IP-Autokonfiguration in mobilen Ad-hoc-Netzwerken

Kilian Weniger

k.weniger@web.de
Institut für Telematik, Universität Karlsruhe (TH)

Abstract: Die Unabhängigkeit mobiler Ad-hoc-Netzwerke (MANETs) von einer Kommunikationsinfrastruktur eröffnet zahlreiche neue Möglichkeiten der mobilen Kommunikation. Auf Grund der speziellen Randbedingungen in diesen Netzen, wie die Notwendigkeit zur Selbstorganisation, die hochdynamische Topologie und die beschränkte Bandbreite und Energie, entstehen jedoch auch neue Anforderungen an Systemkonzepte und Protokolle. Eine essentielle, bisher aber nicht zufriedenstellend gelöste Problemstellung in diesem Zusammenhang ist die effiziente Selbstkonfiguration solcher Netze, auch Autokonfiguration genannt. Dieser Artikel gibt einen Überblick über einige Ergebnisse der Dissertation mit dem Titel "IP-Autokonfiguration in mobilen Ad-hoc-Netzwerken" [Wen04], in der mögliche Ansätze analysiert und eine sehr effiziente und robuste Lösung zur Autokonfiguration von MANETs vorgeschlagen wird. Die Leistungsfähigkeit konnte sowohl durch simulative Analyse und Vergleich mit anderen Verfahren als auch in einem Testsystem bestätigt werden.

1 Einleitung

Heutige Mobilfunksysteme sind auf eine Kommunikationsinfrastruktur angewiesen, die aus Basisstationen, Switches, Routern etc. besteht. Die mobilen Teilnehmer können dabei i.A. nur unter Nutzung dieser Infrastruktur miteinander kommunizieren, selbst wenn sich die Teilnehmer in Funkreichweite voneinander befinden. In den letzten Jahren hat ein neuartiger Netzwerktyp in der Forschungsgemeinde stark an Bedeutung gewonnen: Ein MANET ist ein drahtloses Netz aus mobilen Endgeräten, welches vollkommen unabhängig von einer drahtgebundenen Infrastruktur die Kommunikation ermöglicht. Die mobilen Endgeräte übernehmen dabei selbst die Aufgabe der Infrastruktur und fungieren z.b. als drahtlose Router, um Datenpakete an entfernte Endgeräte weiterzuleiten.

Der Einsatz solcher Netze kann z.B. sinnvoll sein, wenn die Infrastruktur nicht funktionsfähig bzw. nicht vorhanden ist oder um die Kosten zur Nutzung einer Infrastruktur einzusparen. Auch haben solche Netze den Vorteil, dass sie sehr schnell und kostengünstig errichtet werden können. Möglich ist aber auch die Kombination mit einem Infrastrukturnetz, z.B. um den mobilen Teilnehmern einen Zugang zum Internet anbieten zu können, die sich außerhalb des Abdeckungsbereichs z.B. eines WLAN-Hotspots befinden oder um die Sendeleistung von Basisstation und mobilen Geräten zu reduzieren. Weitere Anwendungen finden sich u.a. in der Verkehrstelematik, in Sensornetzen, im Heimbereich, im militärischen Bereich oder in Katastrophenszenarien [CCL03].

MANETs stellen aufgrund ihrer besonderen Eigenschaften hohe Anforderungen an die Kommunikationsprotokolle, erfordern neue Systemkonzepte und eröffnen neue Fragestellungen. So muss sich das Netz als Folge der Unabhängigkeit von einer Infrastruktur selbst organisieren können. Gleichzeitig kann sich die Netztopologie aufgrund der Mobilität der Endgeräte ständig und unvorhersehbar ändern. Dies bedingt u.a. angepasste Routingprotokolle. Außerdem sollten die Protokolle möglichst wenig Kontrollverkehr verursachen, da sowohl die verfügbare Bandbreite als auch die Batterielaufzeit der mobilen Geräte stark beschränkt sind und oftmals die wichtigsten Leistungsengpässe darstellen.

