cluster im Modell A. Somit haben die Cluster  $V_1$  und  $V_2$  im Modell B doppelt so viele Ein- bzw. Ausgänge wie die Cluster  $V_3$  und  $V_4$ . Das wiederum bedeutet, dass  $V_1$  und  $V_2$  im Mittel zwei mal so oft betreten werden wie  $V_3$  und  $V_4$  und somit eine doppelt so große Anzahl Knoten haben. Analog lässt sich für das *AGbMM* aus Abbildung 16 argumentieren. Hier hat  $V_2$  gegenüber  $V_1$  und  $V_3$  doppelt so viele Ein- bzw. Ausgänge und die Knotenanzahl ist ebenso doppelt so

## Verteilung der mobilen Knoten

---

groß wie in den anderen Clustern. Allgemein lässt sich demnach sagen: Ein Element  $a$  eines  $AGbMM$  wird während eines Rundgangs eines mobilen Knotens, durch den Graphen des  $AGbMM$ , im Mittel  $e(a)$  mal betreten. Wobei  $e(a)$  die Anzahl der Eingänge des Elements  $a$  angibt. Das gilt aber nur, wenn der Graph des Modells zusammenhängend ist, alle Kanten in beide Richtungen auftreten und alle Kantengewichte eines Clusters gleichverteilt sind.

Betrachten wir dazu ein weiteres Beispiel. In Abbildung 18 sind fünf Cluster in Form eines Kreuzes angeordnet. Das zentrale Cluster ist mit den anderen Clustern in jeweils beide Richtungen verbunden. Damit ein mobiler Knoten alle Cluster und Kanten des Graphen mindestens einmal betritt, muss der Knoten mindestens vier mal das zentrale Cluster betreten. Wenn man nun davon ausgeht, dass alle Kanten des zentralen Clusters die gleiche Wahrscheinlichkeit haben, so wird ein Knoten im Mittel jede Kante einmal benutzen bevor er alle Elemente des Graphen einmal betreten hat. Das zentrale Cluster wird im Schnitt also vier mal so oft betreten, wie die anderen Elemente des Graphen.

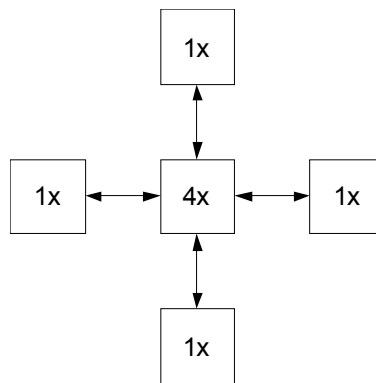


Abbildung 18: Cluster angeordnet in Form eines Kreuzes

Allgemein gilt also für zwei Elemente  $a$  und  $b$  eines  $AGbMM$  mit gleicher Verweildauer:

$$\frac{n(a)}{e(a)} = \frac{n(b)}{e(b)}$$

Wobei  $n$  die durchschnittliche Knotenanzahl eines Elements ist und  $e$  die Anzahl der Ausgänge eines Elementes angibt. Diese Formel gilt nur unter der Voraussetzung, dass der Graph des Modells zusammenhängend ist, Kanten in beide Richtungen existieren und die Kantengewichte gleichverteilt innerhalb der Cluster sind. Weil Kanten grundsätzlich nur einen Ausgang haben, folgt aus der Formel, dass Kanten mit gleicher Verweildauer auch die gleiche Knotenanzahl haben müssen. Anders ausgedrückt, Kanten mit einer höheren Verweildauer haben grundsätzlich eine größere Knotenanzahl als Kanten mit einer geringeren Verweildauer. Diese Aussagen gelten aber nur für Kanten. Denn bei zwei Clustern mit gleicher Verweildauer kann eines der Cluster mehr Eingänge und somit im Mittel mehr mobile Knoten haben.

### Auswirkung der Kantengewichte

Die zweite Eigenschaft, die Auswirkung auf den Verbindungsgrad des Graphen hat, ist die Kantengewichtung. In den vorherigen Kapiteln wurden die Kanten eines Clusters bewusst gleichverteilt gewichtet. Auf diese Weise verteilen sich die Knoten gleichmäßig innerhalb des Modells und ermöglichen eine einfache Berechnung der mittleren Knotenanzahl mittels der Verweildauer und der Anzahl der Eingänge (siehe folgendes Kapitel). Sollen die Kantengewichte bei der Vorhersage der mittleren Knotenanzahl berücksichtigt werden, ergibt sich daraus eine weitaus komplexere Berechnung. Exemplarisch sei hierzu das *AGbMM* der Abbildung 19 gegeben. Bei gleichmäßiger Kantengewichtung ist die Knotenanzahl in  $V_2$  und  $V_3$  doppelt so groß wie in  $V_1$  und  $V_4$ . Wird die Kantengewichtung im Knoten  $V_2$  dahingehend geändert, dass 66% der Knoten nach  $V_1$  gehen und 33% nach  $V_3$ , so ergibt bei insgesamt 200 mobilen Knoten im Modell folgende mittlere Verteilung: 38,54 der Knoten befinden sich in  $V_1$ , 59,90 in  $V_2$ , 40,77 in  $V_3$  und 20,33 in  $V_4$ . Diese Werte lassen sich nicht mehr durch die lokalen Parameter eines Clusters herleiten. Das bedeutet, zur Berechnung der mittleren Knotenanzahl eines Elementes muss der komplette Graph des Modells betrachtet werden.

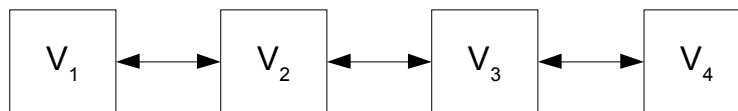


Abbildung 19: Vier Cluster in einer Reihe

Änderungen an der Gewichtung einer Kante haben demzufolge Auswirkung auf alle Elemente des Graphen. Bei allen vorherigen Berechnungen, insbesondere der Approximationen der Verweildauer oder Gewichtung der Elemente entsprechend der Anzahl der Eingänge, ist die Gewichtung eines Elementes nur von den lokalen Parametern eines Elementes abhängig. Zur Integration der Kantengewichte in die Berechnung der Knotenanzahl muss demzufolge eine andere Herangehensweise gewählt werden. Weil der Schwerpunkt der Arbeit auf der Analyse der Nachrichtenverteilung liegt, soll an dieser Stelle nur ein Ansatz vorgestellt werden, um die Kantengewichtung in die Berechnung der Knotenanzahl einzubeziehen. Zudem ist es mit den bisher vorgestellten Methoden bereits möglich, die Knotenanzahl für eine Vielzahl von Modellen zu berechnen. Darüber hinaus kann zur Ermittlung der mittleren Knotenanzahl bei Modellen mit verschiedenen Kantengewichten der Simulator aus Abschnitt 4.2 verwendet werden.

### Ansatz zur Berücksichtigung verschiedenen Kantengewichte

Zur Berechnung der mittleren Knotenanzahl bei unterschiedlichen Kantengewichten innerhalb eines Clusters können Markov Ketten [12] verwendet werden. Eine Markov Kette besteht aus einer Menge von Zuständen wobei ein Zustand mit bestimmter Wahrscheinlichkeit in einen anderen Zustand überführt. Zur Verwendung vorhandener Theorien und Formel von Markov Ketten, muss ein *AGbMM* zunächst in die Darstellung



## Verteilung der mobilen Knoten

einer Markov Kette transformiert werden. In Abbildung 20 ist diese Transformation exemplarisch für eine *AGbMM* dargestellt. Hierbei wird jedes Element des *AGbMM* in einen Knoten der Markov Kette überführt und mit den Übergängen versehen. Anschließend erhält jeder Knoten der Markov Kette entsprechend des Elementes des *AGbMM* eine Verweildauer. Daraus kann dann die durchschnittliche Verteilung auf die einzelnen Knoten der Markov Kette berechnet und diese Verteilung in umgekehrter Richtung auf die Elemente des *AGbMM* übertragen werden.

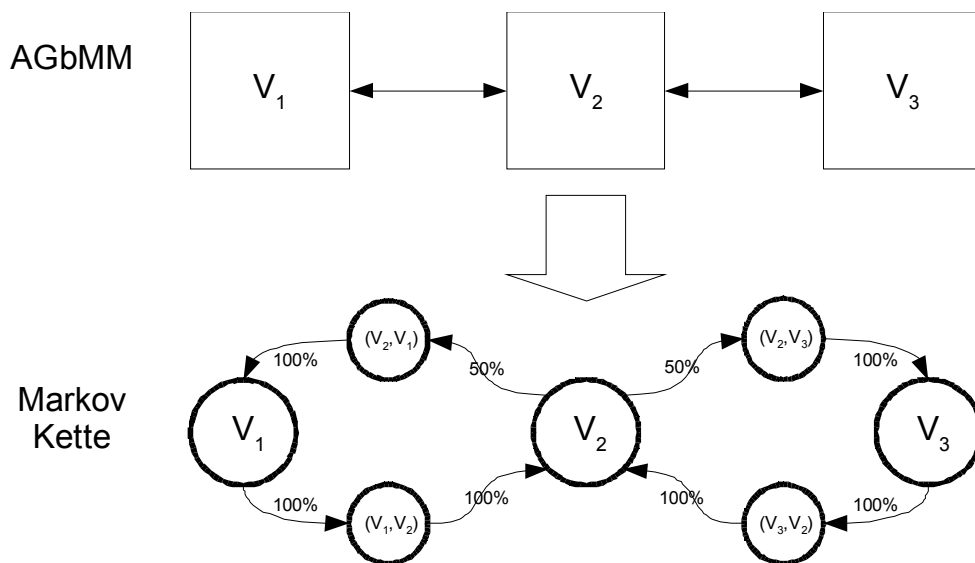


Abbildung 20: Transformation eines *AGbMM* in eine Markov Kette

### 5.3 Geschlossene Formel zur Vorhersage der Knotenverteilung

In den vorherigen zwei Abschnitten wurden die Auswirkungen verschiedener Eigenschaften des *AGbMM* auf die Verteilung der Knoten untersucht. Diese Eigenschaften wurden in zwei Gruppen eingeteilt: Eigenschaften, die Einfluss auf die Verweildauer der Knoten innerhalb der einzelnen Elemente des Modells haben und Eigenschaften, die Auswirkung auf den Verbindungsgrad haben. In diesem Abschnitt werden die Ergebnisse der vorherigen Abschnitte zusammengeführt, um daraus eine geschlossene Formel zur Vorhersage der Knotenverteilung in einem *AGbMM* herzuleiten.

#### Herleitung einer geschlossenen Formel

In Abschnitt 5.1 wurde gezeigt, dass die durchschnittliche Knotenanzahl für ein Element eines *AGbMM* proportional zur mittleren Verweildauer eines Knotens innerhalb des Elementes ist. Für das Modell aus Abbildung 14 berechnete sich die durchschnittliche Knotenanzahl  $n$  für eine  $a \in A$  wie folgt:

$$(1) \quad n(a) = N \frac{t(a)}{\sum_{x \in A} t(x)}$$

Hierbei ist  $t$  die durchschnittliche Verweildauer in einem Element und  $N$  die gesamte Knotenanzahl im *AGbMM*. Dass die Anzahl der Eingänge eines Elementes Auswirkung auf die mittlere Knotenanzahl hat, wurde in Abschnitt 5.2 gezeigt. Demnach gilt für zwei Elemente  $a, b \in A$  mit gleicher Verweildauer (z.B. zwei Cluster mit identischer Fläche und gleichen internen Parametern):

$$(2) \quad \frac{n(a)}{e(a)} = \frac{n(b)}{e(b)}$$

Wobei  $e(a)$  für ein  $a \in A$  die Anzahl der Eingänge des Elementes  $a$  angibt. Diese Formel gilt aber nur unter den Voraussetzungen, dass alle ausgehenden Kanten eines Clusters gleich wahrscheinlich sind, der Graph zusammenhängend ist und alle Kanten in beide Richtungen verlaufen. Eine allgemeine, geschlossene Formel zur Vorhersage der Knotenverteilung muss demzufolge die Bedingungen aus Abschnitt 5.1 und 5.2 berücksichtigen.

Die Berechnung der Knotenanzahl muss demnach sowohl proportional zur durchschnittlichen Verweildauer sein, als auch entsprechend der Anzahl der Eingänge gewichtet werden. Hierzu sei  $f(a) = e(a) \cdot t(a)$  der Einflussfaktor eines Elementes  $a \in A$ . Das heißt,  $f(a)$  ist das Produkt aus der Anzahl der Eingänge und der Verweildauer eines Elementes  $a$ . Außerdem sei  $F = \sum_{x \in A} f(x)$  die Summe der Einflussfaktoren aller Elemente eines *AGbMM*. Angenommen, dass das Verhältnis von mittlerer Knotenanzahl  $n(a)$  eines Elementes  $a$  zur Gesamtzahl der Knoten  $N$  gleich dem Verhältnis von Einflussfaktor  $f(a)$  des Elementes  $a$  zur Summe aller Einflussfaktoren  $F$  ist, so lässt sich die durchschnittliche Knotenanzahl  $n(a)$  für ein  $a \in A$  wie folgt berechnen:

$$(3) \quad n(a) = N \frac{f(a)}{F} = N \frac{f(a)}{\sum_{x \in A} f(x)}$$

Zunächst soll gezeigt werden, dass diese Formel die Bedingungen aus Abschnitt 5.1 erfüllt. Das bedeutet, in einem Modell in dem während eines Rundganges alle Elemente genau einmal betreten werden, ist die Knotenanzahl  $n(a)$  eines Elementes  $a$  proportional zur Verweildauer  $t(a)$  des Elementes  $a$ . Dazu sei ein Modell wie in Abbildung 14 gegeben und für zwei Cluster  $V_i, V_j$  gelte  $t(V_i) = k \cdot t(V_j)$ . Entsprechend der Ergebnisse aus Abschnitt 5.1 muss nun  $n(V_i) = k \cdot n(V_j)$  gelten. Jedes Element muss genau einen Eingang haben, denn laut Voraussetzung wird während eines Rundganges jedes Element genau einmal betreten. Somit gilt  $e(a) = 1$  für alle Elemente  $a$  des Modells. Daraus folgt,

## Verteilung der mobilen Knoten

---

für alle Elemente  $a$  gilt  $f(a)=t(a)$  und demzufolge ergibt sich aus Formel (3) direkt Formel (1).

Analog lässt sich zeigen, dass Formel (3) den Bedingungen aus Abschnitt 5.2 genügt. Das heißt, bei Elementen mit gleicher Verweildauer ist das Verhältnis von mittlerer Anzahl Knoten zur Anzahl der Eingänge gleich. Dazu seien zwei Cluster  $V_i$  und  $V_j$  eines *AGbMM* mit gleichen Attributen gegeben. Daraus ergibt sich, dass  $t(V_i)=t(V_j)$  gilt. Es seien nun  $V_i$  und  $V_j$  mit  $e(V_i)=k \cdot e(V_j)$  gegeben. Entsprechend Formel (2) muss nun folgendes gelten:

$$\begin{aligned} \frac{n(V_i)}{k \cdot e(V_j)} &= \frac{n(V_j)}{e(V_j)} \\ \rightarrow n(V_i) &= \frac{n(V_j) \cdot k \cdot e(V_j)}{e(V_j)} \\ \rightarrow n(V_i) &= k \cdot n(V_j) \end{aligned}$$

Das bedeutet, die mittlere Knotenanzahl in  $V_i$  muss  $k$  mal so groß sein wie die Anzahl der Knoten in  $V_j$ . Das ergibt sich aber direkt aus den Voraussetzungen, wie folgende Umformungen zeigen.

$$\begin{aligned} &n(V_i) \\ &\stackrel{\text{Formel (3)}}{=} N \frac{f(V_i)}{T} \\ &= N \frac{t(V_i) \cdot e(V_i)}{T} \\ &\stackrel{e(V_i)=k \cdot e(V_j)}{=} N \frac{t(V_i) \cdot k \cdot e(V_j)}{T} \\ &\stackrel{t(V_i)=t(V_j)}{=} N \frac{t(V_j) \cdot k \cdot e(V_j)}{T} \\ &= k \cdot N \frac{t(V_j) \cdot e(V_j)}{T} \\ &\stackrel{\text{Formel (3)}}{=} k \cdot n(V_j) \end{aligned}$$

Formel (3) berücksichtigt demnach sowohl die Verweildauer als auch die Gewichtung entsprechend der Eingänge zur Berechnung der mittleren Knotenanzahl eines Elementes.

