

Ein dynamisches Adressierungsverfahren für Mobile Ad-Hoc Netze

Mesut Günes, Jörg Reibel

Lehrstuhl für Informatik IV, RWTH-Aachen
Ahornstr. 55
52074 Aachen
mesut@i4.informatik.rwth-aachen.de
joerg@i4.informatik.rwth-aachen.de

Abstract: Mobile Ad-Hoc Netze (MANET) bestehen aus einer Menge von mobilen Knoten, die drahtlos miteinander kommunizieren. Durch die Knotenmobilität ist die Netztopologie ständigen Veränderungen unterworfen und es existieren keine Einheiten, die jederzeit verfügbar sind.

Hauptproblem bei mobilen Ad-Hoc Netzen ist die schnelle Pfadfindung, welche durch die Knotenmobilität erschwert wird. Bevor jedoch ein Pfad zwischen zwei Kommunikationsendpunkten ermittelt werden kann, müssen die Knoten eindeutig identifiziert werden. Dabei werden zwei Komponenten benötigt: i) einheitliche Adressen, ii) ein Verfahren, welches den Knoten eindeutig Adressen zuweist. Als Adressen bieten sich IP-Adressen an.

In dieser Arbeit wird die *Agentenbasierte Adressierung*, ein dynamisches Adressierungsverfahren für mobile Ad-Hoc Netze, vorgestellt und bewertet. Weiterhin wird ein bekanntes Verfahren aus der Praxis und ein neuer Vorschlag aus der aktuellen Diskussion untersucht und bewertet.

1 Motivation

Ein Ad-Hoc Netz besteht aus einer Menge von mobilen Knoten, die drahtlos miteinander kommunizieren und keine vorhandene Infrastruktur, wie Basisstationen, benötigen. Ad-Hoc Netze bieten sich für beliebige Situationen an, in denen eine temporäre Kommunikationsverbindung gewünscht ist bzw. der Aufbau von Infrastruktur nicht möglich oder nicht gewünscht ist. Als Beispiel kann eine Gruppenbesprechung angeführt werden, bei der die Teilnehmer Daten austauschen möchten. Bei diesem Szenario ist es nicht wünschenswert langwierige Installationen und Konfigurationen durchführen zu müssen.

Im Gegensatz zu anderen drahtlosen Netzen wie GSM und WLAN, bei denen nur eine Verbindungsstrecke drahtlos ist, nämlich die zwischen dem mobilen Knoten und der Basisstation, muss bei Ad-Hoc Netzen der Kommunikationsverkehr typischerweise über mehrere Knoten drahtlos weitergeleitet werden. Diese Art von Netzen werden daher auch *Mobile Multihop Ad Hoc Networks* (MANET) genannt. Durch die Mobilität der Knoten ist die Netztopologie von Ad-Hoc Netzen ständigen

Veränderungen unterworfen. Es gibt keine zentralen Einheiten, die jederzeit erreichbar sind und somit wichtige Aufgaben übernehmen könnten. Weiterhin ist es nicht möglich, zuverlässige Aussagen über das Verhalten einzelner Knoten zu machen. Knoten können zu beliebigen Zeiten zum Netz hinzukommen, temporär oder dauerhaft vom Netz ausscheiden. Ferner kann sich ein Ad-Hoc Netz aufteilen und später wieder vereinigen.

Hauptproblem bei der Kommunikation in Ad-Hoc Netzen ist die schnelle Ermittlung von Pfaden zwischen Kommunikationsendpunkten, was durch die Knotenmobilität erschwert wird. Daher befassen sich die meisten Forschungsarbeiten zu Ad-Hoc Netzen mit dieser Fragestellung [Joh94, RT99, HJ00, Toh02]. Bevor ein Pfad zwischen zwei Knoten in einem Netz ermittelt werden kann, müssen die Knoten identifiziert werden. Zur Adressierung eines Netzes werden zwei Komponenten benötigt: i) einheitliche Adressen ii) ein Verfahren, welches den Knoten die Adressen eindeutig zuweist.

Im Internet werden IP-Adressen verwendet, die von einem Administrator manuell dem Knoten zugewiesen werden; dies geschieht hauptsächlich während der Installation. Um den Konfigurationsaufwand zu reduzieren wird immer mehr das *Dynamic Host Configuration Protocol* (DHCP) eingesetzt. Mit DHCP können Knoten während des Bootvorgangs oder der Netzwerkanmeldung die erforderlichen Adressinformationen von einem DHCP-Server beziehen.

In naher Zukunft werden die Knoten in einem Ad-Hoc Netz dieselben Geräte sein, die auch in lokalen Netzen eingesetzt werden. Hierzu sind Notebooks, PDAs und Smartphones zu zählen. Daher ist es wünschenswert, zur Adressierung von Ad-Hoc Netzen IP-Adressen zu verwenden.

In dieser Arbeit wird die *Agentenbasierte Adressierung*, ein dynamisches Adressierungsverfahren für mobile Ad-Hoc Netze, vorgestellt und bewertet. Weiterhin wird ein bekanntes Verfahren aus der Praxis und ein neuer Vorschlag aus der aktuellen Diskussion untersucht und bewertet. Die Untersuchungen basieren auf Simulationen mit dem ns-2 Simulator.

Aufbau der Arbeit

Die Arbeit ist im weiteren wie folgt aufgebaut. Ansätze aus der Literatur werden in Abschnitt 2 diskutiert. In Abschnitt 3 wird die Agentenbasierte Adressierung beschrieben. Die Simulationsumgebung und die Simulationsparameter werden in Abschnitt 4 vorgestellt. Die Ergebnisse der Untersuchungen und ein Vergleich der unterschiedlichen Verfahren werden in Abschnitt 5 diskutiert. Abschnitt 6 schließt mit einer Diskussion der Arbeit ab.

2 Bisherige Ansätze

Das Ziel der Adressierung ist, die Zuweisung einer eindeutigen und gleichartigen Bezeichnung der Knoten in einem Netz, um sie für Kommunikationsverbindungen wiederzufinden. Im Internet werden zur Adressierung IP-Adressen verwendet, die hierarchisch aufgebaut sind und sowohl einen Knoten als auch das Netz an dem der Knoten angeschlossen ist, eindeutig identifizieren. Die IP-Adresse dient dabei als eine logische Adresse auf der Netzwerkschicht und dient zur Pfadfindung. Für die eigentliche Zustellung eines Paketes an einen konkreten Host in einem lokalen Netz wird die MAC-Adresse der Sicherungsschicht benötigt. Die Abbildung der IP-Adressen auf die MAC-Adressen und umgekehrt wird durch die beiden Protokolle *Address Resolution Protocol* (ARP) [Plu82] und *Reverse Address Resolution Protocol* (RARP) [FMMT84] durchgeführt.

Die Zuweisung der IP-Adressen an Knoten erfolgt meistens manuell durch einen Administrator. Ein Knoten behält die zugewiesene Adresse typischerweise solange er nicht an ein anderes Netz angeschlossen wird. Bei großen Netzen erfordert diese Vorgehensweise einen hohen Verwaltungsaufwand. Bei Netzen mit mobilen Knoten sind dynamische Verfahren erwünscht, welche die Adressierung automatisch ohne den Eingriff des Benutzers durchführen.