Ein Großteil der Forschungsarbeiten konzentrierte sich bisher auf die Entwicklung neuer Routingprotokolle, da diese eine Grundvoraussetzung für die Kommunikation in einem MANET darstellen. So wurden in der MANET-Arbeitsgruppe der Internet Engineering Task Force (IETF) - der Standardisierungsorganisation für Internet-Protokolle - bereits mehrere Routingprotokolle entwickelt. Diese Protokolle setzten allerdings voraus, dass die Geräte bzw. deren Netzwerkschnittstellen bereits konfiguriert sind. Der wichtigste Konfigurationsparameter des IP-Stacks ist die IP-Adresse, die im Netz eindeutig sein muss. Da aber im Gegensatz zu traditionellen Netzen weder ein Administrator noch eine zentrale Instanz (z.B. DHCP-Server) vorhanden sind, die eindeutige Adressen zuweisen können, muss sich das Netz selbst auf eine dezentrale Art und Weise konfigurieren. Eine besondere Herausforderung ist dabei, dass die Zuweisung einer zu einem bestimmten Zeitpunkt eindeutigen Adresse nicht ausreicht, denn aufgrund der Mobilität der Geräte kann es dazu kommen, dass zwei getrennte, unabhängig voneinander konfigurierte Netze miteinander verschmelzen. Das resultierende Netz kann daraufhin doppelte Adressen bzw. Adresskonflikte aufweisen, die möglichst schnell und effizient erkannt und aufgelöst werden müssen. Dauert die Auflösung des Konflikts zu lange, kann es zu zahlreichen Paketverlusten aufgrund von fehlgeleiteten Paketen und damit zur Unterbrechung der Kommunikation kommen.

Die hier zusammengefasste Dissertation beschäftigt sich mit dieser *Selbst- oder Autokonfiguration* auf der Basis von IP. Der Schwerpunkt liegt dabei in der dynamischen Zuweisung und Beibehaltung eindeutiger IP-Adressen im Netz, denn eindeutige Adresse sind von entscheidender Bedeutung: Sie sind eine Grundvoraussetzung für die Funktionsfähigkeit eines Datennetzes. Die hier zusammengefasste Arbeit analysiert existierende Verfahren [WZ04], identifiziert deren Schwachstellen und stellt einen neuartigen Ansatz namens ”Passive AutoConfiguration in Mobile Ad hoc Networks (PACMAN)” vor. Die Arbeit ist im Kontext des BMBF-Projekts ”IPonAir”[1] [ZWS03] entstanden.

Im Folgenden wird zunächst ein Überblick über die wichtigsten Komponenten von PACMAN gegeben (Abschnitt 2). Auf eine der Komponenten wird in Abschnitt 3 etwas näher eingegangen. Schließlich wird in Abschnitt 4 kurz auf die Evaluation von PACMAN in Simulator und Testsystem eingegangen. Für Details bezüglich Analyse, Konzept und Evaluation sei auf die Dissertation [Wen04] und auf dazugehörige Veröffentlichungen (z.B. [Wen05]) verwiesen.

[1] siehe http://www.iponair.de

2 Funktionale Komponenten von PACMAN

PACMAN verfolgt den Ansatz, die strikte Trennung der Protokollschichten gemäß des ISO/OSI-Schichtenmodells aufzuweichen und schicht- und protokollübergreifenden Informationsaustausch zu erlauben. Hauptsächlich werden Informationen vom Routingprotokoll verwendet, um eine effiziente Autokonfiguration zu erreichen. Verschiedene funktionale Komponenten lassen sich unterscheiden, die über wohl definierte Schnittstellen miteinander kommunizieren. Die Komponenten sind in Abbildung 1 dargestellt.