### Zusammenfassung

Für ein *AGbMM* mit dem Graphen  $G=(V,E)$  und folgenden Eigenschaften:

1.  $G$  ist zusammenhängend
2. Die Gewichtungen aller ausgehenden Kanten eines Clusters sind gleich
3. Wenn  $(a,b) \in E$ , gilt auch  $(b,a) \in E$ , für alle  $a,b \in V$

lässt sich die durchschnittliche Knotenanzahl  $n(a)$  für ein Element  $a \in A = V \cup E$ , bei insgesamt  $N$  mobilen Knoten im *AGbMM*, wie folgt berechnen.

$$n(a) = N \frac{t(a) \cdot e(a)}{\sum_{x \in A} t(x) \cdot e(x)}$$

Hierbei ist  $t(a)$  die mittlere Verweildauer im Element  $a$  und  $e(a)$  die Anzahl der Eingänge des Elementes  $a$ . Die mittlere Verweildauer  $t(a)$  berechnet sich folgendermaßen:

$$t(a) = \begin{cases} t(a) = \frac{t_{\min}(a) + t_{\max}(a)}{2} + \frac{2 \cdot d(a)}{v_{\min}(a) + v_{\max}(a)} & \text{wenn } a \in V \\ t(a) = \frac{2 \cdot l(a)}{(v_{\min}(a) + v_{\max}(a))} & \text{wenn } a \in E \end{cases}$$

Hierbei ist:

- $t_{\min}(a)$  bzw.  $t_{\max}(a)$  die minimale bzw. maximale Aufenthaltsdauer im Element  $a$
- $v_{\min}(a)$  bzw.  $v_{\max}(a)$  die minimale bzw. maximale Geschwindigkeit innerhalb des Elementes  $a$
- $l(a)$  die Länge der Kante  $a$
- $d(a)$  durchschnittliche Länge des Weges beim Verlassen des Clusters  $a$

Die Werte  $t_{\min}$ ,  $t_{\max}$ ,  $v_{\min}$  und  $v_{\max}$  sind direkt im *AGbMM* angegeben. Auch die Länge  $l(e)$  einer Kante  $e$  kann exakt bestimmt werden. Einzig die durchschnittliche Länge des Weges  $d$  beim Verlassen des Clusters wurde im Verlauf der Arbeit nicht exakt berechnet. Im Abschnitt 5.1 wurde eine Approximation  $\tilde{d}$  für  $d$  vorgestellt, welche für die meisten praktischen Anwendungen präzise Vorhersagen der mittleren Knotenanzahl ermöglicht. Für ein Cluster  $v$  berechnet sich  $\tilde{d}(v)$  wie folgt:

$$\tilde{d}(v) = \sum_{e \in E_v} (w(e) \cdot \sqrt{|v_x - e_x|^2 + |v_y - e_y|^2})$$

Hierbei ist  $E_v$  die Menge der ausgehenden Kanten von Cluster  $v$ ,  $w(e)$  ist die Gewichtung der Kante  $e$ ,  $v_x$  und  $v_y$  sind die x und y Koordinaten des Mittelpunktes von Cluster  $v$  und  $e_x$  und  $e_y$  die x und y Koordinaten des Startpunktes der Kante  $e$ . Durch die 2. Voraussetzung für den Graphen des *AGbMM* (alle ausgehenden Kanten eines Clusters haben die gleiche Gewichtung) und weil die Summe der Gewichtungen aller ausgehenden Kanten gleich 1 ist, kann  $\tilde{d}(v)$  auch entsprechend berechnet werden:

$$\tilde{d}(v) = \frac{\sum_{e \in E_v} \sqrt{|v_x - e_x|^2 + |v_y - e_y|^2}}{|E_v|}$$

## 5.4 Beispiele zur Berechnung der mittleren Knotenanzahl

Zum Abschluss des ersten theoretischen Kapitels, zur Analyse der Knotenverteilung innerhalb eines *AGbMM*, werden die Ergebnisse einiger ausgewählter Szenarien vorgestellt. Alle dargestellten Modelle erfüllen die in diesem Abschnitt vorgestellten Kriterien zur Berechnung der Knotenanzahl. In den Clustern der abgebildeten Modelle sind der Name, die Dimension, die minimale und maximale Aufenthaltsdauer, die mittlere simulierte und die mittlere berechnete Knotenanzahl des entsprechenden Clusters dargestellt. An den Kanten des Modells sind die Länge der Kante und die mittlere simulierte und berechnete Knotenanzahl angegeben.

Die simulierten Werte entsprechen jeweils der mittleren Knotenanzahl eines Elementes des *AGbMM* gemessen über 24 Stunden mit einer Vorlaufzeit von 500 Sekunden. Bei den Kanten des Modells sind zur besseren Übersicht die Werte kongruenter Kanten zusammengefasst. Das heißt, die Zahlenwerte an den Kanten eines Modells entsprechen der Summe der Messwerte der dazugehörigen Kanten. Zusätzlich ist in den Clustern die Verteilung der mobilen Knoten abgebildet. Dabei kennzeichnen dunklere Bereiche eine höhere Knotendichte und vice versa. Zur Verdeutlichung der Abweichung zwischen den Ergebnissen der Simulation und der Berechnung ist unter jedem Szenario ein Diagramm abgebildet. Es zeigt den Betrag der Abweichung der berechneten Werte zu den simulierten Werten an.

Da es sich bei allen Modellen um geschlossen Systeme handelt, wird in diesem Abschnitt immer die prozentuale Abweichung bezogen auf die Gesamtzahl der mobilen Knoten betrachtet. Das hat den Vorteil, dass bei Elementen mit einer geringen Knotenanzahl kleinere Abweichungen nicht große prozentuale Abweichungen zur Folge haben. Bei allen Simulationen wird zusätzlich mit einer Gesamtzahl von 100 mobilen Knoten gearbeitet,

## Verteilung der mobilen Knoten

somit entspricht der absolute Wert der Abweichung, automatisch der prozentualen Abweichung bezogen auf die Gesamtzahl mobiler Knoten.

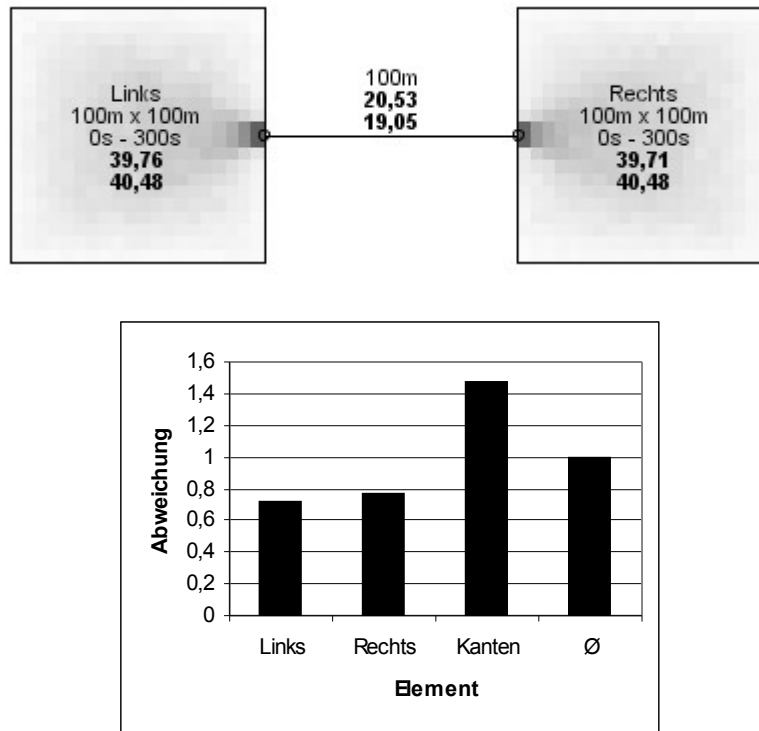


Abbildung 21: 1. Beispiel zur Berechnung der mittleren Knotenanzahl

Das erste Beispiel in Abbildung 21, ist das einfache *AGbMM* aus Abschnitt 5.1. Hierbei haben beide Cluster des Modells die gleiche Aufenthaltsdauer. Aus diesem Grund ist die mittlere Anzahl mobiler Knoten in den beiden Clustern sowohl bei der Berechnung als auch bei der Simulation nahezu identisch. Bei der Simulation unterscheiden sich die Durchschnittswerte um weniger als 0,1%.

Die Abweichung zwischen berechneter und simulierter Knotenanzahl beträgt in beiden Clustern weniger als 0,8. Weil jeweils die berechneten Werte größer sind als die der Simulation, werden insgesamt in den Clustern etwa 1,5 Knoten zu viel berechnet. Da das Modell aber nur aus zwei Clustern und zwei Kanten besteht, müssen diese 1,5 Knoten bei den Kanten zu wenig berechnet worden sein. Das heißt, die Abweichung von Simulation und Berechnung auf den Kanten beträgt rund 1,5%. Insgesamt ergibt sich damit eine durchschnittliche Abweichung zwischen Berechnung und Simulation von etwa 1%, bezogen auf die Gesamtzahl mobiler Knoten.

## Verteilung der mobilen Knoten

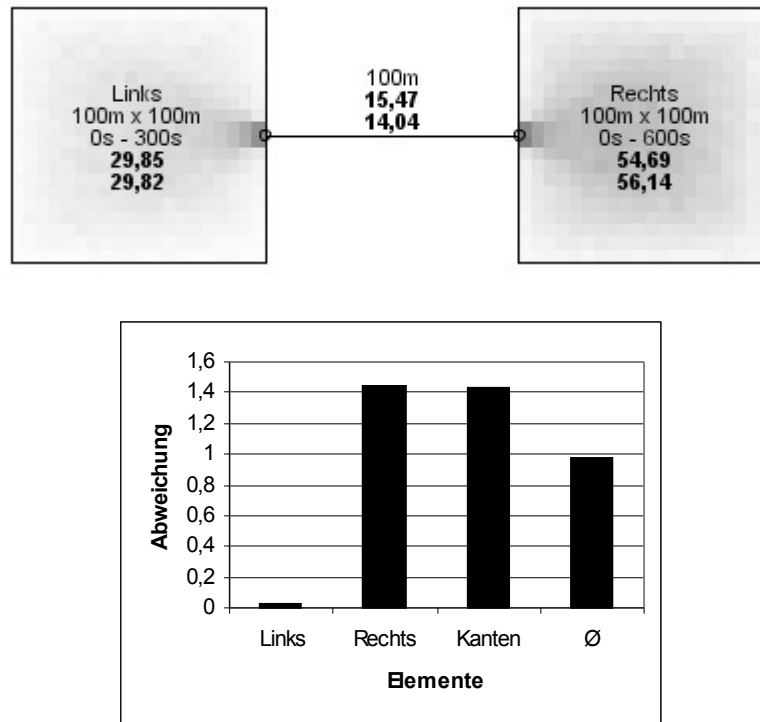


Abbildung 22: 2. Beispiel zur Berechnung der mittleren Knotenanzahl

Als zweites Beispiel ist in Abbildung 22 wiederum das einfache Modell aus Abschnitt 5.1 dargestellt. In diesem Fall sind aber die Aufenthaltsdauern der einzelnen Cluster verschieden. Die Aufenthaltsdauer im linken Cluster ist aus dem Intervall von 0 bis 300 Sekunden und im rechten aus dem Intervall von 0 bis 600 Sekunden. Das bedeutet, die mittlere Aufenthaltsdauer im rechten Cluster ist doppelt so groß wie die im linken. Wie bereits im Abschnitt 5.1 erläutert, ist damit nicht die mittlere Verweildauer im rechten Cluster doppelt so groß wie die im linken. Und somit auch nicht die durchschnittliche Knotenanzahl. Denn die mittlere Verweildauer berechnet sich aus mittlerer Aufenthaltsdauer plus durchschnittlicher Zeit zum Verlassen eines Clusters. Letzteres ist in beiden Clustern annähernd identisch.

Die Berechnung der Knotenanzahl für das linke Cluster entspricht annähernd dem Wert der Simulation. Beim rechten Cluster und somit auch bei den Kanten beträgt die Abweichung der Knotenanzahl rund 1,5. Insgesamt ergibt sich daraus eine mittlere Abweichung zwischen Simulation und Berechnung von rund 1%, bezogen auf die Gesamtzahl mobiler Knoten.

## Verteilung der mobilen Knoten

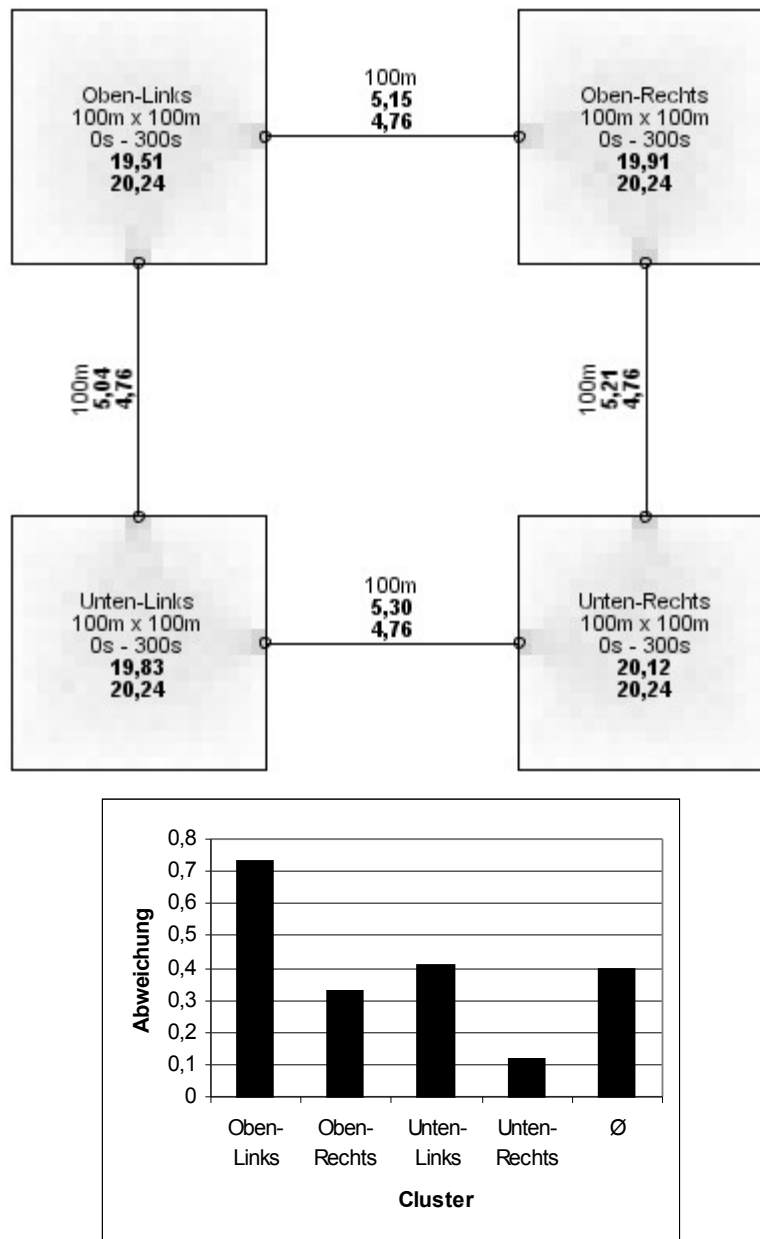


Abbildung 23: 3. Beispiel zur Berechnung der mittleren Knotenanzahl

Das dritte Beispiel ist ein Modell bei dem vier Cluster in Form eines Kreises angeordnet sind (Abbildung 23). Alle Cluster und Kanten haben jeweils annähernd die gleiche Anzahl Knoten. Die Abweichung zwischen Berechnung und Simulation beträgt maximal 0,73 im linken oberen Cluster. Insgesamt beträgt die mittlere Abweichung aller Elemente weniger als 0,5%, bezogen auf die Gesamtzahl mobiler Knoten im Modell.



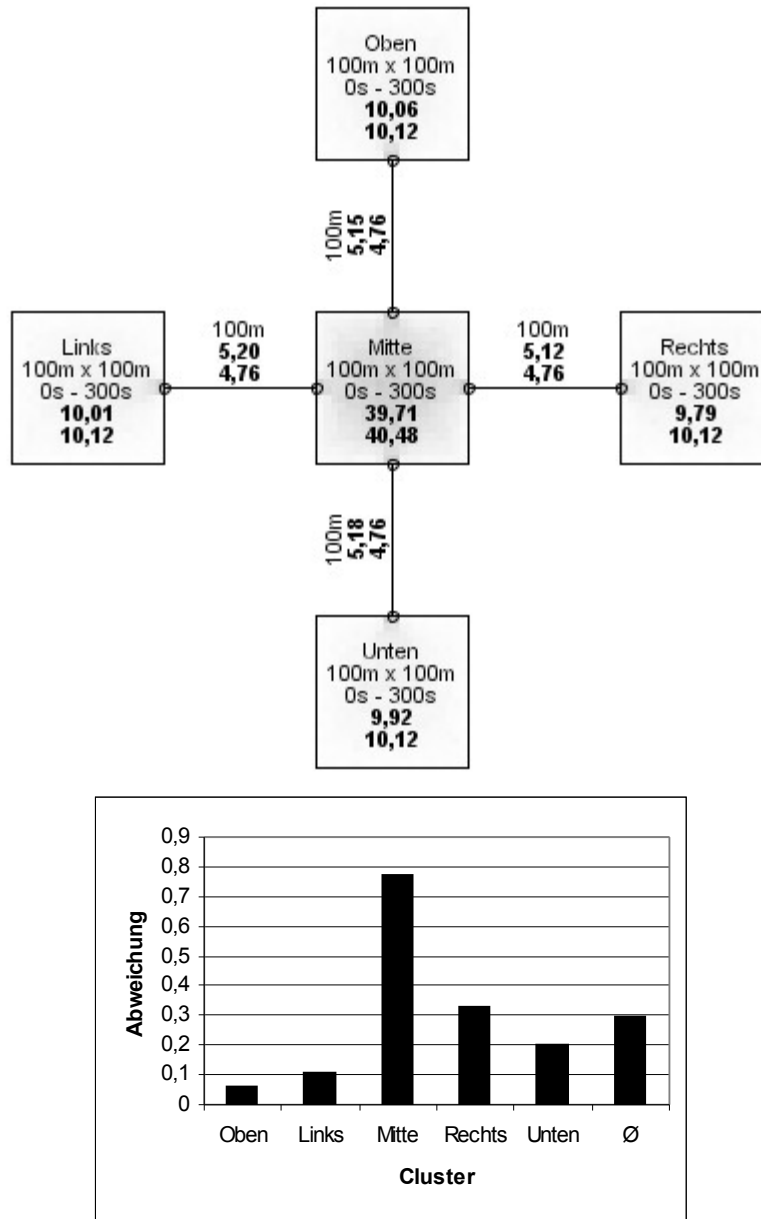


Abbildung 24: 4. Beispiel zur Berechnung der mittleren Knotenanzahl

Als 4. Beispiel sind in Abbildung 24 fünf Cluster in Form eines Kreuzes angeordnet. Das mittlere Cluster hat durchschnittlich ein viermal so große Knotenanzahl wie die anderen Cluster, weil es über vier und alle anderen nur über einen Eingang verfügen. Bei diesem Cluster ist mit etwa 0,8 die Abweichung von Simulation und Berechnung am größten. Im Mittel beträgt die Abweichung zwischen berechneter und simulierter Knotenanzahl 0,3%, der Gesamtzahl mobiler Knoten.