2.1 Dynamic Host Configuration Protocol (DHCP)

Das *Dynamic Host Configuration Protocol* (DHCP) [Dro97] ist eine Erweiterung des BootP [CG85] Protokolls. Es dient der dynamischen Verwaltung von IP-Adressen. Mit DHCP können Knoten während des Bootvorgangs oder der Netzanmeldung benötigte Netzwerkinformationen von einem *DHCP-Server* anfordern. Dazu gehört eine gültige IP-Adresse, die Subnetzmaske, Adressen der primären und sekundären Nameserver und weitere Informationen, wie beispielsweise eine bestimmte Bootdatei. Wenn sich ein Knoten vom Netz abmeldet, wird seine IP-Adresse als frei markiert und kann vom DHCP-Server an andere Knoten vergeben werden. Sollte sich der Knoten später wieder am Netz anmelden, wird er wahrscheinlich eine andere IP-Adresse zugewiesen bekommen. Für die Versorgung der Knoten steht dem DHCP-Server ein Repertoire von IP-Adressen zur Verfügung. Wenn er alle IP-Adressen aus seinem Vorrat vergeben hat, kann er keine weiteren Knoten mehr bedienen. Die zugewiesenen IP-Adressen sind nur eine bestimmte Zeit lang gültig. Nach Ablauf dieser Frist müssen die Knoten die Lebensdauer ihrer Adressen verlängern. Ein DHCP-Server ist für die Versorgung eines Subnetzes konzipiert. Müssen mehrere Subnetze durch ein DHCP-Server bedient werden, können ausgezeichnete Hilfsknoten eingesetzt werden, die zwischen den Knoten und dem DHCP-Server vermitteln. Die Hilfsknoten werden als *DHCP-Relays* bezeichnet.

2.2 Zero Configuration Networking

Die *Zero Configuration Networking Work Group* der IETF arbeitet an einer Pro-

tokollfamilie, die eine automatische Konfiguration von Netzwerken erlaubt. Die Arbeitsgruppe definiert vier Bereiche: (i) IP-Adresskonfiguration [CA01], (ii) Abbildung von IP-Adressen auf Host-Namen [CTAG01], (iii) IP-Multicastkonfiguration [Gut01a] und (iv) Dienstsuche [Gut01b].

Im Arbeitsbereich (i) wird die automatische Generierung einer IP-Adresse aus dem Netz 169.254/16, welches für die Nutzung auf einer lokalen Verbindungsstrecke reserviert ist, beschrieben. Das Verfahren ist für die Konfiguration von Knoten, die am gleichen Leitungsabschnitt angeschlossen sind, entworfen, d.h. die zu konfigurierenden Knoten müssen alle Nachrichten untereinander empfangen können. Ein Knoten wählt sich aus dem reservierten Adressbereich zufällig eine IP-Adresse aus und testet, ob diese schon im Netz verwendet wird. Für die Überprüfung wird ARP verwendet. Wenn die gewählte Adresse besetzt ist, wählt sich der Knoten erneut eine IP-Adresse aus und testet diese. Dies wird solange durchgeführt, bis eine freie IP-Adresse gefunden ist.

2.3 Autokonfiguration für Ad-Hoc Netze

In [PMW⁺01] und [PKP01] wird ein Adressierungsverfahren für Ad-Hoc Netze vorgeschlagen, welches eine Erweiterung der von der Zero Configuration Networking WG vorgeschlagenen Methode ist. Im Gegensatz zum Verfahren aus [CA01], welches die Konfiguration von Knoten an einem Link reguliert, wird hier die Multihop-Architektur von Ad-Hoc Netzen mit berücksichtigt, d.h. die Adressierungsnachrichten können über andere Knoten weitergeleitet werden. Die beiden Dokumente unterscheiden sich in Details, der grundlegende Mechanismus ist bei beiden jedoch gleich. Daher wird im Folgenden eine leicht abstrahierte Version beschrieben.

Ein Knoten wählt sich zufällig eine Adresse aus dem reservierten Netz 169.254/16 bei IPv4 oder eine Adresse mit dem Präfix `MANET_PREFIX` bei IPv6. Danach sendet der Knoten eine Broadcastnachricht, um zu prüfen, ob die gewählte Adresse schon im Netz verwendet wird. Der Knoten wartet eine bestimmte Zeit¹ auf eine Antwort, wenn sich innerhalb dieser Zeit kein anderer Knoten meldet, geht der Knoten davon aus, dass die gewählte Adresse nicht verwendet wird und benutzt diese Adresse. Wenn die gewählte Adresse im Netz schon verwendet werden sollte, erhält der Knoten eine entsprechende Nachricht. Danach wählt sich der Knoten eine andere zufällige Adresse aus und überprüft diese erneut.

Während der Adressfindungsphase verwendet der Knoten eine temporäre Adresse aus einem speziell reservierten Adressbereich. Die Eindeutigkeit dieser temporären Adresse wird durch die kurze Verwendungszeit angenommen.

2.4 Mobile-IP

¹Die Autoren schlagen vor, `ADDRESS_DISCOVERY` auf 600ms zu setzen.

Mobile-IP [Per98] wurde als Erweiterung zu IP vorgeschlagen, um mobilen Hosts temporären Zugriff auf fremde Netze zu gewähren. Der mobile Host ist zwischen den temporären Netzanbindungen offline, d.h. wenn der mobile Host sich an einem Fremdnetz angemeldet hat, bleibt er auch bis zum Ende der Kommunikation angemeldet und verändert seine Adresse nicht.

Das Konzept baut auf zwei definierten Knoten, dem *Home-Agent* und dem *Foreign-Agent*, auf. Der Host besitzt eine feste IP-Adresse, die er verwendet, wenn er in seinem Heimatnetz ist. Beim Anmelden an einem Fremdnetz bekommt der Host eine temporäre Adresse vom Foreign-Agent zugewiesen, welche dem Home-Agent des Hosts bekannt gegeben wird. Dadurch ist der Home-Agent in der Lage, die an den Host gerichteten Pakete an seine aktuelle Position weiterzuleiten.

2.5 Zusammenfassung der bisherigen Adressiermechanismen

Die vorgestellten Verfahren wurden hauptsächlich mit dem Ziel entworfen, den Konfigurationsaufwand von Netzen zu verringern. Die Dynamik von Ad-Hoc Netzen wird daher durch die Verfahren nicht ausreichend widerspiegelt. Bei DHCP wird ein ständig erreichbarer DHCP-Server vorausgesetzt. Durch die Arbeiten der Zero Configuration Networking Arbeitsgruppe soll die Konfiguration von Knoten ohne die Existenz eines DHCP-Servers möglich sein; dies ist jedoch auf die Konfiguration von Knoten in einem Leitungsabschnitt beschränkt. Um das Verfahren für Ad-Hoc Netze anzupassen, wurde es so erweitert, dass die Konfigurationsnachrichten über Zwischenknoten weitergeleitet werden dürfen. Ein Nachteil dieses Ansatzes ist, dass keine netzwerkweite Eindeutigkeit der verwendeten Adressen garantiert ist. Insbesondere wenn sich ein Ad-Hoc Netzwerk aufteilt und später wieder vereint kann es passieren, dass mehrere Knoten die gleiche Adresse verwenden. Für die Überprüfung der Adressen nach der Vereinigung des aufgeteilten Netzes ist kein Mechanismus vorgesehen. Außerdem sieht das Verfahren gegen bösartige Knoten, die auf alle Überprüfungsanfragen antworten und somit andere Knoten am Eintritt in das Ad-Hoc Netz hindern, keinen Schutz vor. Als letztes Verfahren ist Mobile-IP zu erwähnen, welches für die Unterstützung von mobilen Knoten entworfen wurde, die temporär an ein Netz angeschlossen werden, jedoch zwischen zwei Netzanbindungen ausgeschaltet sind. Knoten in einem Ad-Hoc Netz sind jedoch nicht ständig eingeschaltet und erfordern Verfahren, die einen dynamischen Wechsel unterstützen.