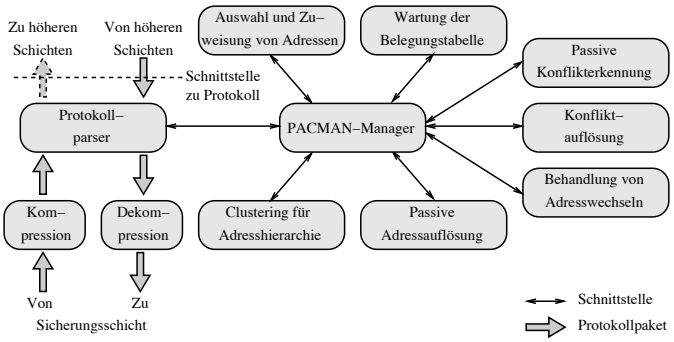


Abbildung 1: Die funktionalen Komponenten von PACMAN

Aufgrund des Fehlens einer zentralen Instanz, weist sich ein Endgerät selbst eine Adresse nach einem probabilistischen Algorithmus zu[2]. Bei der initialen Adresszuweisung können verschiedene Szenarien unterschieden werden. Falls ein Gerät einem bereits konfigurierten Netz beitritt, gewährleistet der Algorithmus und die Verwaltung einer Belegungstabelle, dass keine Adresskonflikte bei der Zuweisung auftreten. Falls zwei oder mehr Geräte sich jedoch gleichzeitig eine Adresse zuweisen, können sie die gleiche Adresse wählen und ein Adresskonflikt kann auftreten, der möglichst schnell erkannt und aufgelöst werden muss. In diesem Zusammenhang stellt sich die Frage, ob die Geräte überhaupt mit einer IP-Adresse identifiziert werden müssen oder ob nicht eine längere oder kürzere Adresse verwendet werden kann. Man kann argumentieren, dass sehr lange Adressen, wie sie z.B. von IPv6 verwendet werden, das Problem der Adresszuweisung lösen können, da in diesem Fall Konflikte bei zufälliger Wahl der Adresse sehr unwahrscheinlich werden[3]. Jedoch würde dann die Größe von Routingprotokoll-Paketen erheblich ansteigen und damit unnötig viel Energie und Bandbreite verbraucht. In dieser Hinsicht sind kurze Adressen zu bevorzugen. Zu kurze Adresse können jedoch viele Adresskonflikte als Folge haben, die erkannt und aufgelöst werden müssen und die zu kurzzeitigen Unterbrechungen der Kommunikation führen können. Abbildung 2 stellt den Zusammenhang von Konfliktwahrscheinlichkeit, Adresslänge und Anzahl Endgeräte im Netz dar.

Je nach Anwendungsfall können sehr kurze Adressen oder etwas längere Adressen sinn-

[2] Genauer gesagt wird die Adresse nicht dem Gerät, sondern der Netzwerkschnittstelle zugewiesen. Hier wird vereinfachend davon ausgegangen, dass jedes Endgerät nur eine Netzwerkschnittstelle besitzt.

[3] Selbst bei IPv6 werden zustandslos gewählte Adresse wegen der Gefahr eines Adresskonflikts auf Eindeutigkeit im Netz geprüft.

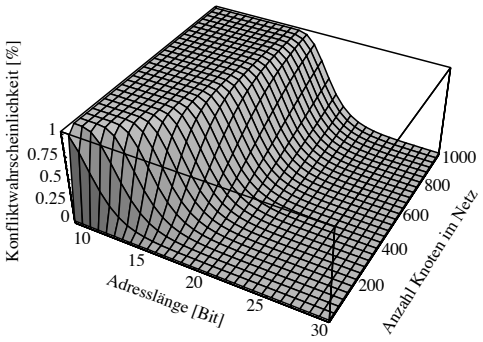


Abbildung 2: Konfliktwahrscheinlichkeit bei zufälliger Wahl der Adressen in Abhängigkeit der Adresslänge und der Anzahl Endgeräte bzw. Knoten im Netz

voll sein. Daher haben die von PACMAN zugewiesenen Adressen eine variable Länge. Die Zuweisung erfolgt nach einem probabilistischen Algorithmus, dem eine maximale Konfliktwahrscheinlichkeit vorgegeben werden kann. Diese kann je nach Anwendungsfall gewählt werden. Je größer die vorgegebene Konfliktwahrscheinlichkeit, desto höher ist die Effizienz der Adressierung, d.h. die zugewiesene Adresse ist "kürzer". Eine geringe Konfliktwahrscheinlichkeit hingegen führt zu weniger effizienter Adressierung. Die Kompatibilität zu IP wird dadurch erreicht, dass die IP-Adressen in ausgehenden Routingprotokollpaketen durch eine weitere Komponente komprimiert werden und damit nur unterhalb der Vermittlungsschicht und auf dem Funkmedium eine variable Länge besitzen. Eingehende Routingprotokollpakete werden entsprechend dekomprimiert, so dass die Adressen oberhalb der Vermittlungsschicht dem Format herkömmlicher IP-Adressen entsprechen.

Um Konflikte bei der Zuweisung zu minimieren, verwaltet jedes Endgerät Zustände über bereits zugewiesene bzw. belegte Adressen in Form einer Belegungstabelle. Diese wird nicht aktiv mit anderen Endgeräten synchronisiert, sondern passiv mit Informationen des Routingprotokolls gewartet. Ein noch nicht konfiguriertes Gerät kann die Belegungstabelle eines benachbarten Geräts anfordern, um eine im Netz eindeutige Adresse zu wählen. Konflikte treten dadurch i.A. nur bei gleichzeitiger Konfiguration mehrerer Geräte oder nach Netzverschmelzungen auf. Diese werden mit einem neuartigen Verfahren erkannt, welches als Erstes dieser Art passiv, also ohne zusätzliche Signalisierung arbeitet. Dabei wird eine Kombination unterschiedlicher Algorithmen verwendet, die eine Erkennung von Anomalien bzw. inkonsistenten Zuständen der Routingprotokoll-Instanzen als Folge von Adresskonflikten erkennen kann. Die auf der Vermittlungsschicht angesiedelte PACMAN-Instanz verwendet also Informationen der Routingprotokollinstanz, welche bei vielen Routingprotokollen auf höheren Schichten angesiedelt ist.

