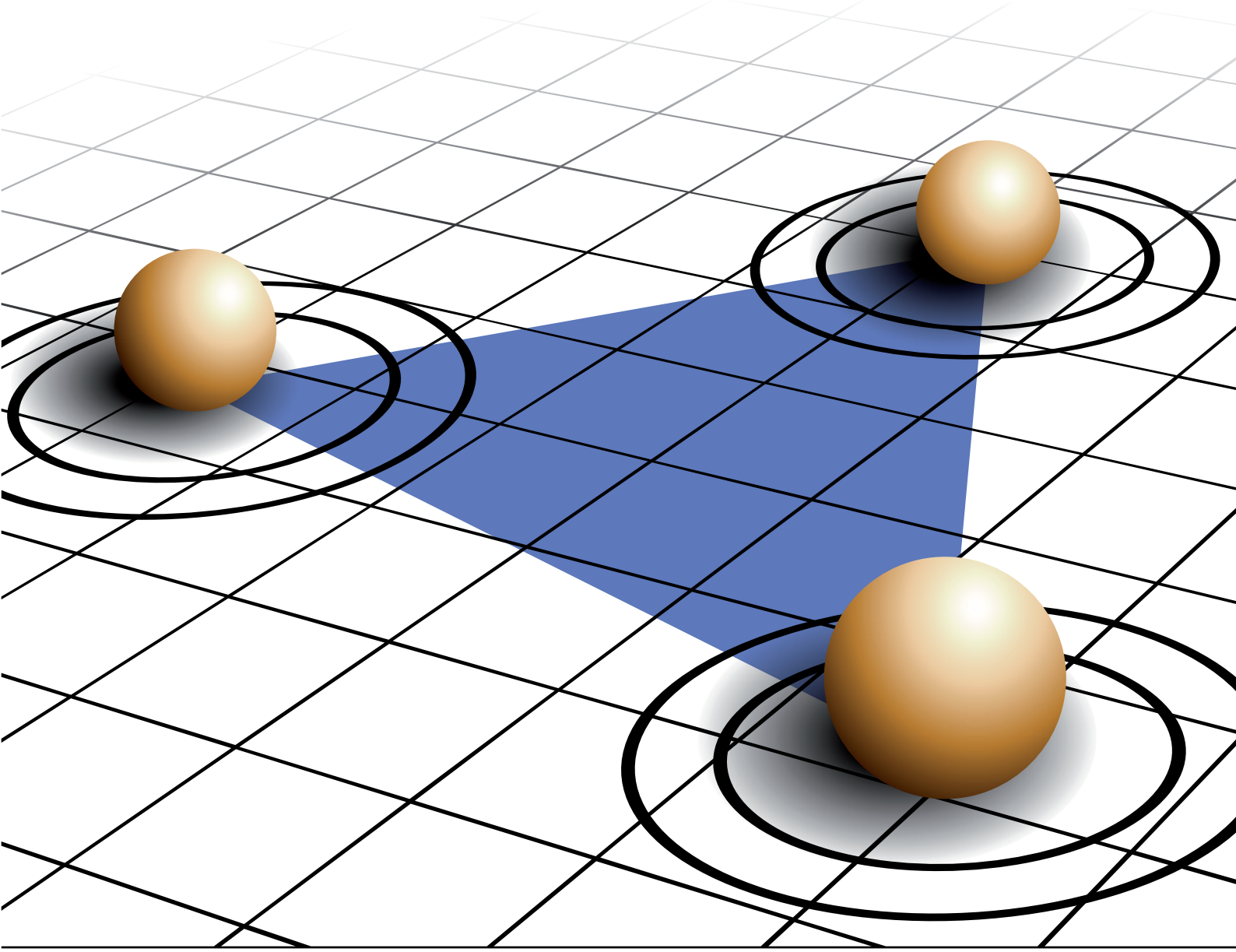
Martina Zitterbart, Peter Baumung (Hrsg.)

Basissoftware für drahtlose Ad-hoc- und Sensornetze



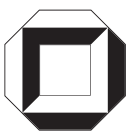
universitätsverlag karlsruhe

Martina Zitterbart, Peter Baumung (Hrsg.)