## Verteilung der mobilen Knoten

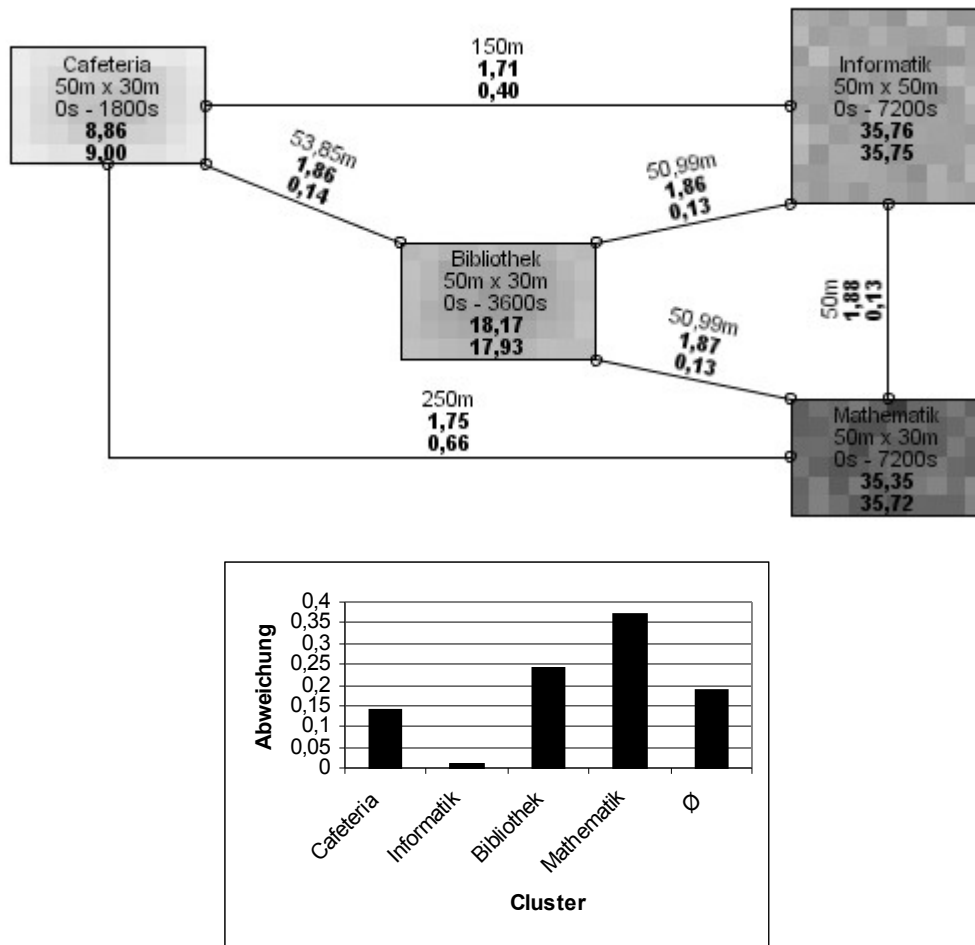


Abbildung 25: 5. Beispiel zur Berechnung der mittleren Knotenanzahl

Das letzte Beispiel zur Berechnung der mittleren Knotenanzahl in Abbildung 25 ist das Modell eines Fachbereiches. Man erkennt, dass sich aufgrund der höheren Aufenthaltsdauern mehr Teilnehmer im Informatik- und Mathematikgebäude aufhalten als in der Bibliothek oder der Cafeteria. Des Weiteren wurden innerhalb des Informatik- und Mathematikgebäudes verhältnismäßig hohe Wartezeiten zwischen den Bewegungen gewählt, infolgedessen gibt es kaum eine Konzentration der Knotendichte im Zentrum dieser Gebäude.

Aufgrund der äußerst geringen Dichten auf den Kanten des Graphen und einer relativ langen Simulationszeit sind die Knotenanzahlen auf den Kanten des Graphen kaum repräsentativ, da hier mit aller Wahrscheinlichkeit Rundungsfehler bei der Berechnung aufgetreten sind. Bei den Clustern beträgt die maximale Abweichung rund 0,37 und die mittlere Abweichung aller Cluster etwa 0,2%, von der Gesamtzahl mobiler Knoten.

## Verteilung der mobilen Knoten

---

Insgesamt kann für die durchgeführten Berechnungen festgestellt werden, dass die Abweichung zu den simulierten Werten in keinem Szenario und keinem Element größer als 2%, bezogen auf die Gesamtzahl mobiler Knoten, war. Das heißt, bei einem Modell bei dem die Bewegung von 100 Knoten simuliert wird, weicht die Berechnung nie um mehr als zwei Knoten von dem exakten Wert ab.

Darüber hinaus ist festzuhalten, dass die Berechnung bei komplexeren Szenarien (Modelle mit mehreren Clustern und Kanten) genauer sind als bei einfachen Modellen. Daraus folgt, dass sich bei mehreren Elementen die Ungenauigkeiten auf die einzelnen Elemente verteilt und somit weniger signifikant sind.

Obwohl die vorgestellten Berechnungen die mittlere Knotenanzahl relativ genau vorhersagen, muss beachtet werden, dass diese einen Erwartungswert repräsentieren. Das bedeutet, es wird ein Wert berechnet, der im Mittel nach einer unendlichen Simulationszeit erreicht wird. Demzufolge kann bei einer kurzen Simulationszeit die gemessene, mittlere Knotenanzahl deutlich von dem berechneten Wert abweichen.

## 6 Vorhersage der Nachrichtenverteilung

In diesem Kapitel wird die Nachrichtenverteilung innerhalb eines *Area Graph-based Mobility Models (AGbMM)* detaillierter untersucht. Ziel des Kapitels ist es, für ein gegebenes *AGbMM* und eine vorgegebene Startposition einer Nachricht, die mittlere Zeit zur Verteilung der Nachricht an die mobilen Knoten abschätzen zu können. Das bedeutet, nach welcher Zeit haben wie viele mobile Knoten die Nachricht erhalten. Hierzu wird untersucht, wie sich eine Nachricht innerhalb des *AGbMM* ausbreitet und dabei die einzelnen mobilen Knoten die Nachricht erhalten. Mit Hilfe dieser Untersuchungen kann dann beispielsweise die mittlere Zeit zur Verteilung eines Software-Updates innerhalb eines MANETs abgeschätzt werden.

Als Protokoll zur Verteilung der Nachricht wird einfaches Fluten verwendet. Das bedeutet, zu Beginn verfügt lediglich ein ausgewählter Knoten über die zu verteilende Nachricht. Dieser Knoten versendet die Nachricht an alle mobilen Knoten in seiner Funkreichweite. Alle Knoten die die Nachricht erhalten haben warten eine zufällige Zeit (Sendeverzögerung) und versenden anschließend die Nachricht selbst an alle Knoten in ihrer Funkreichweite. Die Sendeverzögerung wird dabei durch eine minimale und maximaler Zeit definiert. Aus diesem Intervall wird gleichverteilt eine Zeit gewählt, die der entsprechende Knoten wartet, bevor er die Nachricht weiter versendet. Diese Verzögerung der Nachrichtenverteilung repräsentiert die interne Verarbeitungszeit der Nachricht innerhalb des mobilen Knoten. Empfängt ein Knoten eine Nachricht, die er bereits erhalten hat, versendet er diese Nachricht nicht weiter. Das bedeutet, jeder Knoten versendet jede Nachricht höchstens ein Mal. Daraus folgt, wenn eine festgelegte Nachricht von jedem mobilen Knoten genau einmal versendet wurde, müssen auch alle Knoten diese Nachricht erhalten haben.

Zunächst wird im Abschnitt 6.1 analysiert, wie sich eine Nachricht im *Random Waypoint Mobility Model* ausbreitet. Im Anschluss in Abschnitt 6.2 wird die Nachrichtenverteilung auf einer Geraden untersucht. Im Abschnitt 6.3 werden dann die Ergebnisse der vorherigen Abschnitte zur Berechnung der Nachrichtenverteilung innerhalb eines *AGbMM* verwendet. In allen Abschnitten zur Berechnung der Nachrichtenverteilung wird davon ausgegangen, dass die mittlere Knotenanzahl der einzelnen Elemente bekannt ist. Das bedeutet, wenn die Verteilung einer Nachricht in einem *AGbMM* untersucht wird, ist für alle Elemente des Modells die mittlere Knotenanzahl bekannt. Für eine Vielzahl von *AGbMM* kann zur Berechnung die Formel aus Kapitel 5 verwendet werden. Für komplexere *AGbMM*, bei denen die Berechnung der Knotenanzahl zu aufwendig ist oder aufgrund verschiedener Kantengewichte nicht durchführbar ist, kann zur Bestimmung der mittleren Knotenanzahl der entworfene Simulator aus Abschnitt 4.2 verwendet werden.

### System Modell

Für die Analyse der Nachrichtenverteilung wird in diesem Kapitel das Basismodell des *AGbMM* ohne Erweiterungen aus Abschnitt 3.1 verwendet. Das heißt, es werden keine rekursiven Aufrufe mehrerer *AGbMM* untersucht. Des Weiteren sind alle Berechnungen in diesem Kapitel nur für Modelle anwendbar, bei denen der Graph des *AGbMM* zusammenhängend ist, d.h. alle Cluster und Kanten von jedem Element des Modells erreichbar sind. Zusätzlich wird bei den Berechnungen der Nachrichtenverteilung angenommen, dass immer 100% der mobilen Knoten die Nachricht erhalten.

Ein denkbare Szenario sind Studenten auf einem Campus welche untereinander Informationen austauschen. Jeder Teilnehmer verfügt dazu über ein mobiles Endgerät, etwa ein Handy, PDA oder Laptop. Diese Geräte sind in der Lage untereinander Verbindungen aufzubauen und somit Informationen auszutauschen. Dabei ist jeder Teilnehmer in der Lage Nachrichten zu verschicken und zu empfangen. Des Weiteren wird angenommen, dass alle Teilnehmer an den verschickten Informationen interessiert sind.

Im Rahmen dieser Arbeit wird zur Nachrichtenverteilung WLAN (IEEE 802.11) als Technik verwendet. Es wird angenommen, dass alle Modelle über eine feste Anzahl von Teilnehmer verfügen. Das bedeutet, während der Simulation tritt kein Teilnehmer dem Modell bei oder verlässt selbiges. Bei der Simulation der Nachrichtenverteilung von mobilen Endgeräten, wird zusätzlich angenommen, dass alle Endgeräte über genügend Speicher- und Energiereserven verfügen, so dass es zu keinerlei Ausfällen kommt.

## 6.1 Datenverteilung im Random Waypoint Mobility Model

Dieser Abschnitt stellt Berechnungen zur Vorhersage der Nachrichtenverteilung innerhalb eines *Random Waypoint Mobility Models* vor. Dabei wird als Verteilungsprotokoll einfaches Fluten mit einer Sendeverzögerung von 0,1 bis 0,3 Sekunden verwendet. Das bedeutet, bevor ein mobiler Knoten eine neu erhaltene Nachricht weiter versendet, wartet dieser mindestens 0,1 Sekunden und maximal 0,3 Sekunden. Dabei wird die Verzögerung gleichverteilt aus dem Intervall gewählt, so dass ein mobiler Knoten vor dem Weitersenden im Mittel 0,2 Sekunden wartet.

Erste Untersuchungen zur Nachrichtenverteilung im *Random Waypoint Mobility Model* werden in [13] vorgestellt. Es werden Ansätze aus der Epidemiologie auf MANETs übertragen und dabei gezeigt, dass sich Informationen bzw. Nachrichten in MANETs in gleicher Weise ausbreiten wie Epidemien innerhalb einer Population.

Als Einführung in diese Thematik soll hier ein einfaches Beispiel aus dem Bereich der Epidemiologie vorgestellt werden. Dazu sei eine Stadt mit 5 Millionen Einwohner gegeben ( $N=5.000.000$ ). Zum Zeitpunkt Null sei in dieser Stadt eine Person mit einer bestimmten Krankheit infiziert. Des Weiteren sei die Wahrscheinlichkeit, dass ein Infizierter einen nicht

## Vorhersage der Nachrichtenverteilung

---

Infizierten mit dieser Krankheit ansteckt, Eins zu zwei Hundert ( $\beta=1/200$ ). Zusätzlich sei die mittlere Anzahl von Kontakten zu anderen Personen innerhalb eines Tages rund 50 pro Individuum ( $x=50$ ). Anhand dieser Werte lässt sich die Infektionsrate  $a$  der Krankheit wie folgt berechnen.

$$a = \frac{\beta \cdot x}{N} = \frac{50}{200 \cdot 5.000.000} = \frac{1}{20.000.000} = 0,00000005$$

Mit Hilfe der Infektionsrate  $a$  kann nun der Anteil infizierter Personen  $i(t)$  zum Zeitpunkt  $t$  berechnet werden.

$$i(t) = \frac{1}{1 + (N - 1) \cdot e^{-a \cdot N \cdot t}}$$

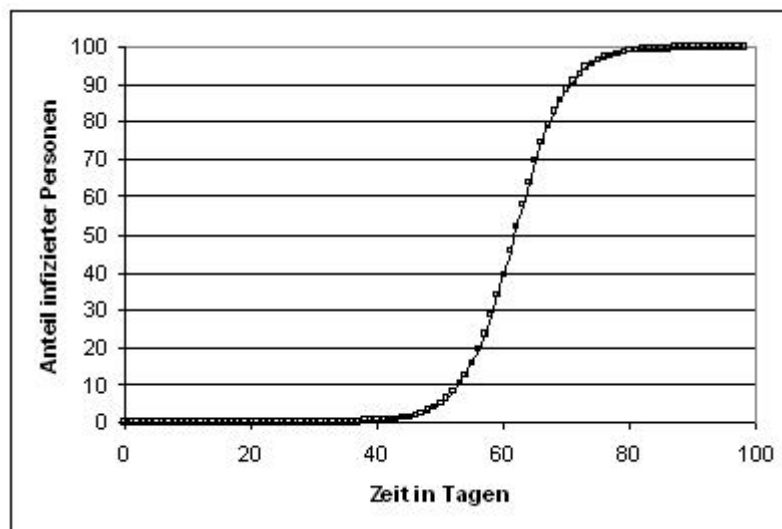


Abbildung 26: Epidemische Verteilung einer Krankheit mit einer Infektionsrate von

In Abbildung 26 ist der prozentuale Anteil infizierten Personen  $i(t)$  des zuvor beschriebenen Szenarios dargestellt. Es ist zu erkennen, dass sich zu Beginn nur wenige Personen mit der Krankheit infizieren. Nach circa 40 Tagen steigt die Zahl neuer Infizierungen drastisch an und erreicht das Maximum nach rund 60 Tagen. Nach etwa 90 Tagen ist dann nahezu die gesamte Bevölkerung der Stadt mit der Krankheit infiziert.

### Erste Ansätze zur Nachrichtenverteilung

In [13] werden die zuvor dargestellten Berechnungen aus dem Bereich der Epidemiologie direkt auf die Nachrichtenverteilung übertragen. Der Anteil erreichter Knoten  $i(t)$  zum Zeitpunkt  $t$  wird analog zur Ausbreitung von Krankheiten oder Seuchen durch folgende Formel berechnet.

## Vorhersage der Nachrichtenverteilung

---

$$i(t) = \frac{I(t)}{N} = \frac{1}{1 + (N-1) \cdot e^{-a \cdot N \cdot t}} \quad (1)$$

Hierbei ist  $I(t)$  die Anzahl der Knoten, die zum Zeitpunkt  $t$  die Nachricht bereits erhalten haben,  $N$  die Gesamtzahl mobiler Knoten im Modell und  $a$  die bereits definierte Infektionsrate. Die Infektionsrate beschreibt dabei den Fortschritt der Nachrichtenverteilung und hängt von verschiedenen Parametern des Bewegungsmodells und des Kommunikationsmodells ab. Parameter des Kommunikationsmodells sind z.B. die Funkreichweite oder die Sendeverzögerung. Wie bereits gezeigt, wird die Infektionsrate  $a$  in [13] folgendermaßen berechnet.

$$a = \frac{\beta \cdot x}{N}$$

Dabei ist  $\beta$  die Wahrscheinlichkeit einer Nachrichtenübermittlung zwischen einem infizierten und nicht infizierten Knoten und  $x$  die Anzahl der Kontakte eines mobilen Knotens pro Zeiteinheit. In [13] wird der Wert  $a$  nicht analytisch bestimmt, vielmehr wird mit Hilfe vieler Simulationen die Infektionsrate interpoliert. Es lässt sich aber schlussfolgern, dass  $x$  durch die Parameter des Bewegungsmodells und  $\beta$  durch die Werte des Kommunikationsmodells bestimmt wird.

Ein erster Ansatz zur Bestimmung der Parameter  $\beta$  und  $x$  ist Folgender:

- $\beta=1$ , weil beim einfachen Fluten die Nachricht immer weitergeleitet wird.
- $x = \frac{\text{Knoten in Funkreichweite}}{\text{Knoten insgesamt}}$

Simulationen verschiedener Szenarien zeigen jedoch, dass die berechneten Verteilungen schneller verlaufen als die durchgeführten Simulationen. Ein Beispiel ist in Abbildung 27 dargestellt. Die Abbildung zeigt die Berechnung und die Simulation der Nachrichtenverteilung innerhalb eines 100 mal 100 Meter großen Modells mit einer Gesamtzahl von 100 mobilen Knoten. Die Grafik zeigt, laut Berechnung sollten 50% der Knoten die Nachricht nach ungefähr 0,15 Sekunden erhalten haben. Entsprechend der Simulation haben aber erst nach circa 0,27 Sekunden 50% der Knoten die Nachricht erhalten. Demzufolge wird eine zu große Infektionsrate bei der Rechnung verwendet, was darauf zurückzuführen ist, dass zu wenig Parameter berücksichtigt werden. So wird die mittlere Sendeverzögerung von 0,2 Sekunden in der Vorhersage gar nicht berücksichtigt. Was genau dazu führt, dass die Berechnung schneller verläuft als die eigentliche Simulation. Obwohl die Kurve der Berechnung und die der Simulation nicht kongruent sind, so ist der Verlauf der Kurven doch sehr ähnlich. Es muss demnach, zur genaueren Vorhersage der Nachrichtenverteilung, die Skalierung der Kurve der Berechnung an die der Simulation

## Vorhersage der Nachrichtenverteilung

angepasst werden. Das bedeutet, die Infektionsrate muss auf detailliertere Art und Weise berechnet werden.