3 Agentenbasierte Adressierung von Ad-Hoc Netzen

In diesem Abschnitt wird die *Agentenbasierte Adressierung* von Ad-Hoc Netzen vorgestellt. Das Ziel beim Entwurf war, ein Adressierungsverfahren zu entwickeln, welches Knoten in mobilen Ad-Hoc Netzen schnell und zuverlässig adressieren kann. Um dieses Ziel zu erreichen, wurden die folgenden Anforderungen aufgestellt:

- **Eindeutigkeit:** die Knoten in einem Ad-Hoc Netz müssen eindeutige IP-

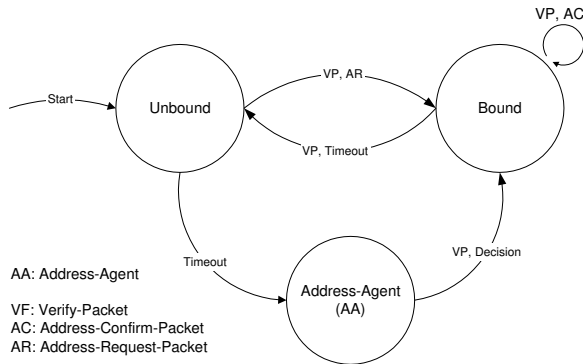


Abbildung 1: Zustandsgraph eines Knoten bei der Agentenbasierten-Adressierung

Adressen bekommen.

- **Einfach:** die benötigten Adressierungsdienste sollen von jedem Knoten im Ad-Hoc Netz übernommen werden können. Ein Knoten, der Adressierungsdienste anbietet, wird als *Adressierungsagent* (AA) bezeichnet.
- **Multihop:** die Adressierung von Knoten, die den Adressierungsagenten nicht direkt erreichen können, muss gewährleistet sein.
- **Robustheit:** gegen Aufteilung und Vereinigung von Ad-Hoc Netzen.
- **Adaptivität:** das Verfahren soll sich schnell an die Topologieänderungen anpassen und für eine schnelle Adressierung sorgen.
 - Wenn kein Adressierungsagent sich in einem Ad-Hoc Netz befindet, muss einer der Knoten die Aufgaben des Agenten übernehmen
 - Wenn mehrere Adressierungsagenten sich in einem Ad-Hoc Netz befinden, sollen sich alle bis auf einen abschalten.

Zustandsgraph bei der Agentenbasierten Adressierung

Bei der Agentenbasierten Adressierung befindet sich ein Knoten in einem der drei möglichen Zustände (siehe Abbildung 1). Ein Knoten befindet sich nach dem Start im Zustand *Unbound*, d.h., er besitzt keine Adresse. Von diesem Zustand, kann der Knoten sich in zwei andere Zustände bewegen. Wenn der ADDRESS_AGENT_DISCOVERY Timer abläuft, geht der Knoten in den *Adressierungsagent-Zustand* (AA) über, wodurch er selbst zum Adressierungsagenten wird und andere Knoten bedienen kann. Bekommt der Knoten nach dem Start und vor dem Ablauf des ADDRESS_AGENT_DISCOVERY Timer eine *Verify*-Nachricht (VP) von einem Adressierungsagenten, antwortet der Knoten mit einer *Address-Request*-Nachricht (AR).

Dadurch geht der Knoten in den Zustand *Bound*, d.h. der Knoten hat eine IP-Adresse. Der Knoten bleibt im Zustand *Bound*, solange er vom Adressierungsagenten *Verify*-Nachrichten erhält und auf diese mit *Address-Confirm*-Nachrichten (AC) antwortet. Den Zustand *Bound* verläßt der Knoten, wenn er lange Zeit kein *Verify*-Paket vom Adressierungsagenten erhält oder auf eine *Verify*-Nachricht nicht antwortet. Den AA-Zustand verläßt ein Knoten nur, wenn sich ein anderer Adressierungsagent im Ad-Hoc Netz befindet und der Knoten die Wahl zum Adressierungsagenten verliert.

Adressierungsverfahren Basiskomponente der *Agentenbasierten Adressierung* bildet der *Adressierungsagent*, der für die Adressierung eines Ad-Hoc Netzes verantwortlich ist. Die Funktionalität des Agenten kann von jedem Knoten übernommen werden. Zur eindeutigen Adressierung der Knoten verwaltet der Adressierungsagent eine *Address-List* (AL) der Knoten, die sich im Ad-Hoc Netz befinden. Die Address-List enthält die Zuordnungen von IP-Adressen zu MAC-Adressen. In regelmäßigen Intervallen sendet der Adressierungsagent *Verify*-Pakete, welche die Address-List und einen Zeitstempel enthalten. Jeder Knoten, der ein *Verify*-Paket erhält, überprüft, ob seine Adresse in der Address-List enthalten ist und schickt ein *Address-Confirm*-Paket an den Adressierungsagenten, falls er weiterhin im Ad-Hoc Netz bleiben möchte. Ein neuer Knoten im Ad-Hoc Netz, der sich in der Address-List im *Verify*-Paket nicht wiederfindet, meldet sich beim Adressierungsagenten mit einem *Address-Request*-Paket an. Erhält der Adressierungsagent nach Ablauf des ADDRESS_CONFIRM_TIMER keine Nachricht von einem Knoten, der in der Address-List enthalten ist, wird seine IP-Adresse als frei markiert.

Auswahl des Adressierungsagenten

Ein Knoten, der neu eingeschaltet wird, wartet ADDRESS_AGENT_DISCOVERY Zeiteinheiten auf ein *Verify*-Paket. Wenn er innerhalb dieser Zeit kein *Verify*-Paket von einem Adressierungsagenten erhält, geht er davon aus, dass kein Adressierungsagent vorhanden ist und wechselt selbst in den AA-Zustand. Danach verschickt er eine *Verify*-Nachricht.

Wenn sich in einem Ad-Hoc Netz mehrere Adressierungsagenten befinden, schalten sich alle bis auf einen Adressierungsagenten ab, sodass die eindeutige Verwaltung der Knoten gewährleistet bleibt und der benötigte Overhead minimiert wird. Die Reduktion der Adressierungsagenten auf eins erfolgt nach dem folgenden Verfahren.

Wenn ein Adressierungsagent AA_k eine *Verify*-Nachricht eines anderen Adressierungsagenten AA_l erhält, verfährt er wie folgt: AA_k berechnet die Anzahl der Knoten in seiner eigenen AL und die Anzahl der Knoten in der *Verify*-Nachricht. AA_k verläßt den Adressierungsagentenzustand, wenn die Anzahl der ihm bekannten Knoten kleiner ist als die von AA_l . Wenn die Anzahl der bekannten Knoten von AA_l und AA_k gleich ist, entscheidet die MAC-Adresse der beiden Agenten. Der Adressierungsagent mit der kleineren MAC-Adresse verbleibt im Adressierungsagentenzustand.