Die gewonnene Effizienz wird allerdings mit einer erhöhten Komplexität durch neue Abhängigkeiten erkauft. In diesem Fall entsteht eine Abhängigkeit des Autokonfigurationsprotokolls vom Routingprotokoll. PACMAN kann jedoch die Notwendigkeit einer Modifikation des Routingprotokolls vermeiden, da es den Kontrollverkehr dieser Protokolle lediglich analysiert, aber nicht verändert. Um diesen interpretieren zu können, wird für jedes zu unterstützende Protokoll ein Parsermodul bereitgestellt, welches die Protokollpake-

te in ein generisches Format überführt. Dieser modulare Ansatz erlaubt eine weitgehende Unabhängigkeit vom Routingprotokoll und vereinfacht die Integration neuer Routingprotokolle.

Wird ein Konflikt erkannt, wird durch eine weitere Komponente entweder die eigene Adresse gewechselt, falls diese betroffen ist, oder das Endgerät mit der doppelten Adresse benachrichtigt, so dass dieses seine Adresse wechseln kann. Da viele Transportprotokolle, wie z.B. TCP, Verbindungen u.a. durch die Adressen der Endpunkte definieren, brechen aktive Transportverbindungen i.A. ohne weitere Maßnahmen nach einem Adresswechsel ab. Die Komponente zur Behandlung von Adresswechsel verhindert dies durch einen Ende-zu-Ende Ansatz.

Eine optionale Optimierung ist die passive Adressauflösung. Normalerweise sind die Protokolle ARP bzw. NDP dafür zuständig, IPv4- bzw. IPv6- auf MAC-Adressen abzubilden. PACMAN erlaubt die Unterstützung von ARP bzw. NDP mit passiven Methoden, wodurch das Senden von ARP- bzw- NDP-Anfragen überflüssig gemacht und damit der Kommunikationsaufwand weiter reduziert werden kann.

Derzeit in der IETF diskutierte Routing-Verfahren für MANETs verwenden kein hierarchisches Routing und sind damit eher für kleine Netze geeignet. Es wurden jedoch auch Protokolle für das Routing in größeren MANETs vorgeschlagen, die z.T. einen hierarchischen Adressraum voraussetzen. Für solche Protokolle erlaubt die Clustering-Komponente von PACMAN die Bildung einer hierarchischen Adressstruktur. In diesem Fall dient die Adresse nicht nur der Identifizierung eines Geräts, sondern auch dessen Lokalisation. Deshalb muss diese Hierarchie nicht nur automatisch aufgebaut, sondern auch ständig der sich ändernden Topologie angepasst werden. Eine Minimierung des Protokoll-Overheads stellt dabei wieder ein wichtiges Entwurfsziel dar. Das in der Arbeit vorgestellte Verfahren ist das erste reaktive bzw. bedarfsorientierte multi-hop Clustering-Verfahren und ist dadurch sehr ressourcensparend, denn die Struktur wird nur zu dem Zeitpunkt und an dem Ort gewartet, wann und wo sie benötigt wird. Um die Wartung möglichst effizient zu gestalten, werden wieder passive Methoden angewendet und Informationen anderer Protokolle und Schichten ausgenutzt. Des Weiteren werden Endgeräte, die sich in Gruppen bewegen, als solche erkannt und zu einem Cluster zusammengefasst. Dadurch sind die Cluster vergleichsweise stabil und die Anzahl der Adresswechsel gering.

3 Passive Erkennung von Konflikten

Adresskonflikte können z.B. nach Netzverschmelzungen auftreten und müssen möglichst schnell erkannt und aufgelöst werden, da sonst eine Weiterleitung von Datenpaketen an ”falsche” Endgeräte und damit eine Unterbrechung der Kommunikation die Folge sein kann. Das in der Arbeit vorgestellte Verfahren ermöglicht erstmals eine passive Erkennung doppelter Adressen und wird auch *Passive Duplicate Address Detection (PDAD)* [Wen03] genannt. Im Gegensatz zur aktiven Erkennung werden bei der passiven Erkennung keine Kontrollnachrichten versendet, so dass kein zusätzlicher Protokoll-Overhead verursacht wird.