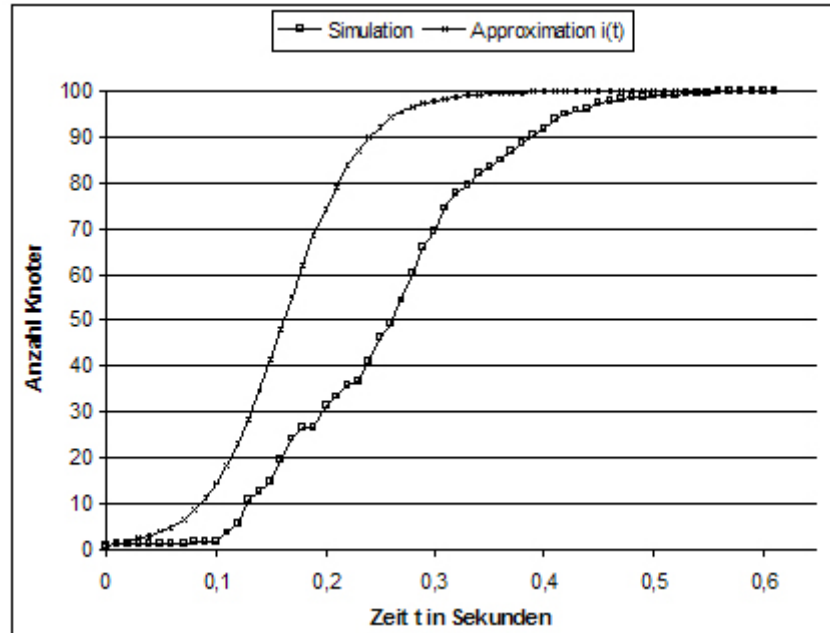


Abbildung 27: Erster Ansatz zur Berechnung der Nachrichtenverteilung innerhalb eines Random Waypoint Mobility Models

Entsprechend Formel (1) haben nach einer Zeit  $t$  genau  $N \cdot i(t)$  Knoten die Nachricht erhalten, wobei  $i(t)$  ein Wert aus dem Intervall  $]0,1[$  ist. Definiert man nun für einen Zeitpunkt  $t_\varepsilon$  den Wert  $\varepsilon$  durch  $i(t_\varepsilon)$ , so ergibt sich aus Formel (1) folgende Formel:

$$\varepsilon = \frac{1}{1 + (N-1) \cdot e^{-a \cdot N \cdot t_\varepsilon}}$$

Diese Formel lässt durch folgende Schritte nach  $a$  umformen



$$\begin{aligned}
 \varepsilon &= \frac{1}{1 + (N-1) \cdot e^{-a \cdot N \cdot t_\varepsilon}} \\
 \Rightarrow 1 &= \varepsilon \cdot (1 + (N-1) \cdot e^{-a \cdot N \cdot t_\varepsilon}) \\
 \Rightarrow \frac{1 - \varepsilon}{\varepsilon \cdot (N-1)} &= e^{-a \cdot N \cdot t_\varepsilon} \\
 \Rightarrow \ln \frac{1 - \varepsilon}{\varepsilon \cdot (N-1)} &= -a \cdot N \cdot t_\varepsilon \\
 \Rightarrow a &= \frac{-\ln \frac{1 - \varepsilon}{\varepsilon \cdot (N-1)}}{N \cdot t_\varepsilon} \quad (2)
 \end{aligned}$$

Geht man nun analog zu Formel (1) davon aus, dass die Infektionsrate während des gesamten Prozesses konstant ist, kann bei bekannten  $\varepsilon$  zu einen Zeitpunkt  $t_\varepsilon$  mit  $t_\varepsilon > 0$  die Infektionsrate  $a$  berechnet werden. Das bedeutet, ist zu einem bestimmten Zeitpunkt das Verhältnis erreichter Knoten zur Gesamtzahl Knoten bekannt, kann daraus die Infektionsrate des entsprechenden Modells berechnet werden.

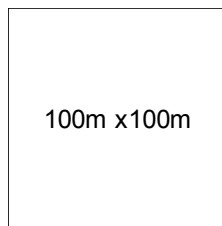


Abbildung 28: Random Waypoint Mobility Model

Betrachten wir dazu das *Random Waypoint Mobility Model* der Abbildung 28. Geht man davon aus, dass sich der Startknoten (der mobile Knoten der zu Beginn als einziger über die Nachricht verfügt) im Mittelpunkt des Quadrates befindet, so kann kein mobiler Knoten mehr als 71m von diesem Knoten entfernt sein. Denn, die Positionen mit maximaler Entfernung zum Mittelpunkt sind jeweils die 4 Ecken des Quadrates und diese sind, entsprechend des Satzes von Pythagoras, jeweils circa 70,71m vom Zentrum entfernt. Das bedeutet, wenn eine Nachricht, ausgehend vom Mittelpunkt des Quadrates, 70,71m in alle Richtungen versendet wurde, haben alle mobilen Knoten innerhalb des Quadrates die Nachricht erhalten. Analog gilt für eine Nachricht, die ausgehend von der linken unteren Ecke des Modells versendet wird, eine maximale Entfernung von circa 141,42m. Demnach gilt: Ausgehend von der Startposition einer Nachricht existiert eine maximale Entfernung, nach welcher alle Knoten innerhalb des Modells die Nachricht empfangen haben.

## Vorhersage der Nachrichtenverteilung

---

Wenn alle Knoten eines Modells die Nachricht erhalten haben, muss demzufolge in Formel (2) der Wert  $\varepsilon$  gleich Eins sein. Weil aber der natürliche Logarithmus von Null nicht definiert ist, darf für  $\varepsilon$  in Formel (2) nicht der Wert Eins verwendet werden. Entsprechend Formel (1) ist  $i(t)$  aus dem Intervall  $]0,1[$  und erreicht theoretisch nie den Wert Eins. Somit darf auch  $\varepsilon$  nicht den Wert Eins besitzen. Nimmt man nun an, dass nach dem Versenden einer Nachricht über die entsprechende Entfernung nicht alle Knoten des Modells, sondern nur 99% Prozent ( $\varepsilon=0,99$ ) die Nachricht erhalten haben, so lässt sich für dieses  $\varepsilon$  die Infektionsrate  $a$  berechnen.

Demnach ergibt sich aus der Startposition einer Nachricht eine Strecke der Länge  $\delta$  für die gilt: Wenn ausgehend von der Startposition die Nachricht in alle Richtung über die Entfernung  $\delta$  versendet wurde, haben etwa 99% Prozent der Knoten die Nachricht empfangen.

Lässt sich nun aus der Länge  $\delta$  die durchschnittliche Sendezeit  $t_\varepsilon$ , einer Nachricht über die Entfernung  $\delta$  berechnen, kann mittels  $\varepsilon$  und  $t_\varepsilon$ , die Infektionsrate  $a$  bestimmt werden. Es wird also versucht, mittels verschiedener Annahmen und Voraussetzungen die Kurve aus Formel (1) entsprechend der Parameter des Modells zu skalieren. Dazu wird ein Zeitpunkt  $t_\varepsilon$  benötigt, an dem die Funktion  $i(t_\varepsilon)$  den Wert  $\varepsilon$  annimmt. Weil  $i(t_\varepsilon)$  für  $t_\varepsilon > 0$  aber eine monoton steigende Funktion mit Werten aus dem Intervall  $]0,1[$  ist, existiert für jeden Wert  $\varepsilon$  aus dem Intervall  $]0,1[$  genau ein  $t_\varepsilon$ . Diese Zeit  $t_\varepsilon$  entspricht dabei der Sendezeit über die längste mögliche Strecke innerhalb des Modells, ausgehend von der Startposition der Nachricht. Demzufolge lässt sich die Infektionsrate eines *Random Waypoint Mobility Models* approximieren, wenn für eine gegebene Strecke der Länge  $\delta$  die Sendezeit für diese Strecke innerhalb des Modells berechnet werden kann.

Kennt man die mittlere Entfernung  $S$  zum äußersten Knoten innerhalb des Funkradius, so kann die mittleren Sendezeit  $s(\delta)$  für eine Strecke der Länge  $\delta$  wie folgt approximiert werden:

$$s(\delta) = \frac{\gamma \cdot \delta}{S}$$

Hierbei ist:

- $\gamma$  der Erwartungswert der Sendeverzögerung
- $S$  die mittlere Sendereichweite einer Nachricht im Modell

Der Erwartungswert der Sendeverzögerung  $\gamma$  kann als Mittelwert aus minimaler und maximaler Sendeverzögerung berechnet werden, weil im Rahmen dieser Arbeit eine gleichverteilte Sendeverzögerung aus dem Intervall 0,1 bis 0,3 Sekunden verwendet wird.

## Vorhersage der Nachrichtenverteilung

---

Bei Verwendung einer anderen Verteilung der Sendeverzögerung, z.B. einer Normalverteilung, muss die Berechnung des Erwartungswertes entsprechend angepasst werden. Zur Berechnung der mittleren Sendereichweite  $S$  nehmen wir an, dass alle erreichbaren Nachbarn eines mobilen Knotens, innerhalb des Funkbereiches, gleichverteilt sind. Das heißt, hat ein mobiler Knoten genau einen Nachbarn innerhalb seines Funkradius

$R$ , so befindet sich dieser im Mittel in einer Entfernung von  $\frac{R}{\sqrt{2}}$ . Denn somit ist die Fläche des Kreises vom mobilen Knoten zu seinem Nachbarn genau halb so groß wie die gesamten Fläche des Funkbereiches (siehe Abbildung 29).

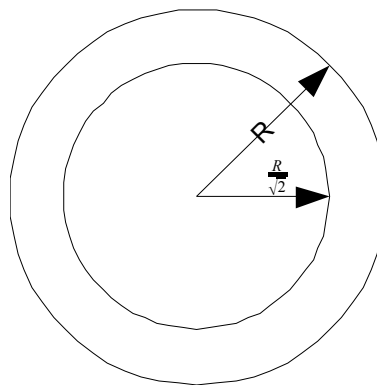


Abbildung 29: Mittlere Entfernung eines Nachbarn innerhalb des Funkbereiches eines mobilen Knotens

Analog lässt sich die mittlere Sendereichweite für  $k$  Nachbarn innerhalb des Funkradius eines mobilen Knotens bestimmen. Dazu wird ein Radius  $r$  gesucht, der folgende Gleichung erfüllt:

$$\frac{\pi \cdot R^2}{\pi \cdot r^2} = \frac{k+1}{k}$$

Das heißt, das Verhältnis der Kreisflächen von  $R$  und  $r$  muss gleich dem Verhältnis von  $k+1$  zu  $k$  sein. Stellt man sich vor, dass  $k$  Knoten innerhalb des Funkbereiches gleichmäßig verteilt werden, so teilen diese den Funkradius in  $k+1$  Kreisringe, welche im Mittel die gleiche Fläche haben sollten. Formt man die obige Gleichung nach  $r$  um, so ergibt sich bei

$k$  Nachbarn innerhalb des Funkbereiches ein Radius von  $r = R \cdot \frac{\sqrt{k}}{\sqrt{k+1}}$ . Dieser Wert

entspricht somit der mittleren Entfernung bis zum äußersten Nachbarn innerhalb des Funkbereiches eines mobilen Knotens, unter der Voraussetzung, dass die Knoten innerhalb des Funkbereiches gleichmäßig verteilt sind. Das heißt, bei durchschnittlich  $x$  Nachbarn innerhalb des Funkbereiches, bei einer Funkreichweite  $R$ , lässt sich die mittlere Sendereichweite  $S$  wie folgt berechnen:

## Vorhersage der Nachrichtenverteilung

---

$$S = R \cdot \frac{\sqrt{x}}{\sqrt{x+1}}$$

Demnach muss zur Bestimmung der mittleren Sendereichweite  $S$  die durchschnittliche Anzahl Nachbarn  $x$  berechnet werden. Dazu nehmen wir an, dass zum einen alle mobilen Knoten gleichverteilt innerhalb der Simulationsfläche sind und zum anderen der komplette Funkbereich eines Knoten im Inneren des Modells liegt. Entsprechend dieser Annahmen, ist das Verhältnis der durchschnittlichen Anzahl Nachbarn  $x$  zur Gesamtzahl mobiler Knoten im Modell  $N$  gleich dem Verhältnis der Fläche des Funkbereiches eines Knotens ( $R^2\pi$ ) zur Gesamtfläche des Modells (Breite mal Höhe des Simulationsbereiches). Somit lässt sich die mittlere Anzahl mobiler Knoten  $x$  innerhalb der Funkreichweite wie folgt bestimmen:

$$x = N \cdot \frac{R^2 \cdot \pi}{width \cdot height}$$

Mit *width* als Breite und *height* als Höhe des *Random Waypoint Mobility Models*. Demnach berechnet sich die gesamte Sendezeit  $s$  für eine Distanz  $\delta$  entsprechend:

$$s(\delta) = \frac{\gamma \cdot \delta}{S} = \frac{\gamma \cdot \delta}{R \cdot \frac{\sqrt{x}}{\sqrt{x+1}}} = \frac{\gamma \cdot \delta \cdot \sqrt{x+1}}{R \cdot \sqrt{x}} \quad \text{mit} \quad x = N \cdot \frac{R^2 \cdot \pi}{width \cdot height}$$

Entspricht  $\delta$  der Entfernung, nach welcher im Mittel  $N \cdot \varepsilon$  Prozent der Knoten die Nachricht erhalten haben, kann in Formel (2)  $t_\varepsilon$  durch  $s(\delta)$  ersetzt werden. Insgesamt lässt sich dann die Infektionsrate  $a$  für ein *Random Waypoint Mobility Model* folgendermaßen berechnen:

$$a = \frac{-\ln \frac{1-\varepsilon}{\varepsilon \cdot (N-1)} \cdot R \cdot \sqrt{x+1}}{N \cdot \gamma \cdot \delta \cdot \sqrt{x}} \quad \text{mit} \quad x = N \cdot \frac{R^2 \cdot \pi}{width \cdot height} \quad (3)$$

Dabei ist:

- $N$  die Gesamtzahl mobiler Knoten im Modell
- $R$  die Funkreichweite eines mobilen Knotens
- $\gamma$  die mittlere Sendeverzögerung
- $\delta$  die Entfernung über die die Nachricht versendet wird
- $\varepsilon$  der Anteil erreichter Knoten zur Gesamtzahl mobiler Knoten nach Versenden der Nachricht über eine Entfernung von  $\delta$

## Vorhersage der Nachrichtenverteilung

---

Mit diesem Wert für die Infektionsrate  $a$  kann nun die Verteilung für ein *Random Waypoint Mobility Model* mittels Formel (1) berechnet werden.

Als Beispiel hierzu sei das *Random Waypoint Mobility Model* der Abbildung 28 gegeben. Die Nachricht wird ausgehend von einem mobilen Knoten im Zentrum des Modells versendet. Innerhalb des Modells befinden sich 100 mobile Knoten, die Funkreichweite  $R$  beträgt 30 Meter und die mittlere Sendeverzögerung  $\gamma$  gleich 0,2 Sekunden. Mit Hilfe der Länge und Breite des Modells von jeweils 100m und der Funkreichweite  $R$  ergibt sich eine durchschnittliche Knotenanzahl innerhalb des Funkradius von

$$x = 100 \cdot \frac{R^2 \cdot \pi}{width \cdot height} = 100 \cdot \frac{900 m^2 \cdot \pi}{10000 m^2} = 9 \cdot \pi \approx 28,27$$

Erreicht die Nachricht eine Ecke des Quadrates, sollten im Mittel alle mobilen Knoten die Nachricht erhalten haben. Demnach ist die maximale Entfernung  $\delta$  über die die Nachricht versendet werden muss  $\sqrt{(50m)^2 + (50m)^2} = 70,71 m$ . Weil  $\varepsilon=1$  für die Berechnung der Infektionsrate nicht definiert ist, nehmen wir an, dass nach einer Entfernung von  $\delta$ , 99 Prozent ( $\varepsilon=0,99$ ) der mobilen Knoten die Nachricht erhalten haben. Daraus ergibt sich eine Infektionsrate von

$$\begin{aligned} a &= \frac{-\ln \frac{1-\varepsilon}{\varepsilon \cdot (N-1)} \cdot R \cdot \sqrt{x+1}}{N \cdot \gamma \cdot \delta \cdot \sqrt{x}} \approx \frac{-\ln \frac{1-0,99}{0,99 \cdot (100-1)} \cdot 30m \cdot \sqrt{28,27+1}}{100 \cdot 0,2s \cdot 70,71 m \cdot \sqrt{28,27}} \\ &\approx \frac{-\ln \frac{0,01}{98,01} \cdot 30m \cdot 5,41}{100 \cdot 0,2s \cdot 70,71 m \cdot 5,31} \approx 0,1986 \end{aligned}$$

Mit Hilfe der Infektionsrate  $a$  und Formel (1) kann nun der Anteil der erreichten Knoten zu einem Zeitpunkt  $t$  berechnet werden. Für den Zeitraum von 0 bis 0,6 Sekunden ist der Wert von  $i(t)$  in Abbildung 30 dargestellt. Der Wert  $i(t)$  wurde zusätzlich mit 100 multipliziert, um den prozentualen Anteil erreichter Knoten anzuzeigen. Neben der Berechnung des Wertes  $i(t)$  für das *Random Waypoint Mobility Model* aus Abbildung 28 ist außerdem die Simulation der Nachrichtenverteilung für des entsprechende Modell dargestellt. Dazu wurden 20 verschiedene Simulationen durchgeführt und aus diesen die mittlere Anzahl erreichter Knoten für die einzelnen Zeitpunkte bestimmt.

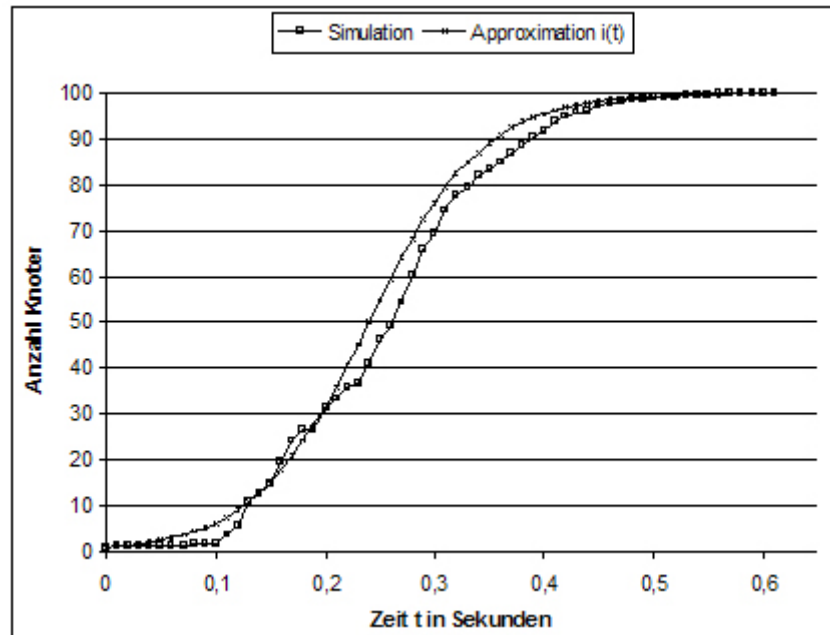


Abbildung 30: Vergleich von Simulation und Berechnung der Nachrichtenverteilung innerhalb eines Random Waypoint Mobility Models

## 6.2 Verteilung auf Geraden

Um im Abschnitt 6.3 die Nachrichtenverteilung für ein *AGbMM* vorhersagen zu können, muss neben der Verteilung innerhalb eines *Random Waypoint Mobility Models* auch die Ausbreitung einer Nachricht auf einer Kante analysiert werden. Die Intention dieses Kapitels ist: für eine gegebene Strecke die mittlere Zeit zu berechnen, die eine Nachricht benötigt um von einem Ende zum anderen Ende der Kante zu gelangen.