Wichtige Parameter der Agentenbasierten Adressierung

Die Leistung der Agentenbasierten Adressierung wird beeinflusst durch:

- Sendeintervall für Verify-Pakete

Wenn das Sendeintervall der Verify-Pakete zu klein ist, wird unnötiger Overhead erzeugt. Ist jedoch das Sendeintervall zu groß gewählt, muss gegebenenfalls ein neuer Knoten zu lange warten, bis er eine Adresse erhält und mit den anderen Knoten kommunizieren kann.

- ADDRESS_AGENT_DISCOVERY Timer

Dieser Timer gibt an, wie lange ein neuer Knoten wartet bis er in den Adressierungsagenten-Zustand übergeht und selbst Adressierungsagent wird. Der Wert dieses Timers hängt sehr stark vom Sendeintervall für Verify-Pakete ab. Über lange Zeit gesehen muss ein neuer Knoten im Durchschnitt $1/2$ Sendeintervall Zeiteinheiten auf ein Verify-Paket warten.

4 Simulationsumgebung

Die Simulationen wurden mit dem Netzwerksimulator *ns-2* [FV00] der UC Berkeley, LBL, USC/ISI, Xerox Parc und den Erweiterungen für drahtlose Netze des CMU Monarch-Projektes durchgeführt. In den Simulationen waren bis zu 50 mobile Knoten beteiligt, die dem IEEE 802.11 Standard entsprechen. Die Knoten kommunizieren mit einer Datenrate von 2Mbit/s und einer Reichweite von 250m. Als Ausbreitungsmodell wurde das *Two Ray Ground Model* eingesetzt, welches die Übertragung des Funksignals sowohl über die direkte Sichtlinie als auch über eine Reflektion am Boden modelliert.

Die Knoten bewegen sich auf einer Simulationsfläche von $1000\text{m} \times 1000\text{m}$ bei einer Simulationsdauer von maximal 3000 Sekunden. Die maximale Bewegungsgeschwindigkeit der Knoten beträgt 5 m/s. Die Bewegung der Knoten ist nicht konstant, sondern variiert zwischen 0 - 5 m/s, dabei legen die Knoten zwischen den einzelnen Bewegungen auch Pausen von bis zu 2 Sekunden ein. Das Bewegungsprofil der Knoten entspricht dem *Random-Waypoint Model* [BMJ⁺98].

Das Hauptinteresse bei den Untersuchungen galt der Zeit, die benötigt wird, um ein bestimmtes Ad-Hoc Netz *vollständig zu adressieren*. Diese Zeit ist wichtig, da die Kommunikation im Netz erst nach der Adressinitialisierung starten kann. Um das Verhalten der Verfahren in verschiedenen Situationen zu untersuchen, wurden mehrere Szenarien ausgewählt, bei denen eine Neuadressierung eines Ad-Hoc Netzes vollständig oder partiell durchgeführt werden musste.

4.1 Vollständige Adressierung eines Netzes (Einschaltvorgang)

In diesem Szenario benötigen Knoten, die in der Simulationsumgebung neu er-

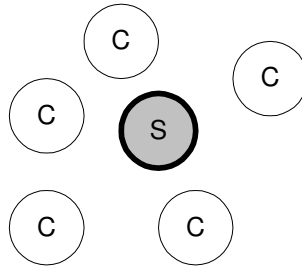


Abbildung 2: Ad-Hoc Netz mit einer Ebene, wobei alle Knoten direkt miteinander kommunizieren können.

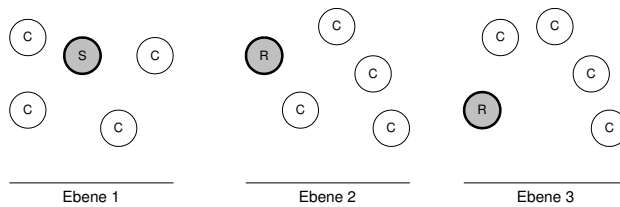


Abbildung 3: Anmeldung über mehrere Hops. Der Adressserver befindet sich in Ebene 1 und die Clients sind auf drei Ebenen verteilt. In Ebene 2 und 3 befinden sich jeweils ein Relay Knoten (R), die für die Weiterleitung zuständig sind.

scheinen, Adressen. Zu Beginn der Simulation besitzt kein Knoten eine Adresse. Zum Zeitpunkt $t = 0$ wird der entsprechende Adressserver aktiv und die Knoten fangen an, Adressanfragen zu stellen. Dies ist etwa mit dem Kaltstart einer ganzen Computerfarm vergleichbar. Dabei fungiert ein Knoten als Adressserver (S), der die anderen Knoten (C) mit Adressen versorgt (siehe Abbildung 2). In diesem Szenario werden zwei Fälle unterschieden:

1. *Eine Ebene:* Alle Knoten können den Adressserver direkt erreichen, so dass keine Weiterleitung des Steuerverkehrs erforderlich ist. Dieses Szenario ist bei einer Gruppenbesprechung mit wenigen Teilnehmern vorstellbar, in der die Teilnehmer ihre Notebooks verwenden. Einer der Computer dient dann als Adressserver.
2. *Mehrere Ebenen:* Nicht alle Knoten können den Adressserver direkt erreichen, daher muss der Steuerverkehr über andere Knoten weitergeleitet werden (siehe Abbildung 3). Knoten, die sich auf der gleichen Ebene befinden, können direkt miteinander kommunizieren. Einige der untersuchten Verfahren besitzen ausgezeichnete Knoten (R), die für die Weiterleitung der Steuerpakete verantwortlich sind.

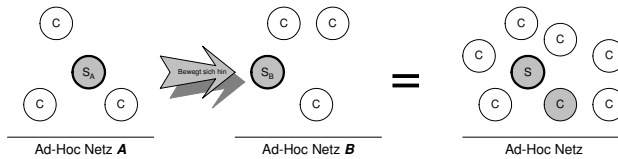


Abbildung 4: Vereinigung von Ad-Hoc Netzen

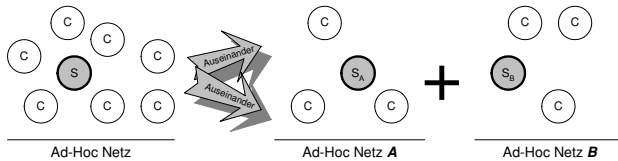


Abbildung 5: Aufspaltung eines Ad-Hoc Netzes

Diese beiden Varianten des Szenarios dienen zur Untersuchung der Eignung der Verfahren für multihop Ad-Hoc Netze.

4.2 Vereinigung von mehreren Netzen

In diesem Szenario entsteht aus mehreren Ad-Hoc Netzen ein neues Netz. Die Knoten der Urnetze bewegen sich gleichmäßig aufeinander zu, bis sie nur noch ein Netz bilden (siehe Abbildung 4). Die Knoten müssen im Zielnetz neu adressiert werden. Nach der vollständigen Verschmelzung der Urnetze darf im neuen Netz nur noch ein Adressserver aktiv sein, um die Eindeutigkeit der Adressen zu gewährleisten. Ein Beispiel für dieses Szenario aus der Realität ist etwa das Zusammentreffen zweier Arbeitsgruppen, die vorher für sich alleine gearbeitet hatten und nun gemeinsame Ergebnisse besprechen wollen. Aus beiden Gruppen entsteht eine neue größere Gruppe.