Basissoftware für drahtlose Ad-hoc- und Sensornetze

Basissoftware für drahtlose Ad-hoc- und Sensornetze

Martina Zitterbart
Peter Baumung
(Hrsg.)



universitätsverlag karlsruhe

Impressum

Universitätsverlag Karlsruhe
c/o Universitätsbibliothek
Straße am Forum 2
D-76131 Karlsruhe
www.uvka.de

Dieses Werk ist unter folgender Creative Commons-Lizenz
lizenziert: http://creativecommons.org/licenses/by-nc-nd/2.0/de/

Universitätsverlag Karlsruhe 2009
Print on Demand

ISBN: 978-3-86644-309-9

Vorwort

Mobilen Systemen gehört die Zukunft!

Die Mobilkommunikation ist von außerordentlicher Bedeutung für eine umfassende und allgegenwärtige Dienste- und Kommunikationsinfrastruktur wie sie von der globalen Informationsgesellschaft zunehmend gefordert wird. Unter diesem Motto wurde das DFG-Schwerpunktprogramm 1140 im Jahre 2001 konzipiert. Der Fokus lag dabei nicht auf "traditionellen" Mobilkommunikationsnetzen und deren drahtlosem Zugang, sondern auf dem zukunftsweisenden Bereich selbst organisierender, drahtloser Netze mit hochgradig mobilen Systemen und Diensten. Zu diesen Netzen zählen mobile Ad-hoc- und drahtlose Sensornetze, die von großer Bedeutung sind, etwa zur Bereitstellung von Netzinfrastruktur in Krisen- und Katastrophenfällen oder zur umfassenden Beobachtung von Naturphänomenen.

Im Mittelpunkt des Interesses standen neue Methoden und Verfahren zur Bereitstellung geeigneter *Basisdienste* in derart hoch mobilen, dezentralen und selbst organisierenden Netzen. Dabei enstanden sind u.a. innovative Konzepte zum Routing (z.B. mittels Adress- oder Geo-basierter Verfahren), zur Realisierung von Gruppendiensten, zum Finden und Verwalten von Diensten bis hin zur Selbstkonfiguration adaptiver Anwendungen und zur Nutzung Schichten-übergreifender Information. Im Kontext drahtloser Sensornetze wurden ebenfalls neuartige Methoden und Verfahren konzipiert, die verschiedene Aspekte von Middleware für drahtlose Sensornetze adressieren, wie etwa das kooperative Sammeln von Daten oder die Synchronisation dynamisch gebildeter Mengen von Sensorknoten. Durch Nutzung von Synergien einzelner Projekte des Schwerpunktprogramms konnten erarbeitete Basisdienste gezielt miteinander kombiniert werden, so dass hieraus neue Ansätze zur Bereitstellung komplexerer, *höherwertiger Dienste*, wie beispielsweise Informations-, Auskunfts- oder Koordinationsdienste, entstehen konnten.

Die insgesamt aus dem Schwerpunktprogramm hervorgegangene *Basissoftware* bildet ein umfangreiches und solides Fundament, um die Entwicklung und Etablierung von Anwendungsgebieten in Ad-hoc- und Sensornetzen maßgeblich zu stützen und zu fördern. Die Tauglichkeit der Konzepte hierfür konnte bereits innerhalb des Schwerpunktprogramms durch die massive Entwicklung Projekt-interner und -übergreifender, prototypischer Anwendungen und Demonstratoren bestätigt und gestützt werden. Mit Einzelbeiträgen der geförderten Projekte gibt der vorliegende Sammelband einen Einblick in die jeweils erzielten Ergebnisse.

Peter Baumung & Martina Zitterbart

Projekte aus dem Bereich mobiler Ad-hoc-Netze:

Otto Spaniol und Stefan Penz
Lehrstuhl für Informatik 4, RWTH Aachen

Ad-hoc Service Management für selbstorganisierende vernetzte mobile Systeme 1

Björn Scheuermann, Christian Lochert und Martin Mauve
Institut für Informatik, Heinrich-Heine-Universität Düsseldorf

Kooperative schichtenübergreifende Überlastkontrolle für mobile Ad-Hoc-Netzwerke 17

Peter Baumung und Martina Zitterbart
Institut für Telematik, Universität Karlsruhe (TH)

MAMAS – Mobility-aware Multicast for Ad-hoc Groups in Self-organizing Networks 33

Sascha Schnaufer, Matthias Transier und Wolfgang Effelsberg
Lehrstuhl für Praktische Informatik IV, Universität Mannheim