Es sei also eine Strecke  $e$  der Länge  $l$  gegeben und auf der Strecke befinden sich insgesamt  $N$  mobile Knoten. Es wird angenommen, dass die Nachricht ausgehend vom Startpunkt der Kante versendet wird. Denn bei der späteren Berechnung zur Verteilung im *AGbMM* werden die Nachrichten ausgehend von einem Cluster über die Kanten zum nächsten Cluster weitergeleitet. Das heißt, in diesem Abschnitt wird der Knoten mit der kürzesten Entfernung zum Startpunkt der Kante als Startknoten der Nachricht verwendet.

Vorausgesetzt, dass die  $N$  Knoten gleichverteilt auf der Kante sind, berechnet sich die mittlere Anzahl  $x$  mobiler Knoten innerhalb des Funkradius  $R$  wie folgt:

$$x = N \cdot \frac{2 \cdot R}{l}$$

Das heißt, das Verhältnis der Anzahl mobiler Knoten  $x$  innerhalb des Funkradius zur Gesamtzahl der Knoten auf der Kanten  $N$  ist gleich dem Verhältnis von doppelter

## Vorhersage der Nachrichtenverteilung

---

Funkreichweite zur Länge der Kante  $l$ . Die Funkreichweite wird verdoppelt, weil ein Knoten eine Nachricht in beide Richtungen der Kante versendet. Diese Formel gilt aber nur unter der Voraussetzung, dass die Kante mindestens die Länge der Funkreichweite  $R$  hat. Für den Fall, dass  $l$  kleiner als  $R$  ist, folgt automatisch  $x=N$ . Mobile Knoten die in der Nähe der Endpunkte der Kante im Mittel weniger Nachbar haben, werden bei der Berechnung von  $x$  nicht berücksichtigt.

Aus der durchschnittlichen Anzahl Nachbarn  $x$  kann jetzt die mittlere Sendereichweite  $S$  berechnet werden. Unter der Voraussetzung, dass die Knoten innerhalb des Funkradius  $R$  gleichverteilt sind, lässt sich die mittlere Sendereichweite  $S$  wie folgt berechnen:

$$S = R \cdot \frac{x}{x+1}$$

Analog zum *Random Waypoint Mobility Model* kann jetzt die Sendezeit  $s$  für eine Strecke der Länge  $\delta$  wie folgt ermittelt werden:

$$s(\delta) = \frac{\gamma \cdot \delta}{S} = \frac{\gamma \cdot \delta}{R \cdot \frac{x}{x+1}} = \frac{\gamma \cdot \delta \cdot (x+1)}{R \cdot x} \quad \text{mit} \quad x = N \cdot \frac{2 \cdot R}{l}$$

Nach einer Zeit  $s(l)$  haben demnach 100% der Knoten auf der Kante die Nachricht erhalten. Eine Funktion  $i(t)$  zur Berechnung des Anteils erreichter Knoten auf einer Kante zur Zeit  $t$  muss somit nachstehende Bedingungen erfüllen:

1.  $i(0) = 0$
2.  $i(s(l)) = 1$

Aufgrund der gleichmäßigen Verteilung der Knoten auf der Kante ist  $i(t)$  eine lineare Funktion und kann somit durch die zwei vorherigen Bedingungen eindeutig bestimmt werden.

$$i(t) = \frac{t}{s(l)}$$

Damit die Werte von  $i(t)$  nur aus dem Intervall  $[0,1]$  sind, wird die Funktion folgendermaßen erweitert:

$$i(t) = \begin{cases} 0 & \text{wenn } t \leq 0 \\ \frac{t}{s(l)} & \text{wenn } 0 < t < s(l) \\ 1 & \text{sonst} \end{cases}$$

## Vorhersage der Nachrichtenverteilung

Als Beispiel ist eine Kante der Länge 500m mit insgesamt 100 mobilen Knoten gegeben. Der Funkradius  $R$  beträgt 30m und die mittlere Sendeverzögerung  $\gamma$  ist 0,2 Sekunden. Daraus ergibt sich eine durchschnittliche Knotenanzahl  $x$  innerhalb der Funkreichweite von

$$x = N \cdot \frac{2 \cdot R}{l} = 100 \cdot \frac{2 \cdot 30\text{m}}{500\text{m}} = 12$$

Damit lässt sich die mittlere Sendereichweite  $S$  bestimmen:

$$S = R \cdot \frac{x}{x+1} = 30\text{m} \cdot \frac{12}{13} \approx 27,69\text{ m}$$

Die Sendezeit für die gesamte Kante der Länge  $l=500\text{m}$  ist demnach:

$$s(l) = \frac{\gamma \cdot l}{S} = \frac{0,2\text{ s} \cdot 500\text{m} \cdot 13}{30\text{m} \cdot 12} = \frac{65}{18}\text{ s}$$

In Abbildung 31 ist die Funktion  $i(t)$  für die beschriebene Kante dargestellt. Zusätzlich ist das Ergebnis der Simulation der Nachrichtenverteilung für diese Kante abgebildet. Dazu wurden 20 verschiedene Testläufe durchgeführt und aus diesen der Mittelwert für die einzelnen Zeitpunkte berechnet.

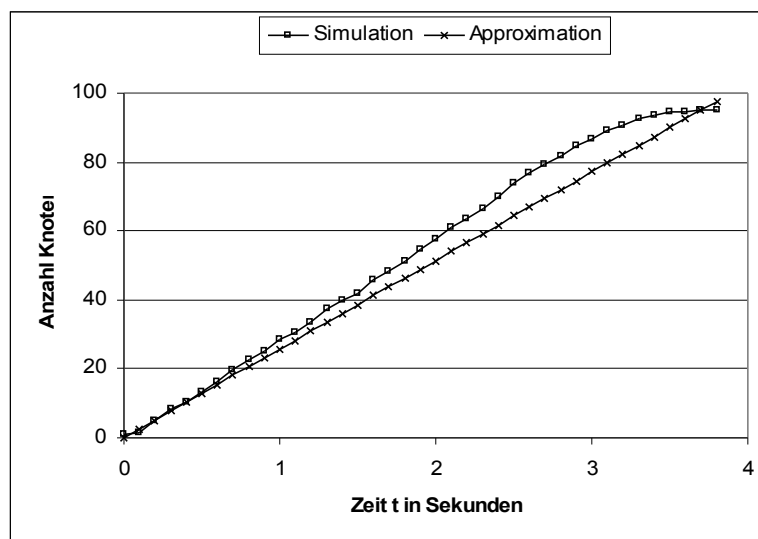


Abbildung 31: Vorhersage der Nachrichtenverteilung auf einer 500m langen Geraden mit 100 mobilen Knoten

Man erkennt, dass die Berechnungen langsamer verlaufen als die Simulation. Das hat verschiedene Gründe: zum einen wird bei Berechnung angenommen, dass die Nachricht immer vom Anfang bis zum Ende der Kante versendet wird. Da die Knoten zufällig auf der Kante verteilt werden, befindet sich weder direkt am Anfang noch am Ende der Kante ein



mobiler Knoten. Somit ist die Entfernung über welche die Nachricht gesendet werden muss, um alle Knoten zu erreichen, auf jedem Fall kleiner als die Länge der Kante. Weiterhin ist wie in Abschnitt 6.1, die Sendeverzögerung aus dem Intervall 0,1 bis 0,3 Sekunden. Das bedeutet, 50% der mobilen Knoten warten vor dem Weitersenden einer Nachricht weniger als 0,2 Sekunden. Das hat zur Folge, dass sich die Nachricht bei hohen Dichten schneller ausbreitet als mit einer mittleren Verzögerung von 0,2 Sekunden.

### 6.3 Verteilung im Area Graph-based Mobility Model

In den vorherigen Kapiteln wurde die Nachrichtenverteilung in einem *Random Waypoint Mobility Model* und auf Kanten analysiert. In diesem Abschnitt wird mit Hilfe dieser Ergebnisse die Nachrichtenverteilung in einem *AGbMM* insgesamt untersucht. Eine Nachricht wird dazu ausgehend von einem mobilen Knoten, innerhalb eines Cluster, in alle weiteren Cluster und Kanten des *AGbMM* versendet. Damit möglichst viele Knoten die Nachricht erhalten, ist eine entsprechend hohe Dichte der mobilen innerhalb der Cluster und auf den Kanten erforderlich. Des Weiteren wird für alle Berechnungen in diesem Abschnitt vorausgesetzt, dass die mittlere Knotenanzahl für die einzelnen Elemente des *AGbMM* bekannt ist. Diese kann entweder mit Hilfe der Formel aus Kapitel 5 näherungsweise berechnet oder durch Verwendung des in Abschnitt 4.2 vorgestellten Simulators approximiert werden.

Die Verteilung einer Nachricht innerhalb eines *AGbMM* setzt sich aus der Verteilung innerhalb der einzelnen Elemente des Modells zusammen. Das heißt, kennt man die Nachrichtenverteilung in den einzelnen Elementen eines *AGbMM*, kann die Ausbreitung im gesamten Modelle durch Addition der Werte aller Elemente berechnet werden. Weil die Nachricht zum Zeitpunkt 0 nur in einem Cluster versendet wird, muss für alle Elemente  $a$  der Zeitpunkt  $t_{Start}$  bestimmt werden, zu dem die Nachricht das Element  $a$  das erste Mal erreicht. Entsprechend der Startzeit  $t_{Start}$  kann dann die Funktion  $i(t)$  für das entsprechend Element auf der Zeitachse verschoben werden.

Demnach besteht die Analyse der Nachrichtenverteilung aus zwei Teilen: zum einen der Berechnung der Startzeit  $t_{Start}$  für jedes Cluster und jede Verbindung des Modells und zum anderen der Berechnung der Verteilung innerhalb der einzelnen Elemente. Mit Hilfe dieser Ergebnisse kann dann die Verteilung im *AGbMM* approximativ berechnet werden.

#### Berechnung der Startzeit

Zur Bestimmung der mittleren Zeit  $t_{Start}$ , nach der die Nachricht das erste Mal die einzelnen Elemente des Modells erreicht, wird folgender Algorithmus vorgestellt.

## Vorhersage der Nachrichtenverteilung

---

```
1   for each  $a \in A$  do
2       starttime( $a$ )  $\leftarrow \infty$ 
3       predecessor( $a$ )  $\leftarrow \emptyset$ 
4   starttime(Start-Cluster)  $\leftarrow 0$ 
5    $C \leftarrow$  Start-Cluster
6   while  $C \neq \emptyset$  do
7        $v \leftarrow$  remove element from  $C$  with smallest start time
8       for each outgoing connection  $e = (v,w)$  of  $v$  do
9           if (predecessor( $e$ ) =  $\emptyset$ ) then
10              predecessor( $e$ )  $\leftarrow v$ 
11               $t \leftarrow$  time to send message from start position in  $v$  to  $e$ 
12              starttime( $e$ )  $\leftarrow$  starttime( $v$ ) +  $t$ 
13               $e' \leftarrow$  get connection  $e'$  with  $e' = (w,v)$ 
14              if ( $e' \neq \text{NIL}$ ) then
15                  predecessor( $e'$ )  $\leftarrow v$ 
16                  starttime( $e'$ )  $\leftarrow$  starttime( $v$ ) +  $t$ 
17               $t' \leftarrow$  time to send message over  $e$ 
18              predecessor( $w$ )  $\leftarrow e$ 
19              starttime( $w$ )  $\leftarrow$  starttime( $e$ ) +  $t'$ 
20               $C \leftarrow C \cup w$ 
```

Zeile 1 bis 3 initialisieren die Startzeit und den Vorgänger aller Elemente, dazu wird die Startzeit auf unendlich gesetzt und der Vorgänger gelöscht. Anschließend erhält das initiale Clusters eine Startzeit von 0 Sekunden und wird zur Menge  $C$  hinzugefügt. Solange weiterhin Elemente in der Menge  $C$  enthalten sind, wird die while-Schleife der Zeilen 6 bis 20 ausgeführt. Nach Betreten der while-Schleife, extrahiert Zeile 7 das Cluster  $v$  mit der geringsten Startzeit aus der Menge  $C$ . Anschließend wird in Zeile 8 über allen hinausgehenden Kanten des Clusters  $v$  iteriert. Hat eine Kante  $e$  noch keinen Vorgänger, wird mit dieser Kante ab Zeile 10 weiter gearbeitet. Zunächst erhält die Kante  $e$  den Cluster  $v$  als Vorgänger. Im Anschluss wird die Startzeit der Kante  $e$  durch die Startzeit von  $v$  und die Zeit  $t$  bestimmt. Dabei ist  $t$  die mittlere Zeit, die benötigt wird, um die Nachricht von der Startposition in  $v$  zur Kante  $e$  zu senden.

In Zeile 13 wird die entgegengesetzte Kante  $e'$  zur Kante  $e$  bestimmt. Das ist die Kante, die gleichermaßen wie Kante  $e$  verläuft, aber entgegengesetzte Start- und Endpositionen hat. Wenn solch eine Kante  $e'$  existiert, erhält diese die gleiche Startzeit und den selben Vorgänger wie die Kante  $e$ . Ab Zeile 17 werden die Werte des nächsten Clusters initialisiert. Das Cluster  $w$  befindet sich am Endpunkt der Kante  $e$ . Die Nachricht erreicht Cluster  $w$  nachdem die Nachricht über die Kante  $e$  versendet wurde. Somit erhält das Cluster  $w$  die Kante  $e$  als Vorgänger und als Startzeit die Startzeit der Kante  $e$ , plus die Zeit

die zum Senden der Nachricht über  $e$  benötigt wird. Zuletzt wird das Cluster  $w$  zur Menge der zu bearbeitenden Cluster hinzugefügt.

Alle Elemente, die vom initialen Cluster aus zu erreichen sind, haben nach Ablauf des Algorithmus eine Startzeit  $t_{Start}$ , welche der mittleren Zeit entspricht, nach welcher die Nachricht das Element das erste Mal erreicht. Elemente in welche die Nachricht nicht versendet wird, besitzen eine unendliche Startzeit. Mit Hilfe der Startzeit  $t_{Start}$  und der Funktion  $i(t)$  aus Abschnitt 6.1 und 6.2 kann der mittlere Anteil erreichter Knoten für eine Elemente des *AGbMM* berechnet werden.

### Anteil erreichter Knoten eines Elementes

In Abschnitt 6.1 und 6.2 wurde für Cluster und Kanten gezeigt wie sich für eine Zeit  $t$  der Anteil erreichter Knoten  $i(t)$  berechnen lässt. Dabei wurde angenommen, dass die Nachricht das erste mal zum Zeitpunkt 0 versendet wird. Beginnt das Versenden der Nachricht erst nach  $t_{Start}$  Sekunden, muss die Funktion  $i(t)$  entsprechend um  $t_{Start}$  auf der Zeitachse verschoben werden. Das heißt, vor dem Zeitpunkt  $t_{Start}$  hat kein mobiler Knoten in diesem Element die Nachricht erhalten. Des Weiteren wurde in Abschnitt 6.1 und 6.2 gezeigt, dass nach einer bestimmten Zeit  $t_{Dauer}$  alle mobilen Knoten die Nachricht erhalten haben. Innerhalb eines Cluster ist  $t_{Dauer}$  die Zeit, die zum Senden vom Ursprungspunkt der Nachricht bis zur entferntesten Position im Cluster benötigt wird. Dieser Wert ist in Abschnitt 6.1 durch  $s(\delta)$  definiert. Bei Kanten ist  $t_{Dauer}$  die Zeit, die zum Senden der Nachricht über die gesamte Kante benötigt wird. In Abschnitt 6.2 wird diese Zeit durch  $s(l)$  berechnet, wobei  $l$  die Länge der Kante angibt. Demnach haben nach  $t_{Start}+t_{Dauer}$  Sekunden alle mobilen Knoten eines Elementes die Nachricht erhalten. Der Anteil erreichter Knoten des Elementes  $a$  eines *AGbMM* zur Zeit  $t$  kann somit wie folgt berechnet werden:

$$i(a, t) = \begin{cases} 0 & \text{wenn } t < t_{Start} \\ 1 & \text{wenn } t > t_{Start} + t_{Dauer} \\ i(t - t_{Start}) & \text{sonst} \end{cases}$$

Hierbei entspricht  $i(t)$  dem Anteil erreichter Knoten des Elementes  $a$  wie in Abschnitt 6.1 und 6.2 berechnet. Der Zeitpunkte  $t_{Start}$  gibt an, wann die Nachricht das erste Mal das Element  $a$  erreicht und  $t_{Dauer}$  entspricht der Zeit, die die Nachricht benötigt um alle Knoten in Element  $a$  zu erreichen. Das bedeutet, sowohl  $i(t)$  als auch  $t_{Start}$  und  $t_{Dauer}$  sind abhängig vom Element  $a$ . Der Wertebereich der Funktion  $i(a, t)$  ist aus dem Intervall  $[0,1]$  und entspricht damit dem Anteil erreichter im Element  $a$ , nicht aber dem Anteil erreichter Knoten im *AGbMM*.