4.3 Aufteilung eines Netzes in mehrere Netze

Dieses Szenario ist das Gegenstück zu dem letzten Fall. Ein großes Netz teilt sich in mehrere kleinere Ad-Hoc Netze auf. Die neuen Netze benötigen einen eigenen Adressserver, der die Knoten im eigenen Netz versorgt (siehe Abbildung 5). Ein Beispiel für dieses Szenario aus der Realität könnte die folgende Situation sein. Innerhalb einer großen Gruppe bilden sich kleine neue Arbeitsgruppen, die nun eigene Besprechungen durchführen.

Die letzten beiden Szenarien dienen zur Untersuchung der *Adaptivität* und *Robustheit* der Verfahren. Dabei ist vorstellbar, dass dynamisch Mengen von Ad-Hoc Netzen entstehen, deren Strukturen sich mit der Zeit ändern, d.h. neue Ad-Hoc

Netze schmelzen zusammen und andere teilen sich in neue Netze auf.

5 Simulationsergebnisse

Im diesem Abschnitt werden die Ergebnisse der durchgeführten Simulationen präsentiert. Die Diskussion der Ergebnisse erfolgt gemäß den vorgestellten Szenarien aus Abschnitt 4. Die präsentierten Ergebnisse beruhen auf dem Median von 10 Simulationsthroughgängen.

5.1 Vollständige Adressierung eines Netzes (Einschaltvorgang)

Die Ergebnisse dieses Abschnitts beruhen auf Simulationen mit bis zu 50 Knoten, die auf mehrere Ebenen gleichverteilt sind. In den Fällen mit nur einer Ebene können alle Knoten direkt miteinander kommunizieren, so dass keine *Multihop-Kommunikation* nötig ist. In den Simulationen mit mehreren Ebenen können nur die Knoten auf der gleichen Ebene direkt miteinander kommunizieren. Der Kommunikationsverkehr zweier Knoten, die sich auf unterschiedlichen Ebenen befinden, erfolgt per multihop, d.h. die Pakete müssen über andere Knoten weitergeleitet werden. Gemessen wurde die Zeit für eine vollständige Adressierung aller Knoten im Ad-Hoc Netz.

DHCP: Bei den Simulationen zu DHCP gibt es maximal 40 Knoten und im Falle einer Ebene nur einen DHCP-Server. In den Fällen mit mehreren Ebenen gibt es zusätzlich zum DHCP-Server auf der ersten Ebene und jeweils einen DHCP-Relay auf den zusätzlichen Ebenen.

In Abbildung 6 sind die Ergebnisse für den Einschaltvorgang von DHCP mit 1, 2 und 3 Ebenen dargestellt. DHCP benötigt für die vollständige Adressierung aller 40 Knoten, die sich auf einer Ebene befinden, maximal 30 Sekunden. Die benötigte Zeit wächst stetig mit der Anzahl der Knoten im Netz. Dies ist durch die erhöhte Last am DHCP-Server zu erklären, der mehr Anfragen bearbeiten muss. Bei zwei Ebenen erhöht sich die Zeit bis alle Knoten adressiert sind auf maximal 125 Sekunden. Hier ist ein leichter Anstieg bis 10 Knoten zu beobachten. Danach stabilisiert sich die Zeit, die für die vollständige Adressierung benötigt wird. Die Vervierfachung der benötigten Zeit wird durch das DHCP-Relay bewirkt, das wie ein Flaschenhals wirkt. Alle Adressanfragen aus der zweiten Ebene werden über den Knoten, der als DHCP-Relay fungiert, an den DHCP-Server weitergeleitet. Bei drei Ebenen erhöht sich die benötigte Zeit für eine vollständige Adressierung aller Knoten im Durchschnitt auf 600 Sekunden. Die nochmalige Vervierfachung der benötigten Zeit wird durch das zweite DHCP-Relay bewirkt.

DHCP scheint sich für die Adressierung von Knoten, die sich über mehrere Hops anmelden müssen, nicht zu eignen. Dies liegt vor allem daran, dass die eingesetzten DHCP-Relays auf der zweiten und dritten Ebene wie ein Flaschenhals wirken. Verwirft der DHCP-Server die Adressanfrage eines Knotens aus der zweiten

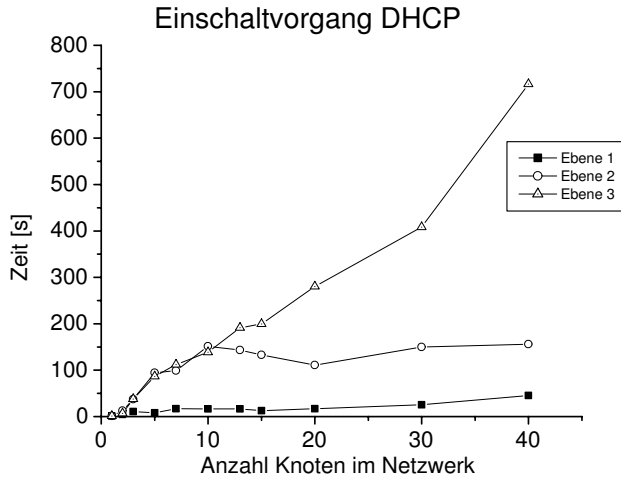


Abbildung 6: Vollständige Adressierung eines Ad-Hoc Netzes über drei Ebenen mit DHCP

oder dritten Ebene, benachrichtigt er zuerst das entsprechende DHCP-Relay, welches danach den Client über den Misserfolg informiert. Der Client hat nun zwei Möglichkeiten: entweder fordert er nochmals eine Adresse vom DHCP-Server an oder er re-initialisiert sich und startet danach eine neue Adressanfrage.

Autokonfiguration: In diesem Fall konnten auch Simulationen über 5 Ebenen und bis zu 50 Knoten durchgeführt werden. Die Knoten sind wiederum über die Ebenen gleichverteilt. Abbildung 7 stellt die Ergebnisse zur Adressierung mit Autokonfiguration dar. Alle Kurven liegen sehr nahe beieinander, im Bereich von 1-2 Sekunden. Für die vollständige Adressierung eines Ad-Hoc Netzes mit 50 Knoten, die über 5 Ebenen verteilt sind benötigt die Autokonfiguration unter 2 Sekunden.

Die benötigte Zeit für die vollständige Adressierung mit Autokonfiguration hängt von zwei Faktoren ab: i) der Zeit, die der Knoten auf eine Nachricht aus dem Netz wartet ii) der Anzahl der Versuche, bis der Knoten eine freie IP-Adresse findet. Für die zu wartende Zeit wird in [PKP01] der Defaultwert von $60\text{ms} \times \text{Netzdurchmesser}$ vorgeschlagen. In den durchgeführten Simulationen betrug der maximale Netzdurchmesser 10, womit die Wartezeit bei 600ms liegt und daher vernachlässigbar klein ist. Bei größeren Netzen könnte sich die Wartezeit als großer Nachteil erweisen.