Die grundlegende Idee des neuartigen Ansatzes ist es, durch Erkennung von Anomalien im Kontrollverkehr anderer Protokolle Hinweise auf Adresskonflikte zu erhalten. PACMAN verwendet zu diesem Zweck den Kontrollverkehr des Routingprotokolls. Dazu muss mindestens ein Endgerät im Netz Routinginformationen von mindestens zwei in den Konflikt involvierten Endgeräten erhalten, um Anomalien erkennen zu können. Im Fall proaktiver Routingprotokolle ist eine permanente Erkennung aller doppelten Adressen im Netz prinzipiell möglich. Im Fall reaktiver Routingprotokolle können zumindest Konflikte zwischen den Endgeräten erkannt werden, die in den Aufbau bzw. die Wartung von Routen involviert sind. Alle anderen Endgeräte sind nicht aktiv an der Weiterleitung beteiligt. Adresskonflikte dieser Endgeräte sind für das Netz zu diesem Zeitpunkt daher nur von untergeordneter Bedeutung.

Eine Anomalie ist eine Abweichung vom Normalverhalten, in diesem Fall also Ereignisse, welche entweder nie bei eindeutigen Adressen im Netz, aber immer bei doppelten Adressen oder selten bei eindeutigen Adressen, aber oft bei doppelten Adressen auftreten. Im ersten Fall ist eine Anomalie eindeutig und die Komponente zur Konfliktauflösung kann benachrichtigt werden, sobald ein entsprechender Hinweis vorliegt. Im zweiten Fall kann nur eine Wahrscheinlichkeit bestimmt werden, mit der ein Konflikt besteht. Eine Entscheidung kann erst nach weiteren Hinweisen erfolgen.

Zur Entwicklung entsprechender Algorithmen wurden existierende Routingprotokolle analysiert, klassifiziert und Modelle für die grundlegenden Mechanismen definiert. Anhand dieser wurden dann durch manuelle Inspektion 15 Algorithmen zur Anomalieerkennung entwickelt. Diese PDAD-Algorithmen, in implementierter Form PDAD-Module genannt, können je nach Eigenschaften des verwendeten Routingprotokolls unterschiedlich kombiniert und konfiguriert werden. Die grundlegende Idee ist, einen aus empfangenen Routinginformationen abgeleiteten Zustand der Routingprotokollinstanz eines Endgeräts mit dem Zustand der eigenen Routingprotokollinstanz zu vergleichen, um Konflikte der eigenen Adresse zu erkennen, oder diesen mit dem letzten bekannten Zustand der Instanzen eines weiteren Endgeräten zu vergleichen, um Konflikte fremder Adressen zu erkennen.

Da das vorliegende System ein dynamisches ist, ist ein Zustand nur in Verbindung mit einem Zeitpunkt aussagekräftig. In Bezug auf die PDAD-Algorithmen ist der Sendezeitpunkt eines Routingprotokollpakets von entscheidender Bedeutung. Empfängt ein Endgerät ein Routingprotokollpaket eines anderen Endgeräts, kennt dieser den Sendezeitpunkt des Pakets nicht. Der Sender könnte den Sendezeitpunkt zwar mit dem Routingprotokollpaket verbreiten, damit der Empfänger mit diesem etwas anfangen kann wäre aber ein gemeinsamer Bezugspunkt, also synchronisierte Uhren, erforderlich. Da dies nur mit hohem Aufwand oder zusätzlichem Gerät, wie z.B. einem GPS-Empfänger, möglich ist, wurde dieser Ansatz nicht verfolgt. Stattdessen wird auf Basis des Empfangszeitpunktes der Sendezeitpunkt mit hinreichender Genauigkeit geschätzt. Dazu wird angenommen, dass die maximale Zeit t_d, die eine bestimmte Routinginformation im Netz verbleibt, bei endlicher Anzahl von Endgeräten endlich ist und abgeschätzt werden kann. Ist dies der Fall, gilt $t_e - t_s < t_d$ mit Sendezeitpunkt t_s und Empfangszeitpunkt t_e. t_d ist dabei u.a. von der Länge der Warteschlangen in den Netzwerkschnittstellen der Endgeräte und der Dauer des Medienzugriffs abhängig. Beides ist i.A. begrenzt. Bei IEEE 802.11 ist der Medienzugriff für ein Paket beispielsweise standardmäßig auf ca. 500ms begrenzt.