### Anteil erreichter Knoten im Area Graph-based Mobility Model

Der Wertebereich einer Funktion zur Beschreibung des Anteils erreichter Knoten im *AGbMM* muss aus dem Intervall  $[0,1]$  sein. Zu Beginn hat ein mobiler Knoten die Nachricht. Im Laufe der Zeit, wird die Nachricht in die einzelnen Elemente des Modells versendet und erreicht somit immer mehr Knoten. Nach einer gewissen Zeit haben letztendlich alle mobilen Knoten die Nachricht erhalten. Aus den vorherigen Abschnitten ist bekannt, wann die Nachricht die einzelnen Elemente erreicht und wie sie sich innerhalb dieser ausbreitet. Insbesondere gibt die Funktion  $i(a,t)$ , für eine Element  $a$ , an wie groß der Anteil erreichter Knoten zum Zeitpunkt  $t$  ist. Das heißt, wird  $i(a,t)$  entsprechend dem Verhältnis mobiler Knoten im Element  $n(a)$  zur Gesamtzahl Knoten im Modell  $N$  gewichtet, entspricht dies dem Anteil erreichter Knoten im *AGbMM*. Aus der Summe dieser Werte für alle Elemente des Modells kann dann der gesamte Anteil erreichter Knoten im Modell berechnet werden. Demnach berechnet sich der Anteil erreichter Knoten  $i(t)$  eines *AGbMM* mit den Elementen  $A$  zum Zeitpunkt  $t$  wie folgt:

$$i(t) = \sum_{a \in A} \frac{n(a)}{N} \cdot i(a, t)$$

Als Beispiel hierzu betrachten wir das Modell der Abbildung 32 mit einer Gesamtzahl von 200 mobilen Knoten. Die mittlere Verteilung der Knoten auf die einzelnen Elemente des Modells ist mit den Formeln aus Kapitel 5 berechnet worden. Es befinden sich im Schnitt 80,95 Knoten in den einzelnen Clustern und 38,10 Knoten auf beiden Kanten des Modells. Das bedeutet, auf den beiden Kanten befinden sich jeweils 19,05 Knoten. Ausgehend vom Mittelpunkt des linken Clusters wird jetzt eine Nachricht im Modell versandt.

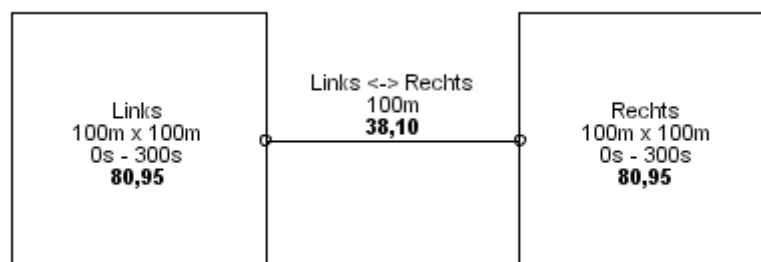


Abbildung 32: Einfaches *AGbMM* als Beispiel zur Berechnung der Nachrichtenverteilung

Zunächst muss ermittelt werden, wann die Nachricht die einzelnen Elemente das erste Mal erreicht. Bis zu den Kanten muss die Nachricht eine Entfernung von 50m zurücklegen und weitere 100m um das rechte Cluster zu erreichen. Entsprechend der vorgestellten Berechnungen benötigt die Nachricht zum Erreichen der Kanten etwa 0,34 Sekunden und bis zum rechten Cluster 0,93 Sekunden. Mit Hilfe der Funktion  $i(a,t)$  kann der Anteil

erreichter Knoten für ein Element  $a$  des Modells berechnet werden. Die Summe aller Elemente  $a$  des Wertes  $i(a,t)$  zum Zeitpunkt  $t$  ergibt dann den Anteil erreichter Knoten  $i(t)$  im  $AGbMM$  zum Zeitpunkt  $t$ .

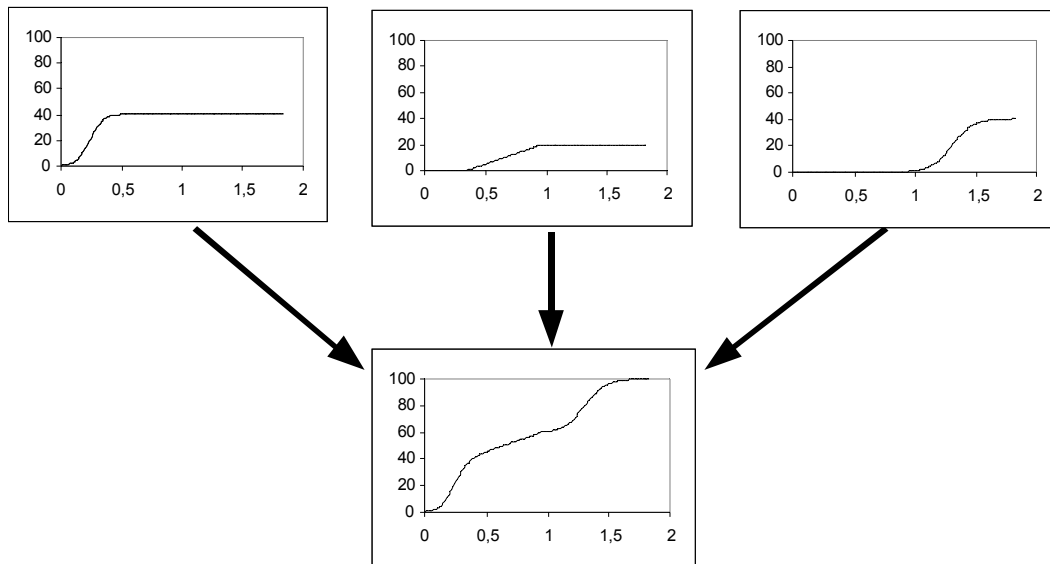


Abbildung 33: Berechnung des Anteils erreichter Knoten mit Hilfe der einzelnen Elemente eines  $AGbMM$

Der Anteil erreichter Knoten der einzelnen Elemente des Modells und deren Summe  $i(t)$  ist in Abbildung 33 dargestellt. Die drei oberen Diagramme zeigen die Verteilung der Nachricht innerhalb der einzelnen Elemente. Das linke Diagramm zeigt die Verteilung im linken Cluster und das Rechte die des rechten Clusters. Zur besseren Übersicht ist die Summe beider Kanten im mittleren Diagramm dargestellt. Im linken Diagramm und somit im linken Cluster beginnt die Ausbreitung der Nachricht zum Zeitpunkt 0. Nach etwa 0,5 Sekunden haben alle Knoten im linken Cluster die Nachricht erhalten. Nach 0,34 Sekunden erreicht die Nachricht beide Kanten und wird auf diesen verteilt. Nach etwa 0,93 Sekunden hat die Nachricht das rechte Cluster erreicht, bis zu diesem Zeitpunkt ist der Funktionswert des rechten Diagramms 0. Bis circa 1,8 Sekunden wird die Nachricht an alle Knoten im rechten Cluster verteilt. Im untere Diagramm ist die Summe der drei oberen Diagramme dargestellt. Das heißt, das Diagramm entspricht somit der analytischen berechneten Verteilung  $i(t)$  des  $AGbMM$ .

## 6.4 Beispiele zur Berechnung der Nachrichtenverteilung

In diesem Abschnitt werden einige ausgewählte Beispiele zur Berechnung der Nachrichtenverteilung innerhalb eines  $AGbMM$  vorgestellt. Dazu wurden jeweils Modelle

## Vorhersage der Nachrichtenverteilung

---

mit einer entsprechend hohen Knotendichte verwendet um eine Verteilung der Nachricht in alle Elemente zu garantieren.

Es wurden für jedes Szenario 20 unabhängige Simulationen durchgeführt und aus diesen der Mittelwert der Nachrichtenverteilung bestimmt. Die Diagramme zeigen jeweils, wie viel Prozent aller Knoten des Modells zu einem konkreten Zeitpunkt die Nachricht bereits erhalten haben. In allen Diagrammen sind jeweils die Kurven der simulierten und der berechneten Verteilung eingezeichnet. Neben der approximierten Berechnung sind zusätzlich in jedem Diagramm eine untere und eine obere Schranke für die Berechnung der Verteilung dargestellt.

Bei der unteren Schranke wird berechnet wie sich eine Nachricht im ungünstigsten Fall innerhalb des Modells ausbreitet. Das heißt, in den Clustern und auf den Kanten wird jeweils von den längsten möglichen Strecken ausgegangen. Die obere Schranke hingegen berechnet die Verteilung für den besten anzunehmenden Fall. Sowohl die Approximation als auch die Simulation sollten jeweils innerhalb dieser Schranken verlaufen. Demnach können die Schranken dafür verwendet werden, um die Nachrichtenverteilung innerhalb eines bestimmten Bereiches einzugrenzen.

Das erste Beispiel ist in Abbildung 34 dargestellt, es handelt sich um das einfache *AGbMM* aus Abschnitt 5.1. Insgesamt befinden sich 200 mobile Knoten innerhalb des Modells, somit ist eine ausreichend hohe Dichte garantiert, um die Nachricht in alle Elemente zu verteilen. Die mittlere Anzahl mobiler Knoten ist für jedes Element durch den fett dargestellten Zahlenwert angegeben. Die zu verteilende Nachricht wird von dem Knoten aus versendet, der sich zum Zeitpunkt Null am nächsten zum Mittelpunkt des linken Clusters befindet. Das heißt, zuerst wird die Nachricht im linken Cluster verteilt, erreicht anschließend die Kanten und breitet sich zum Schluss im rechten Cluster aus.

Die unteren und oberen Schranken grenzen wie beabsichtigt die Simulation und Approximation ein. Zu Beginn verbreitet sich die Nachricht im linken Cluster und erreicht nach circa 0,5 Sekunden die Kanten, dort breitet sie sich linear aus bis sie nach etwa einer Sekunde das rechte Cluster erreicht. Die Approximation weicht während der gesamten Verteilung nie mehr als 0,15 Sekunden von den Ergebnissen der Simulation ab. Das entspricht einer Abweichung von weniger als 10% zur gesamten Simulationszeit.

Es ist festzustellen, dass innerhalb der Cluster die Verteilung falsch vorhergesagt wird. So ist bei der Berechnung immer davon ausgegangen worden, dass innerhalb des Clusters die Knoten gleichverteilt sind. Wie aber bereits gezeigt, ist aufgrund der Struktur des *AGbMM* sowohl im Zentrum als auch an den Eingängen eines Clusters die Dichte höher als in den anderen Bereichen. Demzufolge werden zu Beginn der Verteilung im rechten Cluster mehr Knoten erreicht als angenommen. Darüber hinaus, wird die Nachricht nicht wie bei der epidemischen Verteilung angenommen, von einem Punkt in alle Richtungen verteilt.

## Vorhersage der Nachrichtenverteilung

Vielmehr breitet sich die Nachricht im rechten Cluster ausgehend vom Eingang, in nur eine Richtung aus und erhält somit einen nahezu linearen Verlauf.

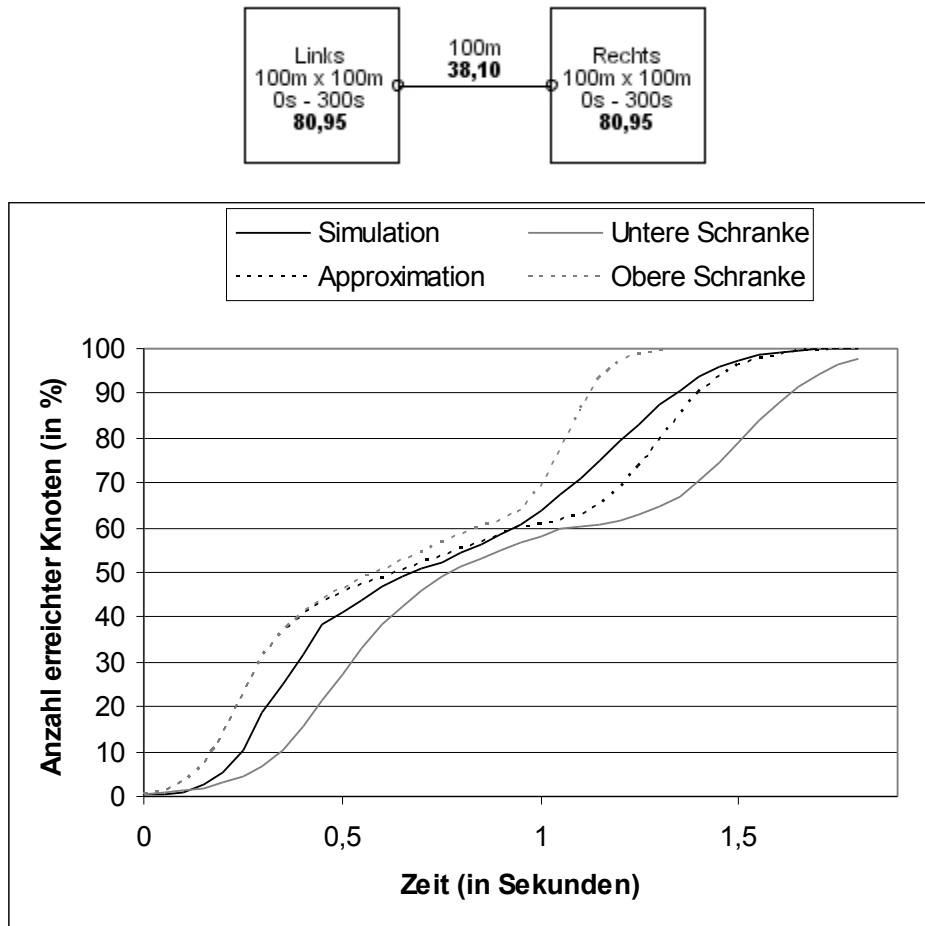


Abbildung 34: 1. Beispiel zur Berechnung der Nachrichtenverteilung

Als zweites Beispiel wird in Abbildung 35 die Nachrichtenverteilung in einem Modell mit fünf Clustern vorhergesagt. Diese fünf Cluster sind in Form eines Kreuzes angeordnet und durch entsprechende Kanten miteinander verbunden. Insgesamt befinden sich 500 mobile Knoten innerhalb des Modells. Die jeweils mittlere Anzahl mobiler Knoten der einzelnen Elemente ist durch den fett dargestellten Zahlenwert angegeben. Die Nachricht wird von dem Knoten initial versendet, der sich zum Zeitpunkt Null am nächsten am Zentrum des linken Clusters befindet. Die Nachricht breitet sich demnach erst im linken Cluster und anschließend im mittleren Cluster aus. Von hier aus wird die Nachricht dann nahezu zeitgleich in die restlichen drei Cluster verteilt. Es ist zu erkennen, zu welchen Zeitpunkten die Nachricht welches der Elemente erreicht. Weil sich aufgrund der vier Eingänge rund zwei Fünftel aller Knoten im zentralen Cluster befinden, steigt nach etwa einer Sekunde die Zahl neuer Infizierungen deutlich an.

## Vorhersage der Nachrichtenverteilung

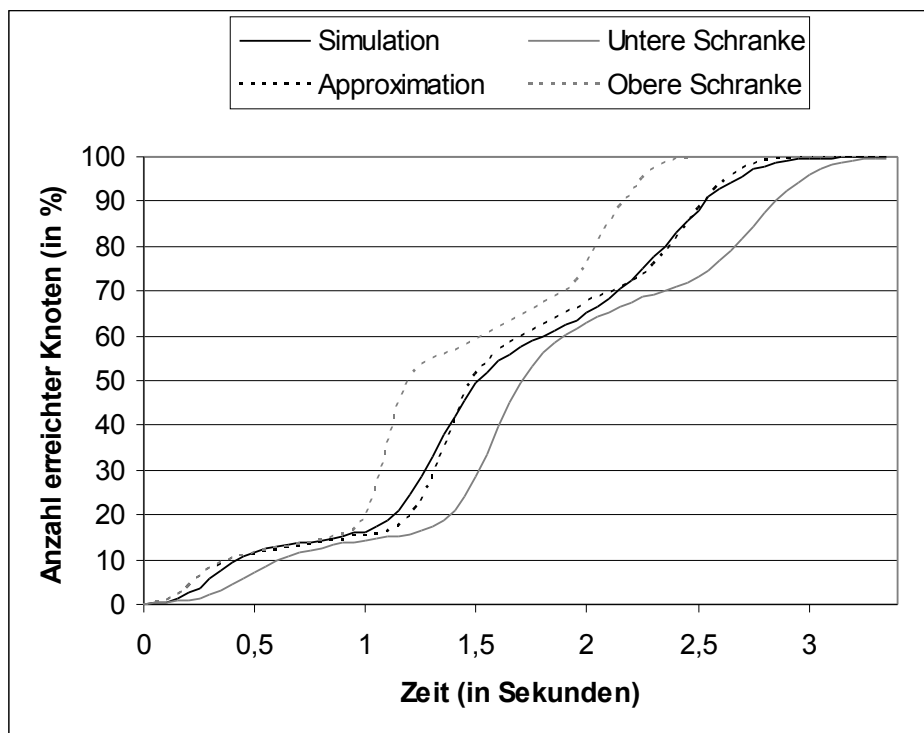
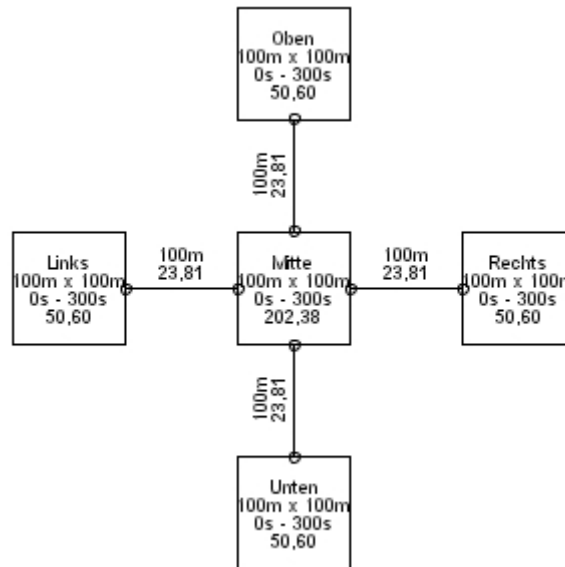


Abbildung 35: 2. Beispiel zur Berechnung der Nachrichtenverteilung

Während der gesamten Verteilung grenzen die untere und obere Schranke die Werte der Approximation und Simulation ein. Die Approximation weicht nie mehr als 0,15 Sekunden vom Wert der Simulation ab. Das entspricht einer Abweichung von weniger als 5% zur gesamten Simulationszeit.



## 7 Experimente mit Pull-Protokollen

Beim einfachen Fluten versendet ein Knoten eine neue erhaltene Nachricht automatisch an alle mobilen Knoten in seiner Nachbarschaft. Dabei benötigt ein mobiler Knoten meist nur eine geringe Zeit um die Nachricht zu verarbeiten, so dass die Verteilung einer Nachricht in den meisten Fällen nach kurzer Zeit beendet ist. Das heißt, die Nachricht breitet sich lediglich durch das Versenden von einem Knoten zum Nächsten aus. Bei Szenarien mit einer geringen Knotendichte ist es demnach nahezu unmöglich durch das Fluten einer Nachricht alle Teilnehmer zu erreichen. Für solche Szenarien empfiehlt sich daher die Verwendung anderer Verteilungsprotokolle.