Die durchschnittliche Anzahl der Versuche, bis der Knoten n eine freie Adresse findet, kann analog zum Geburtstagsproblem berechnet werden, wenn man annimmt, dass schon $n - 1$ Knoten im Netz sind und eine Adresse haben. Dabei wird von einem Klasse B Netz ausgegangen, d.h. 2^{16} Adressen. Für den 50. Knoten beträgt

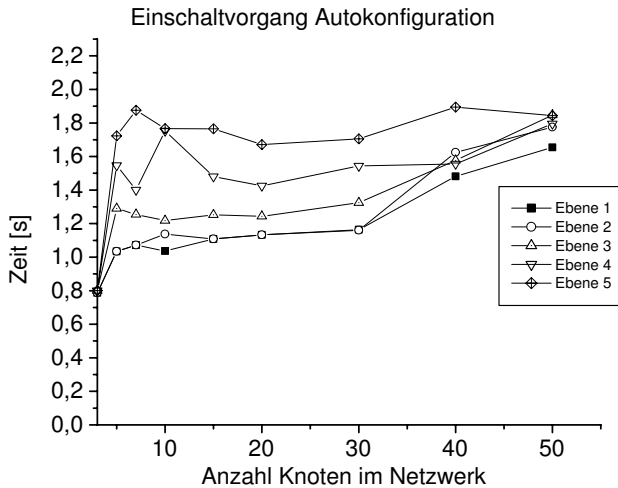


Abbildung 7: Vollständige Adressierung eines Ad-Hoc Netzes über fünf Ebenen mit Autokonfiguration

die Wahrscheinlichkeit für einen Konflikt beim ersten Versuch 0,018. Für den 305. Knoten beträgt die Konfliktwahrscheinlichkeit beim ersten Versuch schon 0,507 und den 780. Knoten 0,99. Bei wenigen Knoten kann man also davon ausgehen, dass keine Adresskonflikte entstehen.

Agentenbasierte Adressierung: Abbildung 8 stellt die Ergebnisse der Simulationen mit der Agentenbasierten Adressierung dar. Wie bei der Autokonfiguration konnten auch bei der Agentenbasierten-Adressierung Simulationen mit bis zu 5 Ebenen und 50 Knoten durchgeführt werden.

Die Zeit für eine vollständige Adressierung aller Knoten auf bis zu drei Ebenen liegt unter 2 Sekunden. Die benötigte Zeit bei einer und zwei Ebenen liegt bis zu 30 Knoten bei unter 1 Sekunde. Aus dem Graphen ist zu erkennen, dass bei mehr als 30 Knoten die benötigte Zeit sich von einer Sekunde auf ca. 2 Sekunden verdoppelt. Dies lässt sich durch den erhöhten Verkehr erklären, der beim Fluten der Adresspakete im Netz entsteht. Die benötigte Zeit zur Adressierung auf 4 Ebenen beträgt 3-4 Sekunden, dabei variiert der Graph wenig. Bei 5 Ebenen wiederholt sich diese Beobachtung, bei 4-5 Sekunden Gesamtzeit. Die Graphen der Ebenen 3, 4 und 5 sind am Anfang höher und nehmen dann ab. Dies liegt daran, dass bei geringer Anzahl der Knoten, die auf mehrere Ebenen verteilt sind, öfter Pakete nicht bis zum Adressierungsagenten durchkommen. Später sind genug Knoten im ganzen Ad-Hoc Netz, dass dieser Effekt nicht auftritt.

Wie aus Abbildung 8 zu erkennen ist, wächst die Zeit für die vollständige Adres-

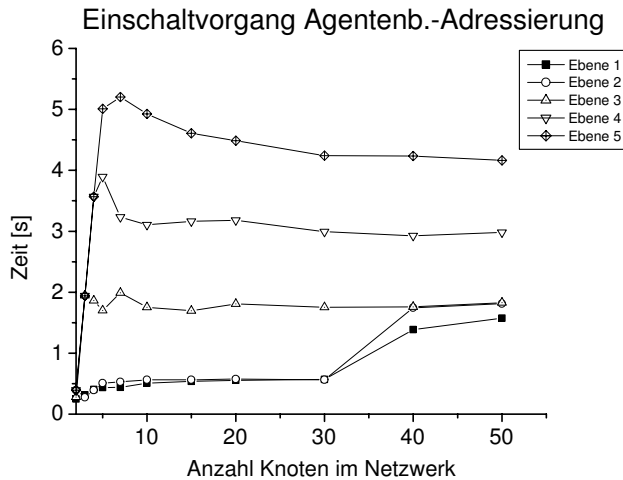


Abbildung 8: Vollständige Adressierung eines Ad-Hoc Netzes über fünf Ebenen mit der Agentenbasierten-Adressierung

sierung des Netzes etwa linear mit der Anzahl der Ebenen, auf denen die Knoten verteilt sind.

Es stellt sich nun die Frage, wie die Verbesserung von 600 Sekunden bei DHCP auf 5 Sekunden bei der Agentenbasierten Adressierung erzielt wird. Bei der Agentenbasierten Adressierung gibt es keinen ausgezeichneten Knoten auf den zusätzlichen Ebenen, der als Flaschenhals den Verkehr aufhält. Alle Knoten sind in der Lage, Pakete weiterzuleiten. Dadurch erreichen Adressanfragen den Agenten immer sowohl auf dem kürzesten Weg als auch auf dem längsten Weg. Der Nachteil, welcher durch das Flooding entsteht, wirkt sich hier nun als ein großer Vorteil aus, da verworfene Anfragen ein wenig später nochmals beim Agenten ankommen und bearbeitet werden können. Gegenüber DHCP zeigt sich die große Netzbelastung, die für die Adressierung benötigt wird, als nachteilig.

5.2 Vereinigung von zwei Netzen zu einem Netz

In diesem Abschnitt werden die Ergebnisse für das zweite Szenario diskutiert. Dabei bewegt sich Ad-Hoc Netz 1 auf Ad-Hoc Netz 2 mit 5m/s zu, bis sich die beiden Netze zu einem Netz vereinigt haben. Die Simulationen wurden mit 5, 10, 20, 30 und 40 Knoten durchgeführt, die auf beide Ad-Hoc Netze aufgeteilt sind. In den Graphen ist auf der x-Achse der Anteil der Knoten aufgezeichnet, die sich im Netz 2 befinden, d.h. die Knoten, die sich nicht bewegen. Gemessen wurde die Zeit bis alle Knoten vom Adressserver im Netz 2 eine Adresse erhalten hatten.

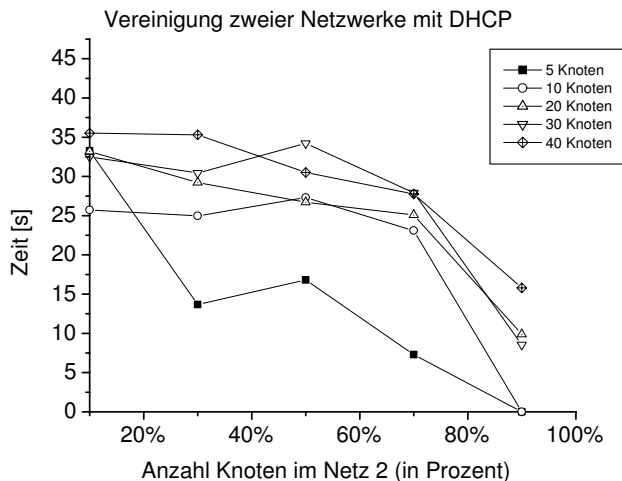


Abbildung 9: Vereinigung zweier Ad-Hoc Netze zu einem neuen Ad-Hoc Netz mit DHCP

DHCP: In Abbildung 9 sind die Ergebnisse für DHCP dargestellt. Je weniger Knoten sich im Netz 2 befinden, desto länger dauert es, bis das Endnetz vollständig adressiert ist. Bei 20% der Knoten im Netz 2 dauert die vollständige Adressierung mit DHCP ca. 35 Sekunden. Bei 90% der Knoten von Netz 2 dauert die Adressierung maximal 17 Sekunden.