PBM – Positionsbasierter Multicast für mobile Ad-Hoc-Netze 49

Stephan Schuhmann, Klaus Herrmann und Kurt Rothermel
Institut für Parallele und Verteilte Systeme, Universität Stuttgart

Selbstkonfiguration adaptiver Anwendungen in ubiquitären Systemen 67

Projekte aus dem Bereich drahtloser Sensornetze:

Reinhardt Karnapke und Jörg Nolte
Brandenburgische Technische Universität Cottbus

COCOS – ***Coordinated Communicating Sensors*** 85

Jan Blumenthal und Dirk Timmermann
Institut für Angewandte Mikroelektronik und Datentechnik, Universität Rostock

Middleware für mobile spontan vernetzte Sensornetzwerke 97

Carsten Buschmann[1], Stefan Fischer[1], Norbert Luttenberger[2] und Jochen Koberstein[2]
[1] *Institut für Telematik, Universität zu Lübeck*
[2] *AG Kommunikationssysteme, Christian-Albrechts-Universität zu Kiel*

SWARMS – Software Architecture for Radio-Based Mobile Self-Organizing Systems 109

Entwicklung eines Ad-hoc Service Managements für selbstorganisierende vernetzte mobile Systeme

Otto Spaniol und Stefan Penz

Lehrstuhl für Informatik 4, RWTH Aachen

Zusammenfassung – Die spontane Vernetzung mobiler Endgeräte zu einem drahtlosen Ad-hoc-Netzwerk erfordert neuartige Mechanismen zur selbstorganisierenden und konfigurationsfreien Vermittlung von Dienstangeboten. Im Rahmen des Projekts wurde daher eine Dienstvermittlungsplattform entwickelt, mit deren Hilfe Benutzerapplikationen umfassende Informationen über verfügbare Dienstangebote erfassen können. Hierzu zählen nicht nur die funktionalen und nicht-funktionalen Eigenschaften der jeweiligen Dienstanbieter, sondern zusätzlich auch relevante Qualitätsparameter der zugehörigen Netzwerkverbindungen sowie strukturelle Informationen über die zu Grunde liegende Netzwerktopologie. Bei der Konzeption der Teilsysteme wurde besonderen Wert auf deren praktische Umsetzbarkeit und die Verkehrseffizienz gelegt.

1. Ziele des Gesamtprojekts

Mobile Ad-hoc-Netzwerke entstehen durch den spontanen Zusammenschluss mobiler Endgeräte über eine Funkschnittstelle. Diese Netzwerke verfügen im Allgemeinen über keine verwaltete Infrastruktur, so dass Dienste, die in stationären Netzwerken von dedizierten Servern mit allgemein bekannten Adressen angeboten werden, von den mobilen Geräten übernommen werden müssen. Mobile Geräte können jedoch jederzeit dem Netzwerk beitreten oder dieses auch wieder verlassen. Diese Netzwerkdynamik erfordert Dienstvermittlungssysteme, mit deren Hilfe aktuell verfügbare Dienstanbieter und ihre Merkmalen ermittelt werden können.

Da die Erreichbarkeit von Dienstanbietern in dynamischen Netzwerken nicht dauerhaft garantiert werden kann, sollte dem Benutzer die Möglichkeit gegeben werden, bei Verbindungsabbruch zu einem Alternativanbieter zu wechseln. Diese Anbieterwechsel sollten vom Dienstvermittlungssystem aktiv unterstützt werden, um eine durchgängige Dienstnutzung zu gewährleisten.

Die hohe topologische Dynamik mobiler Ad-hoc-Netzwerke beeinflusst zudem die Übertragungseigenschaften auf praktisch allen Protokollschichten. Dies führt unter anderem dazu, dass die Übertragungsqualität starken räumlichen und zeitlichen Schwankungen unterlegen ist. Die Netzwerkverbindungen zu einzelnen

Dienstanbietern können daher sehr unterschiedliche Eigenschaften aufweisen. Diese Eigenschaften sollten daher bei der Auswahl der Dienstanbieter berücksichtigt werden können. In mobilen Funknetzwerken ist insbesondere die Stabilität einer Kommunikationsverbindung ein entscheidendes Qualitätskriterium, um eine langfristige Dienstnutzung zu gewährleisten. Zudem sind die Übertragungskapazitäten in diesen Netzwerken vergleichsweise gering, so dass die verfügbare Übertragungsbandbreite zwischen Dienstnutzer- und Dienstanbieter ein weiteres wichtiges Qualitätsmerkmal darstellt.