Damit die Ausbreitungsdauer der Routinginformationen im Netz endlich ist, muss das Routingprotokoll zwei Anforderungen erfüllen: Die Routinginformationen dürfen nur abzählbar häufig im Netz weitergeleitet werden (endliche Weiterleitung) und die Zustände nicht unendlich lange in Endgeräten gehalten werden (Soft-State). Beide Anforderungen werden von den bekannten Routingprotokollen erfüllt. Wäre dies nicht der Fall, würden diese unnötig viel Ressourcen verbrauchen und das Routingprotokoll könnte nach Sequenznummerüberläufen nicht zwischen alten und neuen Routinginformationen unterscheiden, wodurch permanente Routingschleifen und damit lange andauernde Kommunikationsunterbrechungen entstehen können.

Die PDAD-Komponente ist selbst wieder aus mehreren funktionalen Komponenten aufgebaut. Wird ein Routingprotokollpaket empfangen, wird je nach Klasse des Routingprotokolls entweder die proaktive oder die reaktive Komponente eingesetzt. Diese speichert die benötigten Informationen aus dem Paket in der *Routingprotokollpaket (RP)-Tabelle*. Je nach Konfiguration, die vom verwendeten Routingprotokoll abhängt, werden unterschiedliche PDAD-Module aufgerufen. Liefern diese einen Hinweise auf einen Konflikt, wird die Konfliktwahrscheinlichkeit in eine *Konfliktwahrscheinlichkeit (KW)-Tabelle* eingetragen. Diese wird vor allem für die probabilistischen Algorithmen benötigt, die eine Entscheidung erst nach mehreren Hinweisen treffen können. Dazu wird die Konfliktwahrscheinlichkeit über mehrere Pakete geglättet und erst bei Überschreitung einer bestimmten Wahrscheinlichkeit wird die Komponente zur Konfliktauflösung benachrichtigt.

Zur Veranschaulichung wird im Folgenden ein einfacher Basis-Algorithmus vorgestellt: Der PDAD-NH-Algorithmus ist auf die Klasse der Link-State-Routingprotokolle anwendbar und nutzt die Zustandshaltung bei Verwendung bidirektionaler Links aus. Die meisten Link-State-Routingprotokolle, wie z.B. OLSR [CJ03], deklarieren nur bidirektionale Link-States oder kennzeichnen diese im Routingprotokollpaket als solche. Eine bidirektionale Nachbarschaftsbeziehung besteht nur, wenn sich beide Endgeräte gegenseitig "hören". Der Zustand bezüglich der Nachbarschaftsbeziehung besteht also bei beiden Endgeräten gleichermaßen. Ein Endgerät A, das eine Routinginformation mit einem bidirektionalen Link zu der eigenen Adresse empfängt, muss daher im Fall eindeutiger Adressen innerhalb der zurückliegenden Zeitspanne t_d ein Nachbar von dem Endgerät gewesen sein, das die Routinginformation verbreitet hat. Ist dies nicht der Fall, wurde die Routinginformation von einem anderen Endgerät mit der gleichen Adresse gesendet und ein Konflikt der Adresse von Endgerät A liegt vor. Um dies zu prüfen, muss jedes Endgerät seine Nachbarn, die es in der zurückliegenden Zeitspanne t_d hatte, in einer *Nachbarschafts (NH)-Tabelle* T_{nh} speichern.

Abbildung 3 zeigt einen Beispielsgraph. Ein ausgefüllter Kreis stellt ein Endgerät bzw. einen Knoten, ein gestrichelter Kreis den dazugehörigen Senderadius und eine Zahl die Adresse des Knotens dar. Knoten A und E besitzen die gleiche Adresse 1. Knoten F ist Nachbar von Knoten E und sendet eine Routinginformation RI, die bidirektionale Link-States LS zu Knoten mit den Adressen 1 und 5 anzeigt. Die Ursprungsadresse OA von Knoten F ist 6, die Sequenznummer SN der Nachricht 9. Nachdem Knoten A diese RI empfängt, kann dieser seine eigene Adresse 1 als doppelt erkennen, da diese in der Liste der Link-States der RI, die Ursprungsadresse 6 aber nicht in der Nachbarschaftstabelle enthalten ist.