Autokonfiguration: Abbildung 10 zeigt die Ergebnisse für die Adressierung mit Autokonfiguration. Die Ergebnisse liegen sehr nahe beieinander, zwischen 1,0 und 1,8 Sekunden.

Agentenbasierte-Adressierung: In Abbildung 11 sind die Ergebnisse für die Agentenbasierte Adressierung abgebildet. Die Zeiten für die vollständige Adressierung des Endnetzes bewegt sich hier zwischen 0,1 Sekunde bis 1,1 Sekunde.

5.3 Aufteilung eines Netzes in mehrere Netze

In diesem Abschnitt werden die Ergebnisse für das dritte Szenario diskutiert, bei dem sich ein großes Ad-Hoc Netz in mehrere kleinere Netze aufteilt. Die Knoten, die im ursprünglichen Netz verbleiben, benötigen keine neuen Adressen. Die Knoten, die sich abgespalten haben und in einem neuen Netz auftauchen, benötigen neue Adressen, um mit den anderen Knoten zu kommunizieren. Für jedes neue Netz wird ein neuer Adressserver benötigt, der die anderen Knoten mit Adressen versorgt.

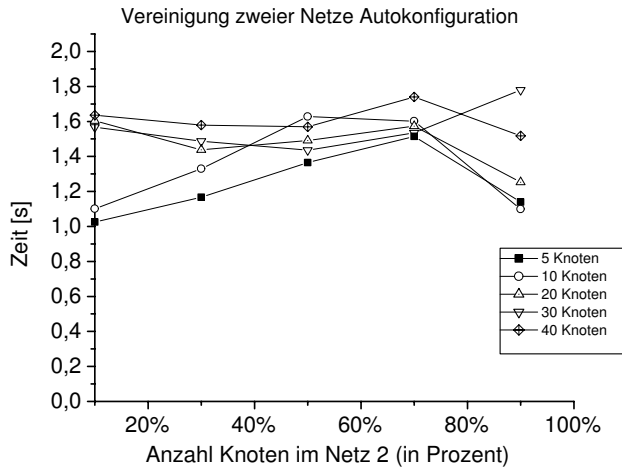


Abbildung 10: Vereinigung zweier Ad-Hoc Netze zu einem neuen Ad-Hoc Netz mit Autokonfiguration

DHCP: Da DHCP die automatische Aktivierung eines DHCP-Servers nicht unterstützt, wurde bei den Simulationen in den neuen Ad-Hoc Netzen die Wahl eines DHCP-Servers vom Simulationsskript gesteuert.

Autokonfiguration: Die Adressierung mit Autokonfiguration sieht zur Zeit keinen Mechanismus für das Erkennen von Aufteilung und Vereinigung von Ad-Hoc Netzen vor. Um bei den Untersuchungen eine Vergleichsmöglichkeit zu haben, wurde die Erkennung auch bei Autokonfiguration vom Simulationsskript gesteuert. Die Knoten im ursprünglichen Netz behalten ihre Adressen. Die Knoten im neuen Netz wählen zufällig neue Adressen und überprüfen, ob diese schon vergeben sind.

Agentenbasierte Adressierung: Der Adressierungsagent registriert bei seinen regelmäßigen Aktualisierungen die ausgeschiedenen Knoten. Diese IP-Adressen werden als frei markiert und können neu vergeben werden. Die Knoten im neuen Netz warten auf das Verify-Paket eines Adressierungsagenten. Danach beginnt die Adressierung. Die Bestimmung des Adressierungsagenten verläuft gemäß dem Verfahren aus Abschnitt 3.

Die Ergebnisse dieses Abschnitts stimmen bei allen drei Verfahren mit den Ergebnissen des ersten Szenarios mit nur einer Ebene überein. Dies ist durch die folgenden Faktoren zu erklären. Die Adressierung mit DHCP und Autokonfiguration kennen keine Mechanismen, um die Aufteilung und Vereinigung von Ad-Hoc Netzen zu erkennen. Um eine Vergleichsmöglichkeit zu haben, wurde in diesen beiden Fällen die Auswahl jeweils vom Simulationsskript gesteuert, wodurch keine zusätzliche Zeit

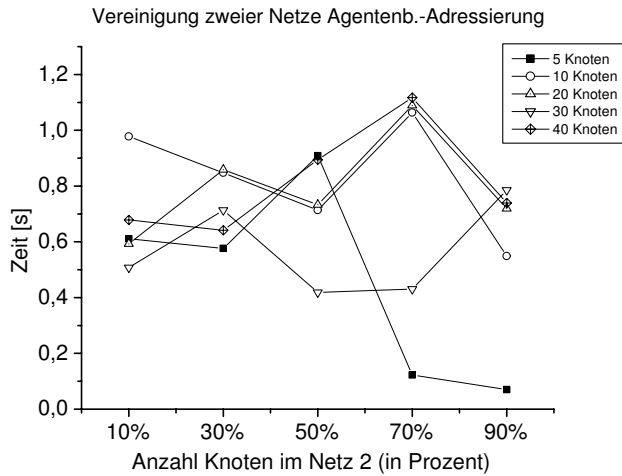


Abbildung 11: Vereinigung zweier Ad-Hoc Netze zu einem neuen Ad-Hoc Netz mit Agentenbasierter-Adressierung

in Anspruch genommen wurde. Bei der Agentenbasierten Adressierung wurde die Auswahl gemäß dem Verfahren aus Abschnitt 3 durchgeführt.

5.4 Vergleich der Verfahren

DHCP: Für die Adressierung mit DHCP werden ein DHCP-Server und bei mehreren Ebenen entsprechende DHCP-Relays benötigt. Es hat sich gezeigt, dass sich DHCP für die Adressierung von Ad-Hoc Netzen mit nur einer Ebene recht gut eignet, jedoch sehr schlechte Leistung erbringt, sobald die Knoten auf mehrere Ebenen verteilt sind. Weiterhin unterstützt DHCP nicht die Aufteilung und Vereinigung von Ad-Hoc Netzen. Ein weiterer Nachteil von DHCP ist die hohe Komplexität vom DHCP-Server und DHCP-Client.

Autokonfiguration: Das Verfahren arbeitet schnell und zuverlässig. Gegenüber den anderen Verfahren benötigt die Autokonfiguration keine zentrale Einheit für die Adressierung, welches einen zusätzlichen Vorteil bietet. Die ausgewählten Adressen werden im Netz geflutet, um die Eindeutigkeit der Adressen sicher zu stellen. Die Adressierung mit Autokonfiguration eignet sich daher grundsätzlich für den Einsatz in Ad-Hoc Netzen. Das Verfahren besitzt in der aktuellen Version in [PKP01] jedoch einige Schwächen, die gegen einen sinnvollen Einsatz des Verfahrens sprechen. Die Eindeutigkeit der Adressen im ganzen Ad-Hoc Netz ist nicht gewährleistet. Die Überprüfung findet jeweils nur einmal beim Start eines Knotens statt. Weiterhin unterstützt der Vorschlag keine Mechanismen für das Erkennen der Aufteilung und

Vereinigung von Ad-Hoc Netzen. Ein bösartiger Knoten, der auf alle Adressanfragen antwortet, kann die Adressierung der Knoten verhindern. Ferner wird durch das Fluten der Nachrichten eine hohe Last erzeugt.