Ein Ansatz um die Bewegung der mobilen Knoten zur Verteilung einer Nachricht zu nutzen sind Pull-Protokolle. Hierbei erfragt ein mobiler Knoten in regelmäßigen Abständen neue Informationen in seiner Nachbarschaft. Wenn einer der Nachbarn über neue Informationen verfügt, so sendet er diese zurück an den Initiator. Weil bei dieser Herangehensweise die Zeit zwischen den Anfragen deutlich größer ist als die Sendeverzögerung beim einfachen Fluten, wird eine Nachricht zum größten Teil mit Hilfe der Bewegung der mobilen Knoten innerhalb des Modells verteilt.

In diesem Kapitel wird die Nachrichtenverteilung mittels Pull-Protokollen in homogenen und heterogenen Bewegungsmodellen verglichen. Als Exemplar der homogenen Bewegungsmodelle wird das *Random Waypoint Mobility Model* und auf Seiten der heterogenen das *Area Graph-based Mobility Model (AGbMM)* verwendet. Die Untersuchungen orientieren sich an den Experimenten aus [14]. Darin wurden diverse Experimente mit verschiedenen Verteilungsprotokollen unter Verwendung des *Random Waypoint Mobility Models* realisiert. Das bedeutet, die in [14] umgesetzten Szenarien werden in eine möglichst vergleichbares *AGbMM* überführt und mit ihnen die gleichen Simulationen wie in [14] durchgeführt. In Abschnitt 7.1 werden zunächst die Protokolle vorgestellt mit denen anschließend die Experimente in Abschnitt 7.2 ausgeführt werden. Im Anschluss werden im Abschnitt 7.2 die Simulationen und deren Resultate beschrieben und mit den Ergebnissen aus [14] verglichen. Das System Modell wird analog zum Kapitel 6 gewählt. Das heißt, es wird WLAN (IEEE 802.11) als Technik wird zur Nachrichtenverteilung verwendet.

### 7.1 Verteilungsprotokolle

Dieser Abschnitt gibt einen Überblick über die bei den Experimenten verwendeten Verteilungsprotokolle. Dabei liegt der Schwerpunkt auf Pull-Protokollen, weil diese bei Szenarien mit geringerer Knotendichte, Nachrichten besser verteilen als einfaches Fluten. Die Gemeinsamkeit aller Pull-Protokollen ist, dass ein mobiler Knoten in regelmäßigen Abständen den aktuellen Informationsstand seiner Umgebung erfragt. Das bedeutet, zu

bestimmten Zeitpunkten sendet ein Knoten eine Anfrage an alle seine Nachbarn, ob diese über neue Informationen verfügen. Ist dies der Fall, sendet der entsprechende Knoten die jeweiligen Informationen an den Knoten, der die Anfrage initiiert hat.

### **Statisches Pull**

Beim statischen Pull sendet ein mobiler Knoten in regelmäßigen Abständen eine Anfrage an die mobilen Knoten in seinem Funkbereich. Dazu wird ein Intervall festgelegt, welches angibt, wie lange ein Knoten zwischen zwei aufeinander folgenden Anfragen wartet. Zum Beispiel könnte ein Knoten alle 5 Minuten neue Informationen bei seinen Nachbarn erfragen. Wenn bei der Simulation statisches Pull als Verteilungsprotokoll verwendet wird, erhalten alle mobilen Knoten das gleiche Intervall. Dabei beginnen alle Knoten mit der ersten Anfrage zu unterschiedlichen Zeitpunkten, um überlagernde Signale zu vermeiden.

### **Adaptives Pull**

Hierbei wird ausgenutzt, dass beim Abgleich der Informationen alle Knoten innerhalb des Funkradius diese Nachricht ebenfalls erhalten. Das bedeutet, jeder Knoten registriert welche neuen Informationen sich in seiner Umgebung befinden und kann diese mitzählen. Nachdem die Anzahl neuer Informationen eines Knoten einen festgelegten Grenzwert  $x$  überschritten hat, initiiert dieser Knoten eine Anfrage. Weil allein durch diesen Mechanismus keine Nachrichten verteilt werden würden, ist ein zusätzliches Intervall wie beim statischen Pull erforderlich. Das bedeutet, ist bis zu einem festgelegten Zeitpunkt der Grenzwert  $x$  nicht überschritten, wird ungeachtet dessen eine Anfrage initiiert. Somit ist gewährleistet, dass in regelmäßigen Abständen Nachrichten über vorhandene Informationen innerhalb des Modells verteilt werden.

### **Fluten + Pull**

Zusätzlich zu den reinen Pull-Protokollen sollen diese mit dem einfachen Fluten kombiniert werden. Hierzu versendet der Initiator einer Information diese sofort an alle Knoten in seiner Funkreichweite. Die Empfänger leiten die Information anschließend automatisch weiter. Somit wird eine Information direkt nach dem Erzeugen so weit wie möglich im Netz durch einfaches Fluten verteilt. Demzufolge muss eine neue Information nicht explizit von den anderen Teilnehmer erfragt werden. Die Kombination von Fluten und Pull-Protokoll wird mit beiden zuvor beschriebenen Pull-Protokollen umgesetzt.

## **7.2 Experimentelle Untersuchungen**

In diesem Abschnitt werden die Simulationen mit den zuvor beschriebenen Protokollen und deren Ergebnisse beschrieben. Des Weiteren werden die Ergebnisse den Simulationen aus [14] gegenübergestellt. Für die Simulationen wurde wie bereits in Kapitel 6 ns2 verwendet. Dabei wird eine Funkreichweite von 100 Metern, eine Bandbreite von 11Mbps/s und eine

## Experimente mit Pull-Protokollen

---

Größe der Datenobjekte von 256Byte verwendet. Darüber hinaus beträgt die Simulationszeit bei allen Szenarien eine Stunde.

Für die Bewegung der Knoten wurde in [14] ein *Random Waypoint Mobility Model* der Größe 1000m x 1000m verwendet. Dieses homogene Modell wird in ein heterogenes Modell mit ähnlicher Fläche und Parametern überführt (Abbildung 36).

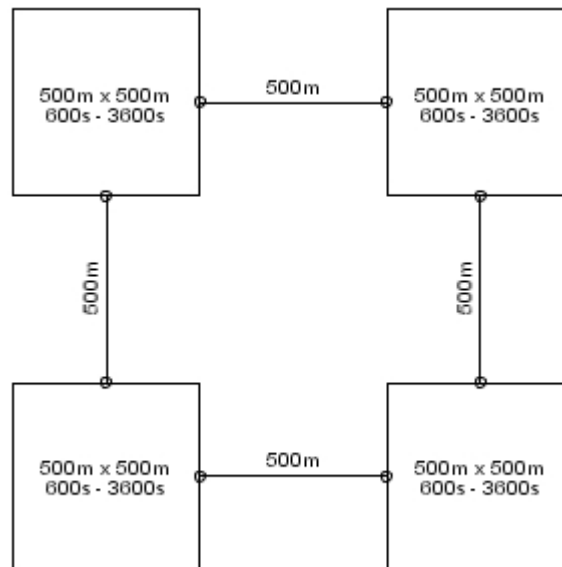


Abbildung 36: Heterogenes Modell mit vier Clustern

Das Modell hat, analog zum *Random Waypoint Mobility Modell* aus [14], eine Fläche von einem Quadratkilometer in der sich die Knoten frei bewegen können. Wobei die Fläche hier in vier Cluster mit einer Fläche von jeweils 250.000m<sup>2</sup> aufgeteilt ist. Zusätzlich verfügt das Modell über Wege die die benachbarten Cluster miteinander verbinden. Die Geschwindigkeit der Knoten innerhalb der Cluster beträgt 1,5m/s und auf den Kanten 2m/s. Zwischen den einzelnen Bewegungen in einem Cluster warten die mobilen Knoten zwischen 0 bis 60 Sekunden. Der Hauptunterschied dieses Modells zum *Random Waypoint Mobility Model* ist, dass sich die Knoten nicht mehr innerhalb des gesamten Quadratkilometers frei bewegen können. Das bedeutet, um eine Nachricht im gesamten Modell zu verteilen muss diese über die Kanten zu den anderen Clustern weitergeleitet werden.

Im Gegensatz zu Kapitel 6 wird nicht mehr die Verteilung einer einzelnen Nachricht untersucht. Vielmehr werden im Verlauf der Simulation fortlaufend Updates von unterschiedlichen Knoten erzeugt. Die Zeitpunkte der Updates sind hierbei normalverteilt über die gesamte Simulationszeit. Das heißt, zu Beginn und zum Ende der Simulation werden weniger Updates erzeugt als nach 30 Minuten Simulationszeit, wo die Zahl der neuer Updates das Maximum erreicht.

## Experimente mit Pull-Protokollen

---

Es wurden vier unterschiedliche Szenarien mit verschiedener Knotenanzahl (100 und 150) und verschiedener Anzahl von Updates (100 und 400) simuliert. Zur Simulationen wurden die gleichen Implementierungen und Parameter der Protokolle wie in [14] verwendet, somit können die Ergebnisse von *AGbMM* und *Random Waypoint Mobility Model* direkt verglichen werden. Für jedes Szenario wurden Simulationen mit den folgenden Protokolle und Parametern durchgeführt:

1. statisches Pull mit den Intervallen 60, 90, 120, 180, 240, 300 und 360 Sekunden.
2. Adaptives Pull mit den Grenzwerten 0, 1, 2, 4 und 8 und einem Intervall von jeweils 180 Sekunden.
3. Fluten kombiniert mit statischem Pull mit den Werten aus 1.
4. Fluten kombiniert mit adaptiven Pull mit den Werten aus 2.

Zum Vergleich der Modelle und Protokolle werden folgende Wert betrachtet:

**Aktualität der Daten:** Dieser Wert gibt an wie schnell die Informationen innerhalb des Netzes verbreitet werden. Der absolute Wert für ein gesamtes Szenario ergibt sich aus dem Mittelwert der Aktualität aller mobilen Knoten. Das heißt, wie viele aller erzeugten Updates hat ein Knoten bereits empfangen. Die Formel beruht auf dem Konzept von Cho und Garcia-Molina, welches in [15] vorgestellt wird.

**Netzlast:** Dieser Wert stellt die durchschnittliche Anzahl gesendeter und empfangener Daten pro mobilen Knoten während der gesamten Simulationszeit dar.

Im Folgenden werden die Ergebnisse des Szenarios mit 150 mobilen Knoten und 400 Updates genauer untersucht, die entsprechenden Werte der Simulationen sind in Abbildung 37 dargestellt.

Die Abzisse des Diagramms zeigt die durchschnittliche Netzlast pro Knoten und die Ordinate die mittlere Aktualität. Das bedeutet, je weiter ein Wert sich in der linken oberen Ecke des Diagramms befindet, umso effizienter ist das entsprechende Protokoll. Die einzelnen Protokolle sind jeweils durch eine Kurve dargestellt, wobei die Punkte der Kurve verschiedene Parameter repräsentieren. Beim statischen Pull mit und ohne Fluten sinkt die Netzlast bei wachsendem Intervall. Das heißt, der Punkt der Kurve mit der geringsten Last pro Knoten hat das größte Pull-Intervall (360 Sekunden) und umgekehrt. Analog verhält es sich bei den Protokollen mit adaptiven Pull, hier weisen die Punkte mit dem größten Grenzwert (8), die geringste Netzlast auf und die mit dem kleinsten Grenzwert (0), die höchste Last. Weil die Synchronisierung ebenfalls Netzlast erzeugt, unabhängig davon ob Daten aktualisiert werden oder nicht, verlaufen die Kurven der einzelnen Protokolle logarithmisch. Neben den Pull-Protokollen wurde zusätzlich eine Simulation mit einfachen Fluten durchgeführt, dieser Wert aber aufgrund der großen Abweichung (Aktualität=18,53 und Netzlast=127,50) nicht in das Diagramm eingezeichnet.

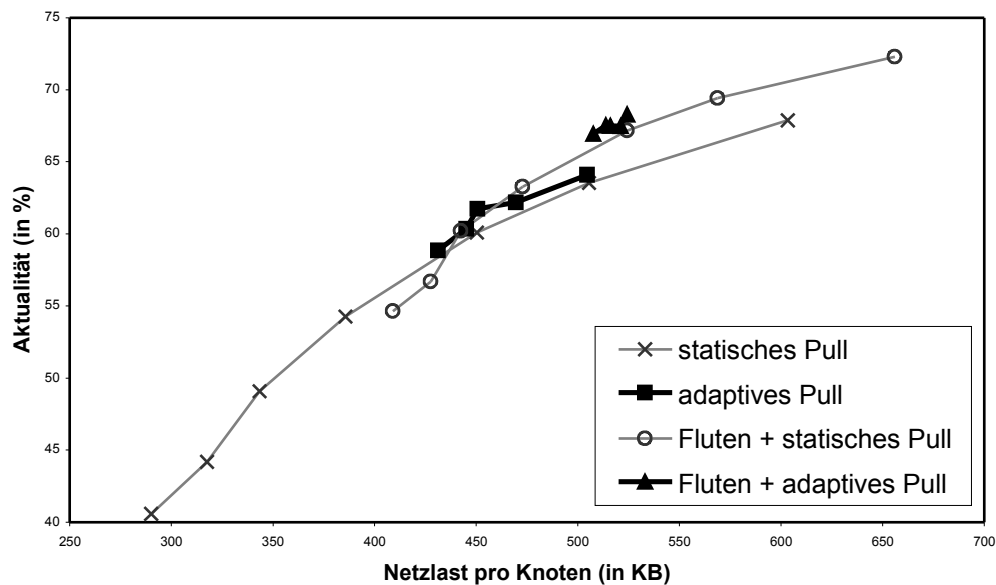


Abbildung 37: Vergleich verschiedener Pull-Protokolle bei Verwendung eines Area Graph-based Mobility Models mit 150 Knoten und 400 Updates

Durch die Verwendung von Pull-Protokollen verbessert sich die Aktualität um mindestens 20% gegenüber einfachem Fluten. Demnach sind, bei Szenarien mit einer geringen Dichte, Pull-Protokollen in jedem Fall effizienter als einfaches Fluten.

Weiterhin ist zu erkennen, dass die Protokolle kombiniert mit Fluten einerseits immer eine bessere Aktualität aufweisen als die entsprechenden Protokolle ohne Fluten, andererseits dafür aber das Netz stärker belasten. Ein Vergleich von statischem Pull mit der Kombination von statischem Pull + Fluten zeigt, dass bei kürzeren Intervallen der Unterschied der Protokolle geringer ist als bei großen Intervallen. Der Grund dafür ist, dass bei häufiger Synchronisation der Einfluss des Flutens weniger ausgeprägt ist als bei größeren Intervallen. Eine Gegenüberstellung der Pull-Protokolle, sowohl mit als auch ohne Fluten, zeigt, dass die adaptiven Protokolle jeweils etwas effizienter sind als die Protokolle mit statischem Pull. Die adaptiven Protokolle haben bei gleicher Aktualität eine etwas geringere Netzlast als das entsprechende statische Pull.

Zum Vergleich der Ergebnisse des *AGbMM* mit denen des *Random Waypoint Mobility Models* sind in Abbildung 38 die Resultate aus [14] dargestellt. Das Diagramm zeigt die Ergebnisse der Simulation unter Verwendung des *Random Waypoint Mobility Models* für die gleichen Parameter und Protokolle wie in Abbildung 37. Es ist zu erkennen, dass die Informationen im *AGbMM* insgesamt weniger aktuell sind (im Mittel um die 20%) als im *Random Waypoint Mobility Model*. Das heißt, die Updates können zwischen den vier Clustern nicht mehr so gut verteilt werden wie innerhalb eines einzelnen großen Bereichs.

## Experimente mit Pull-Protokollen

Ebenso ist festzustellen, dass der Effizienzgewinn der adaptiven Protokolle gegenüber den statischen Protokollen beim *Random Waypoint Mobility Model* größer ist als beim *AGbMM*. Im *Random Waypoint Mobility Model* ist die Netzlast pro Knoten beim adaptiven Pull bis zu 13% geringer als beim statischen Pull mit einer entsprechenden Aktualität. Im Gegensatz dazu beträgt beim *AGbMM* der Unterschied der Netzlast zwischen adaptiven und statischen Pull maximal 6%.

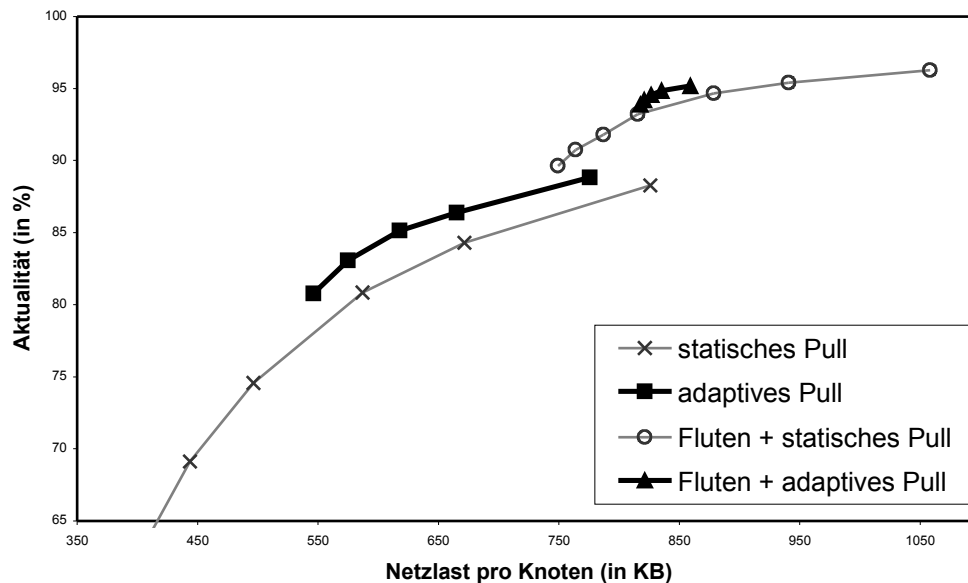


Abbildung 38: Vergleich verschiedener Pull-Protokolle bei Verwendung eines *Random Waypoint Mobility Models* mit 150 Knoten und 400 Updates

Betrachtet man die Auswirkungen der Hinzunahme des Flutens zu den Pull-Protokollen innerhalb der verschiedenen Bewegungsmodelle, so ist festzustellen, dass die Verbesserung der Aktualität der Informationen im *AGbMM* wesentlich geringer ausfällt als beim *Random Waypoint Mobility Model*. Bei *AGbMM* liegt die Steigerung der Aktualität beim statischen Pull bei maximal 15% und beim adaptiven Pull bei maximal 10%. Beim *Random Waypoint Mobility Model* hingegen, wird durch Hinzunahme des Flutens die Aktualität um bis zu 30% beim statischen und maximal 14% beim adaptiven Pull erhöht. Dies ist das Resultat der geringen Knotendichte der Kanten des *AGbMM*. Wird eine Nachricht mittels Fluten im *AGbMM* verteilt, so wird diese von maximal ein Viertel der Knoten empfangen, weil die Nachricht nie über die Kanten von einem Cluster zum nächsten weitergeleitet wird. Beim *Random Waypoint Mobility Model* hingegen werden beim Fluten einer Nachricht nur wenige Knoten am Rand des Modells nicht erreicht. Demzufolge werden im *Random Waypoint Mobility Model* beim Fluten einer Nachricht mehr Knoten erreicht als im *AGbMM*.