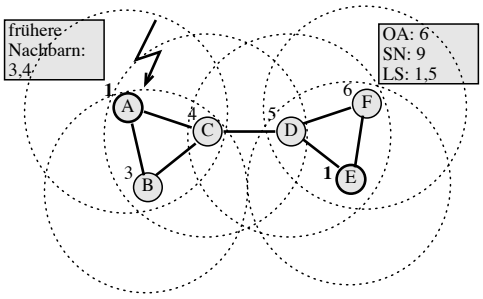


Abbildung 3: Beispielszenario einer Konflikterkennung mit dem PDAD-NH-Algorithmus

Der PDAD-NH-Algorithmus basiert auf der Annahme, dass die Endgeräte mit der gleichen Adresse unterschiedliche Link-States besitzen. Wenn im obigen Beispiel Knoten B oder C ebenfalls Adresse 6 hätten, hätte der Konflikt durch dieses Routingprotokoll-Paket und mit diesem Algorithmus alleine nicht erkannt werden können. Einige Routingprotokolle wie OLSR verwenden optimiertes Fluten und reduzierte Link-State-Listen, um Bandbreite einzusparen. Auch in diesem Fall wird eine Kombination mit weiteren Algorithmen erforderlich.

Die entwickelten Basis-PDAD-Algorithmen wurden auf Anwendbarkeit auf die drei bekannten Routingprotokolle OLSR [CJ03], AODV [PBRD03] und FSR [GHP02] untersucht. Dabei konnte gezeigt werden, dass jeweils eine Kombination der entwickelten Algorithmen existiert, die alle Konflikte erkennen kann, solange zumindest ein Endgerät im Netz eine eindeutige Adresse besitzt.

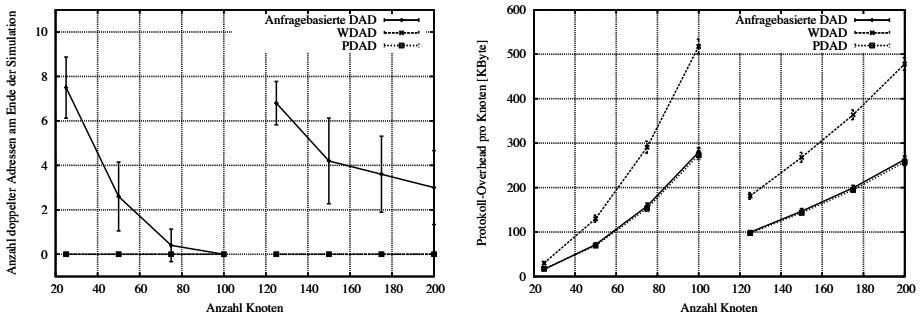
4 Evaluation in Simulator und Testsystem

Die Funktions- und Leistungsfähigkeit der einzelnen Komponenten und von PACMAN insgesamt wurde sowohl in einer Simulationsumgebung als auch in einem Testsystem untersucht und mit der von anderen Ansätzen verglichen. Dabei hat sich gezeigt, dass PACMAN sowohl deutlich effizienter als auch robuster als bisherige, in der Literatur beschriebene Ansätze ist.

Das PDAD-Verfahren von PACMAN wurde u.a. mit dem Anfrage-basierten DAD- und dem Weak DAD-Verfahren durch diskrete, Ereignis-basierte Simulationen verglichen. Zunächst wurden jeweils fünf zufällig ausgewählten Paaren von Endgeräten zu Beginn der Simulation die gleiche Adresse zugeteilt. Diese zehn doppelten Adressen sollten dann durch das Verfahren erkannt werden. Alle Versuche wurden 20 mal in unterschiedlichen Szenarien durchgeführt. Ein Ergebnis der Untersuchungen war, dass das Anfrage-basierte Verfahren nicht immer alle Konflikte erkennen konnte. Weak DAD und PDAD konnten hingegen immer alle Konflikte erkennen (siehe Abbildung 4(a)).

Abbildung 4(b) zeigt den von den einzelnen Verfahren während der Simulationszeit erzeugten Protokoll-Overhead inklusive des Overheads des Routingprotokolls FSR. Es ist

zu erkennen, dass Weak DAD den Protokoll-Overhead des Routingprotokolls signifikant erhöht. Die Anfrage-basierte DAD hingegen erzeugen nur wenig, PDAD gar keinen Overhead. Lediglich die Benachrichtigung anderer Endgeräte über einen Adresskonflikt trägt hier zum Protokoll-Overhead bei.