Agentenbasierte Adressierung: Die Agentenbasierte Adressierung erfüllt die gestellten Anforderungen aus Abschnitt 3. Es eignet sich für den Einsatz in Ad-Hoc Netzen, da es die gestellte Aufgabe sowohl schnell als auch zuverlässig erfüllt. Im Gegensatz zu den anderen Verfahren ist die Agentenbasierte Adressierung in der Lage, die Aufteilung und Vereinigung von Ad-Hoc Netzen zu erkennen. Die Eindeutigkeit der vergebenen Adressen wird durch den Adressierungsagenten sicher gestellt. Weiterhin unterstützt das Verfahren die dynamische Anpassung der Anzahl der Adressierungsagenten. Ein Manko ist die hohe Last, die für die Adressierung erzeugt wird, da die Nachrichten im Netz geflutet werden.

6 Zusammenfassung und Ausblick

Mobile Ad-Hoc Netze zeichnen sich durch ihre hohe Dynamik aus. Alle Knoten sind mobil und können sich frei bewegen. In Ad-Hoc Netzen gibt es keine Einheiten, die jederzeit verfügbar sind. Daher erfordern diese Netze verteilt arbeitende, adaptive Verfahren.

In dieser Arbeit wurde die Agentenbasierte Adressierung, ein neues dynamisches Adressierungsverfahren für mobile Ad-Hoc Netze, vorgestellt. Weiterhin wurde ein bekanntes Verfahren aus der Praxis (DHCP) und ein neuer Vorschlag aus der aktuellen Diskussion (Autokonfiguration) untersucht und bewertet. Die Untersuchung beruhte auf einer Auswahl von Szenarien, die für zukünftige Ad-Hoc Netze typische Situationen wiedergeben.

Die Untersuchungsergebnisse haben gezeigt, dass existierende Adressierungsverfahren für den Einsatz in mobilen Ad-Hoc Netzen nicht geeignet sind, da sie die Dynamik dieser Netze nicht genügend berücksichtigen. Die Adressierung per Autokonfiguration stellt hier einen interessanten Ansatz dar, der jedoch noch eine ganze Reihe von Schwächen besitzt, die beseitigt werden müssen.

Die Agentenbasierte Adressierung erfüllt die geforderten Anforderungen von mobilen Ad-Hoc Netzen und arbeitet zuverlässig. Das Verfahren ist einfach, robust und adaptiv. Die Agentenbasierte Adressierung ist in der Lage, die Aufteilung und Vereinigung von Ad-Hoc Netzen zu erkennen. Hierdurch wird eine höhere Effizienz erreicht, da unnötiger Overhead gespart wird. Ein Manko des Verfahrens ist die hohe Last, die für die Adressierung benötigt wird.

Die Adressierung ist eine wichtige Grundlage der Kommunikation in Netzen. Daher erfordert diese Fragestellung in mobilen Ad-Hoc Netzen weitere Forschungsarbeiten. Offene Fragen sind: Skalierbarkeit, Schutz gegen Angriffe und die Reduzierung der erzeugten Last. Weiterhin sind Verfahren für die Anbindung von mobilen Ad-

Hoc Netzen an Festnetze, wie Lokale Netze und das Internet, erforderlich. Hier könnte Mobile-IP eine interessante Rolle spielen, wodurch ein Gateway für mobile Ad-Hoc Netze aufgebaut werden könnte.

Literatur

- [BMJ⁺98] Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu, and Jorjeta Jetcheva. A performance comparison of multihop wireless ad hoc network routing protocols. *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '98)*, pages 85–97, 1998.
- [CA01] Stuart Cheshire and Bernard Aboba. Dynamic configuration of ipv4 link-local addresses. <http://www.ietf.org/internet-drafts/draft-ietf-zeroconf-ipv4-linklocal-04.txt>, July 2001.
- [CG85] W.J. Croft and J. Gilmore. Bootstrap protocol, RFC 951. <http://www.ietf.org/rfc/rfc951.txt>, Sep 1985.
- [CTAG01] Octavian Catrina, Dave Thaler, Bernard Aboba, and Erik Guttman. Zeroconf multicast address allocation protocol (zmaap). <http://www.ietf.org/internet-drafts/draft-ietf-zeroconf-zmaap-02.txt>, Oct 2001.
- [Dro97] R. Droms. Dynamic host configuration protocol, RFC 2131. <http://www.ietf.org/rfc/rfc2131.txt>, Mar 1997.
- [FMMT84] Ross Finlayson, Timothy Mann, Jeffrey Mogul, and Marvin Theimer. A reverse address resolution protocol, rfc 903. <http://www.ietf.org/rfc/rfc903.txt>, June 1984.
- [FV00] Kevin Fall and Kannan Varadhan. *The ns Manual*, Nov 2000.
- [Gut01a] Erik Guttman. An api for the zeroconf multicast address allocation protocol (zmaap). <http://www.ietf.org/internet-drafts/draft-ietf-zeroconf-zmaap-api-00.txt>, Jun 2001.
- [Gut01b] Erik Guttman. Zeroconf host profile applicability statement. <http://www.ietf.org/internet-drafts/draft-ietf-zeroconf-host-prof-01.txt>, July 2001.
- [HJ00] Yih-Chun Hu and David B. Johnson. Caching strategies in on-demand routing protocols for wireless ad hoc networks. In *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking*. ACM, Boston, MA, August 2000.

- [HV99] Gavin Holland and Nitin Vaidya. Analysis of TCP performance over mobile ad hoc networks. In *Proceedings of the fifth annual ACM/IEEE international conference on Mobile computing and networking*, pages 219–230, 1999.
- [Joh94] David B. Johnson. Routing in ad hoc networks of mobile hosts. In *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications*, Dec 1994.
- [Per98] Charles E. Perkins. *Mobile IP: design and principles and practices*. Addison-Wesley, Bonn, Paris, Reading Massachusetts, 1998.
- [PKP01] Jung-Soo Park, Yong-Jin Kim, and Sung-Woo Park. Stateless address autoconfiguration in mobile ad hoc networks using site-local address, draft-park-zeroconf-manet-ipv6-00.txt. <http://www.ietf.org/internet-drafts/draft-park-zeroconf-manet-ipv6-00.txt>, July 2001.
- [Plu82] David C. Plummer. An ethernet address resolution protocol, RFC 826. <http://www.ietf.org/rfc/rfc826.txt>, 1982.
- [PMW⁺01] Charles E. Perkins, Jari T. Malinen, Ryuji Wakikawa, Elizabeth M. Belding-Royer, and Yuan Sun. IP address autoconfiguration for ad hoc networks, draft-perkins-manet-autoconf-01.txt. <http://www.ietf.org/internet-drafts/draft-perkins-manet-autoconf-01.txt>, November 2001.
- [RT99] Elizabeth M. Royer and Chai-Keong Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications*, pages 46–55, Apr 1999.
- [Toh02] Chai-Keong Toh. *Ad hoc mobile wireless networks: protocols and systems*. Prentice Hall, 2002. ISBN: 0-13-007817-4.