## Experimente mit Pull-Protokollen

---

Aus dem gleichen Grund ist die Steigerung der Netzlast beim *Random Waypoint Mobility Model* durch Kombination der Pull-Protokolle mit Fluten stärker ausgeprägt als beim *AGbMM*. Die Differenz der Netzlast zwischen den Protokollen statisches Pull und statisches Pull + Fluten mit einem Intervall von 360 Sekunden beträgt z.B. beim *Random Waypoint Mobility Model* über 350KB. Beim *AGbMM* hingegen ist die Differenz kleiner als 120KB.

Abschließend lässt sich feststellen: Bei heterogenen Szenarien mit einer geringen Dichte weisen Pull-Protokolle eine deutlich höhere Aktualität der Daten auf als einfaches Fluten. Somit ist die Verwendung dieser Protokolle in entsprechenden Szenarien auf jeden Fall sinnvoll. Des Weiteren haben die Simulationen gezeigt, dass die Nachrichtenverteilung mit Pull-Protokollen in homogenen Szenarien effektiver ist als in äquivalenten heterogenen Szenarien. Ungeachtet dessen weisen adaptive Protokolle, sowohl bei homogenen als auch bei heterogenen Szenarien, eine bessere Performanz auf als Protokolle mit statischem Pull. Wobei der Effizienzgewinn, der adaptiven Protokolle gegenüber den statischen Protokollen, bei homogenen Szenarien höher ist als bei heterogenen Szenarien. Infolge der schwachen Performanz des einfachen Flutens in heterogenen Szenarien mit geringer Dichte, ist der Effizienzgewinn durch Kombination der Pull-Protokolle mit Fluten in homogenen Szenarien etwa 50% größer als in vergleichbaren heterogenen Modellen. Die Wahl des Bewegungsmodells ist somit von entscheidender Bedeutung bei der Bewertung von Verteilungsprotokollen.

## 8 Abschlussdiskussion und Ausblick

Im diesem Kapitel erfolgt eine Zusammenfassung der durchgeführten Arbeitsschritte und der wichtigsten Ergebnisse der Diplomarbeit. Es werden einige ausgewählte Ansätze für weiterführende Arbeiten vorgestellt und die Ergebnisse in ihrer Gesamtheit diskutiert.

### 8.1 Abschlussdiskussion

#### Das Area Graph-based Mobility Model

Das *Area Graph-based Mobility Model (AGbMM)* ist ein heterogenes Bewegungsmodell, das erstmals in [4] vorgestellt wurde. Es wird durch einen Graphen  $G=(V,E)$  mit einer Menge von Knoten  $V$  und Kanten  $E$  beschrieben. Die Knoten des Graphen werden als rechteckige Flächen (Cluster) dargestellt und repräsentieren Bereiche, in denen sich die mobilen Knoten frei bewegen können. Die Bewegung eines mobilen Knotens besteht aus zwei Teilen. Zum einen aus der Bewegung innerhalb eines Clusters, welche durch ein *Random Waypoint Mobility Model* beschrieben wird und zum anderen aus der Bewegung von einem Cluster zum nächsten, nach Ablauf der Aufenthaltsdauer in einem Cluster.

Im Verlauf der Arbeit sind verschiedene Erweiterungen zum Basismodells des *AGbMM* entstanden. Durch diese Erweiterungen kann ein beliebiges Bewegungsmodell als internes Modell innerhalb eines Cluster verwendet werden. Es ist sogar möglich, innerhalb eines Clusters rekursiv ein weiteres *AGbMM* einzubinden. Somit ist das *AGbMM* kein klassisches Bewegungsmodell mehr, wie etwa das *Random Waypoint Mobility Model*, vielmehr ist es ein Meta-Modell, welches zur Beschreibung der Bewegung anderer Bewegungsmodelle verwendet werden kann. Mit Hilfe der Erweiterungen ist es so möglich, komplexere Szenarien zu konstruieren, vorhandene Modelle wiederzuverwenden und Modelle auf verschiedenen Ebenen zu betrachten.

Aufgrund seiner heterogenen Struktur weist das *AGbMM* kaum mehr Gemeinsamkeiten mit den klassischen homogenen Bewegungsmodellen, wie dem *Random Walk* oder dem *Random Waypoint Mobility Model*, auf. Der entscheidende Unterschied des *AGbMM* zu diesen Modellen, ist die ungleichmäßige Verteilung der mobilen Endgeräte und die unterschiedlichen Topologien. Das heißt, es kann eine Vielzahl verschiedener Bereiche mit abweichenden Dichten geben. Dies ist oft in der Realität zu beobachten. Zusätzlich können sich die Knoten beim *AGbMM* in bestimmten Bereichen nur auf vorgegebenen Strecken fortbewegen. Bei den klassischen Bewegungsmodellen ist dies nicht möglich, hier können sich die Knoten im gesamten Simulationsbereich frei bewegen.

Im Vergleich mit anderen komplexeren Bewegungsmodellen, wie etwa dem *City Section Mobility Model* oder dem *Graph-Based Mobility Model*, ist der wesentliche Unterschied, die



Existenz von Bereichen innerhalb derer sich die mobilen Knoten frei bewegen können. Aufgrund dessen und durch den rekursiven Ansatz ist das *AGbMM* für mehr Szenarien anwendbar als andere komplexe Bewegungsmodelle.

Zusammenfassend lässt sich festhalten, dass das *AGbMM* ein flexibles Bewegungsmodell ist, welches die Realität besser abbildet als viele vorhandene Modelle, wie das *Random Waypoint Mobility Model*. Das *AGbMM* ist für eine Vielzahl verschiedener Szenarien anwendbar und dementsprechend besser zur Analyse und Optimierung von Verteilungsprotokollen geeignet als die häufig verwendeten homogenen Modelle.

### Implementierung

Es wurde ein *AGbMM*-Simulator in Java entwickelt. Mit diesem Simulator ist es möglich, Modelle einzulesen und die Bewegung mobiler Knoten zu simulieren. Zusätzlich kann die Bewegung der Knoten in eine Datei exportiert und anschließend in Netzwerksimulatoren wie etwa ns2 verwendet werden. Als Format zur Beschreibung des *AGbMM* wurde XML gewählt. Dazu sind XML-Tags und Attribute vorgestellt worden, so können entsprechende Modelle für den Simulator entworfen werden. Weiterhin verfügt der Simulator über eine Vielzahl verschiedener Funktionen um die Modelle analysieren und visualisieren zu können.

### Verteilung der mobilen Knoten

Der erste Teil der theoretischen Analyse ist die Berechnung der Knotenverteilung auf die einzelnen Elemente eines *AGbMM*. Dazu wurden die Auswirkungen der einzelnen Parameter auf die Verteilung der Knoten innerhalb des *AGbMM* untersucht. Das Resultat des Kapitels ist folgende Formel zur Approximation der mittleren Knotenanzahl  $n(a)$  eines Elementes  $a \in V \cup E$ , für ein vorgegebenes *AGbMM* mit dem Graphen  $G=(V,E)$ :

$$n(a) = N \frac{t(a) \cdot e(a)}{\sum_{x \in A} t(x) \cdot e(x)}$$

$N$  ist dabei die Gesamtzahl mobiler Knoten im Modell,  $e(a)$  die Anzahl der Eingänge des Elementes  $a$  und  $t(a)$  die mittlere Verweildauer eines mobilen Knotens im Element  $a$ . Für die komplexe Berechnung der mittleren Verweildauer  $t(a)$  wurde eine Approximation hergeleitet, die für die meisten praktischen Anwendungen ausreichend ist. Die Formel zur Berechnung der mittleren Knotenanzahl  $n(a)$  ist für Modelle mit folgenden Eigenschaften anwendbar:

1.  $G$  ist zusammenhängend
2. Die Gewichtungen aller ausgehenden Kanten eines Clusters sind gleich
3. Wenn  $(a, b) \in E$ , gilt auch  $(b, a) \in E$ , für alle  $a, b \in V$

Bei den vorgestellten Beispielen, zur Berechnung der mittleren Knotenzahl, konnte sie mit einem maximalen Fehler von 2% zur Gesamtzahl mobiler Knoten bestimmt werden. Das heißt, wird bei einem Modell mit einer Gesamtzahl von 100 mobilen Knoten die Verteilung der Knoten berechnet, so weicht diese um maximal zwei Knoten vom exakten Wert ab. Wobei der mittlere Fehler etwa 1% beträgt.

### **Analyse der Nachrichtenverteilung**

Der zweite Teil der theoretischen Analyse ist die Nachrichtenverteilung innerhalb eines *AGbMM*. Hierfür wurde die Nachrichtenverteilung zunächst im *Random Waypoint Mobility Model* und auf Geraden untersucht. Diese Ergebnisse wurden anschließend zusammengeführt und daraus eine Formel zur Berechnung des Anteils erreichter Knoten  $i(t)$  zum Zeitpunkt  $t$  abgeleitet. Mit der Formel ist es möglich, für ein gegebenes *AGbMM* und eine vorgegebene Startposition einer Nachricht abzuschätzen, nach welcher Zeit ein bestimmter Prozentsatz der Knoten die entsprechende Nachricht erhalten hat.

Die Berechnung der Nachrichtenverteilung im *AGbMM* besteht aus zwei Teilen: Zunächst wurde für jedes Element des Modells die entsprechende Verteilung berechnet. Im Anschluss wurden die Verteilungen aller Elemente des *AGbMM* addiert und ergaben somit die gesamte Verteilung einer Nachricht innerhalb des *AGbMM*. Zur Berechnung der Nachrichtenverteilung innerhalb eines bestimmten Elementes des *AGbMM* muss zuerst die Zeit ermittelt werden, nach der die Nachricht im Durchschnitt das erste Mal das entsprechende Element erreicht. Ein Algorithmus, der diese Zeiten für alle Elemente eines *AGbMM* berechnet, wurde ebenfalls vorgestellt.

Zusätzlich wurde die Berechnung der Zahl erreichter Knoten an zwei Beispielen mit den Ergebnissen entsprechender Simulationen verglichen. Für beiden Szenarien konnten die Zeitpunkte, zu denen eine bestimmte Anzahl mobiler Knoten die Nachricht erhalten hat, mit einer Abweichung von maximal 10% zur gesamten Simulationszeit bestimmt werden.

### **Experimente mit Pull-Protokollen**

In diesem Abschnitt wurde die Nachrichtenverteilung in homogenen und heterogenen Szenarien mit geringer Knotendichte miteinander verglichen. Dazu sind Simulationen mit verschiedenen Pull-Protokollen durchgeführt worden. Die Experimente haben gezeigt, dass bei heterogenen Szenarien mit einer geringen Dichte Pull-Protokolle eine wesentlich bessere Verteilung der Informationen ermöglichen als einfaches Fluten. Demnach ist der Einsatz von Pull-Protokollen in entsprechenden Szenarien zweckmäßig. Weiterhin wurde gezeigt, dass adaptive Pull-Protokolle sowohl bei homogenen als auch bei heterogenen Szenarien effektiver sind als statische Pull-Protokolle. Zusätzliche ergaben die Simulationen, dass der Effizienzgewinn von adaptiven Protokollen gegenüber statischen Protokollen bei homogenen Modellen größer ist als bei heterogenen Modellen.

Darüber hinaus war festzustellen, dass bei heterogenen Modellen kaum Vorteile durch die Kombination von Pull-Protokollen mit Fluten erzielt wurden. Insgesamt war der Effizienzgewinn in homogenen Szenarien rund 50% größer als in vergleichbaren heterogenen Modellen. Das folgt daraus, weil bei heterogenen Szenarien mit einer relativ geringen Knotendichte einfaches Fluten äußerst ineffizient ist.

Als Fazit kann demnach festgestellt werden, dass die Wahl des Bewegungsmodells, bei der Simulation von MANETs, signifikant für die Bewertung der Verteilungsprotokolle ist.

## 8.2 Ausblick

Dieser Abschnitt gibt einen Ausblick auf Erweiterungen oder neue Aufgabenstellungen, welche aufbauend auf dieser Arbeit in Zukunft genauer untersucht werden könnten.

### Implementierung

Der vorhandene Simulator für *AGbMM* könnte um weitere Bewegungsmodelle z.B. das *Random Direction Mobility Model* erweitert werden. Somit wäre es möglich, innerhalb eines Cluster auch andere Modelle außer dem *Random Waypoint Mobility Model* und dem *AGbMM* zu verwenden. Darüber hinaus könnten Exports für weitere Netzwerksimulatoren implementiert werden.

Zur einfacheren und effizienteren Modellierung von Szenarien kann ein Programm kreiert werden, welches eine grafische Oberfläche zur Konstruktion von *AGbMM* zur Verfügung stellt.

### Theoretische Untersuchungen und Experimente

In Kapitel 5 wurde die Verteilung der mobilen Knoten im *AGbMM* analysiert. Diese Rechnungen waren nur für *AGbMM* mit bestimmten Eigenschaften anwendbar. Wie bereits innerhalb des Kapitels vorgestellt, wäre ein Ansatz zur Berechnung der Knotenverteilung in beliebigen *AGbMM* die Verwendung von Markov-Ketten.

Bei der Nachrichtenverteilung könnte einerseits die Verteilung mit einfachem Fluten genauer untersucht und berechnet und hierbei insbesondere die Knotenverteilung innerhalb der Cluster berücksichtigt werden. Andererseits könnten perspektivisch auch andere Verteilungsprotokolle oder Routing-Protokolle unter Verwendung des *AGbMM* analysiert werden. Da ein *AGbMM*-Simulator mit entsprechender Schnittstelle für ns2 existiert, müssten demzufolge nur die zu analysierenden Protokolle in ns2 implementiert werden.

## 9 Anhang

### 9.1 Literaturverzeichnis

- [1] G. Krüger, D. Reschke. Mobilkommunikation. *Lehr- und Übungsbuch Telematik, 2. Auflage*. Fachbuchverlag Leipzig im Carl Hanser Verlag München Wien. Kapitel 12, Seite 345.
- [2] D. Johnson, D. Maltz, J. Broch. DSR The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks. *Ad Hoc Networking*. Addison-Wesley. Kapitel 5, Seiten 139-172, 2001.
- [3] C. E. Perkins. Ad-hoc on-demand distance vector routing. *Ad-hoc on-demand distance vector routing, in MILCOM '97 panel on Ad Hoc Networks*, Nov. 1997.
- [4] S. Bittner, W.-U. Raffel, M. Scholz. The Area Graph-based Mobility Model and its Impact on Data Dissemination. In: *In Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW' 05), 2005 International Workshop on Pervasive Wireless Networking (PWN05)*, 268-272, Kauai Island, Hawaii, 8-12 March 2005.
- [5] T. Camp, J. Boleng, V. Davis. A Survey of Mobility Models for Ad Hoc Network Research. *Wireless Communications & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2(5): 483-502, 2002.
- [6] X. Hong, M. Gerla, G. Pei, C. Chiang. A group mobility model for ad hoc wireless networks. In: *Proceedings of the ACM International Workshop on Modeling and Simulation of Wireless and Mobile Systems (MSWiM)*, August 1999.
- [7] E. Royer, P.M. Milliar-Smith, L. Moser. An analysis of the optimum node density for ad hoc mobile networks. In: *Proceedings of the IEEE International Conference on Communication (ICC)*, 2001.
- [8] V. Davies. Evaluating mobility models within an ad hoc network. *Master's thesis*, Colorado School of Mines, 2000.
- [9] J. Tian, J. Hähner, C. Becker, I. Stephanov, K. Rothermel. Graph-Based Mobility Model for Mobile Ad Hoc Network Simulation. In: *Proceedings 35th Annual Simulation Symposium (ANSS-35 2002)*, 337-344, San Diego, CA, USA, 14-18 April 2002.

- [10] A. Jardosh, E. M. Belding-Royer, K. C. Almeroth, S. Suri. Towards realistic mobility models for mobile ad hoc networks. In: *Proceedings of the 9th annual international conference on Mobile computing and networking*, 217-229, San Diego, CA, USA, 2003.
- [11] C. Bettstetter. Smooth is Better than Sharp: A Random Mobility Model for Simulation of Wireless Networks. In: *Proceedings of the 4th ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, 19-27, Rome, Italy, 2001.
- [12] J. R. Norris. Introduction. *Markov Chains*. Cambridge Series in Statistical and Probabilistic Mathematics. 1997.
- [13] A. Khelil, C. Becker, J. Tian, K. Rothermel. An epidemic model for information diffusion in MANETs. In: *Proceedings of the 5th ACM international workshop on Modeling analysis and simulation of wireless and mobile systems*, 54-60, Atlanta, Georgia, USA, 2002.
- [14] J.-H. Böse, F. Bregulla, K. Hahn, M. Scholz. Adaptive Data Dissemination in Mobile ad-hoc Networks. In: *A. B. Cremers, R. Manthey, P. Martini, V. Steinhage (Hrsg.): INFORMATIK 2005 - Informatik LIVE!, Band 2, Beiträge der 35. Jahrestagung der Gesellschaft für Informatik e.V. (GI)*, 528-532, Bonn, 19.-22. September 2005.
- [15] J. Cho, H. Garcia-Molina. Synchronizing a database to improve freshness. In: *Proceedings of the 2000 ACM SIGMOD international conference on management of data.*, 117-128 Dallas, Texas, 2000.