(a) Anzahl erkannter doppelter Adressen verschiedener DAD-Ansätze

(b) Protokoll-Overhead verschiedener DAD-Ansätze (inklusive Routingprotokoll-Overhead)

PACMAN wurde außerdem in Linux implementiert[4] und in einem WLAN-basierten MANET bestehend aus bis zu 14 HP iPAQ Pocket PCs untersucht. Dabei wurden frei verfügbare Implementierungen der Routingprotokolle OLSR und FSR verwendet. Aufgrund des passiven Ansatzes von PACMAN waren Änderungen an den Routingprotokollimplementierungen nicht notwendig. Abbildung 4 zeigt den Aufbau des Testsystems als Demonstrator, der u.a. auch auf der ACM Mobicom 2004 vorgestellt wurde. Zu Demonstrations- und Testzwecken kann eine für diesen Zweck entwickelte Software namens "Wireless Network Topology Emulator (WNTE)"[5] mit Hilfe von MAC-Filtern eine Multi-hop-Topologie emulieren. Die gewünschte Topologie des MANET kann auf einem Laptop graphisch dargestellt und eingestellt werden. Damit konnte die Funktions- und Leistungsfähigkeit von PACMAN auch in einem realen System untersucht und bestätigt werden.

Abbildung 4: Foto des Testsystems (WLAN-basiertes MANET mit sechs Pocket PCs)

[4]PACMAN für Linux steht unter http://pacman-autoconf.sourceforge.net zum Download bereit
[5]WNTE für Linux steht unter http://wnte.sourceforge.net zum Download bereit

Literatur

[CCL03] I. Chlamtac, M. Conti und J.-N. Liu. Mobile ad hoc networking: imperatives and challenges. *Elsevier Ad Hoc Networks*, 1:13–64, Januar 2003.

[CJ03] T. Clausen und P. Jaqcuet. Optimized Link State Routing Protocol (OLSR). RFC 3626, Oktober 2003.

[GHP02] M. Gerla, X. Hong und G. Pei. Fisheye State Routing Protocol (FSR) for Ad Hoc Networks. IETF Draft, Juni 2002.

[PBRD03] C. Perkins, E. Belding-Royer und S. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561, Juli 2003.

[Wen03] K. Weniger. Passive Duplicate Address Detection in Mobile Ad Hoc Networks. In *Proc. of IEEE Wireless Communications and Networking Conference (WCNC) 2003*, New Orleans, USA, Marz 2003.

[Wen04] K. Weniger. *IP-Autokonfiguration in mobilen Ad-hoc-Netzwerken*. Shaker Verlag, Aachen, Germany, September 2004. ISBN: 3-8322-3167-6.

[Wen05] Kilian Weniger. PACMAN: Passive Autoconfiguration for Mobile Ad hoc Networks. *IEEE Journal on Selected Areas of Communications (JSAC) Special Issue on 'Wireless Ad Hoc Networks'*, Marz 2005.

[WZ04] Kilian Weniger und Martina Zitterbart. Address Autoconfiguration in Mobile Ad Hoc Networks: Current Approaches and Future Directions. *IEEE Network Magazine Special issue on 'Ad hoc networking: data communications & topology control'*, Juli 2004.

[ZWS03] M. Zitterbart, K. Weniger und O. Stanze. IPonAir - Drahtloses Internet der nächsten Generation. *Praxis der Informationsverarbeitung und Kommunikation (PIK) Themenheft 'Mobile Ad-hoc-Netzwerke'*, Dezember 2003.

Kilian Weniger erhielt nach seinem Studium der Elektrotechnik/Nachrichtentechnik im Jahr 2000 den Abschluss Dipl.-Ing. von der TU Braunschweig. In den Jahren 2000 und 2001 war er am Daimler-Chrysler-Forschungszentrum in Palo Alto, Kalifornien, tätig und beschäftigte sich mit der drahtlosen Datenkommunikation von und zu Fahrzeugen. Anschließend war er wissenschaftlicher Mitarbeiter an dem von Prof. Zitterbart geleiteten Institut für Telematik der Universität Karlsruhe (TH) und war dort u.a. maßgeblich am BMBF-Projekt IPonAir beteiligt. Im Juli 2004 promovierte er mit Auszeichnung und erhielt den Grad Dr.-Ing. verliehen. Seine Dissertation wurde mehrfach ausgezeichnet, u.a. mit dem Förderpreis für Natur- und Ingenieurwissenschaften der Vodafone-Stiftung für Forschung. Seit November 2004 ist er am europäischen Forschungszentrum der Firma Panasonic tätig.