Um die knappen Übertragungskapazitäten nicht zusätzlich signifikant zu belasten, sollte auch das Dienstvermittlungssystem selbst nur eine möglichst geringe Verkehrslast erzeugen. Neben der Verkehrseffizienz war die Robustheit des Systems gegen Knotenausfälle und Verbindungsabbrüche das wichtigste Optimierungsziel.

Die Projektlaufzeit gliederte sich in drei Bewilligungsphasen, die im Folgenden näher beschrieben werden. Die erste Projektphase konzentrierte sich auf die eigentlichen Vermittlungsmechanismen und auf Verwaltungskonzepte für dynamische Dienstinformationen. In der zweiten Projektphase wurden vor allem effiziente Messverfahren für Dienstgüteparameter entworfen und in die Vermittlungsplattform integriert. Der Schwerpunkt der dritten Projektphase lag in der Entwicklung eines passiven Erfassungsverfahrens, das stabile Teilstrukturen in ansonsten dynamischen Netzwerken zu identifizieren versucht. Diese Kontextinformationen dienen vor allem einer Optimierung der Anbieterauswahl, aber auch Routing-Protokolle können von diesen Informationen profitieren.

2. Verwaltung dynamischer Dienstinformationen

Schon in herkömmlichen, kabelgebundenen Netzwerken werden Dienstvermittlungsmechanismen benötigt, um Netzwerkdienste (z.B. Dateiserver oder Telefonie-Gateways) konfigurationsfrei ermitteln zu können. In den letzten Jahren wurde zu diesem Zweck eine Vielzahl von Systemen entwickelt, die zwar zum Teil auf sehr ähnlichen Grundkonzepten beruhen, sich jedoch in vielen Details voneinander unterscheiden.

Daher wurden die wichtigsten dieser Vermittlungssysteme zu Beginn der ersten Projektphase eingehend qualitativ untersucht und auf ihre Verwendbarkeit in mobilen Ad-hoc-Netzwerken hin analysiert. Dabei zeigte sich, dass das *Service Location Protocol SLP*[4] viele Eigenschaften hat, die es als Basiskomponente für die Dienstvermittlung in mobilen Ad-hoc-Netzwerken prädestinieren.

Die grundlegenden Vermittlungsmechanismen von SLP werden in Abbildung 1 dargestellt. Dabei unterstützt SLP zwei Modi. Die Wahl des Modus hängt von der Existenz eines so genannten *Directory-Agents* ab, der die Informationsver-

(a) Zentrale Dienstvermittlung mit Directory-Agent

(b) Dezentrale Dienstvermittlung über Multicast

Abb. 1: SLP Dienstsuchemodi

mittlung zwischen Dienstnutzer (*User-Agent*) und Dienstanbieter (*Service-Agent*) übernehmen kann. Existiert eine solche Vermittlungsinstanz, registriert der Service-Agent seine Dienstangebote beim Directory-Agent, der diese Registrierung bestätigt. Anschließend kann der User-Agent die benötigten Dienstinformationen direkt beim Directory-Agent erfragen.

Der Vorteil dieses Vermittlungsmodus liegt in der hohen Verkehrseffizienz. Während des gesamten Vermittlungsprozesses finden nur wenige Unicast-Übertragungen statt, so dass die Verkehrslast äußerst gering ist. Gerade beim Einsatz in Ad-hoc-Netzwerken ist bei solchen zentralisierten Verfahren jedoch problematisch, dass bei einem Ausfall des Directory-Agents keine Vermittlung mehr stattfinden kann. SLP begegnet diesem Problem auf zwei Wegen: Zum einen erlaubt es den redundanten Einsatz mehrerer Directory Agents in einem Netzwerk und zum anderen bietet es bei Ausfall des letzten Directory-Agent einen dezentralen Vermittlungsmodus, der in Abbildung 1(b) dargestellt wird.

Grundsätzlich bilden alle Service-Agents eine Multicast-Gruppe mit vordefinierter Gruppenadresse. Beim dezentralen Vermittlungsmodus senden die User-Agents ihre Anfragen per Multicast direkt an die Gruppe der Service-Agents. Dienstanbieter, die einen gewünschten Dienst zur Verfügung stellen, antworten auf diese Anfrage mit einer Unicast-Antwort. Die Multicast-Anfrage erzeugt abhängig vom verwendeten Routing-Protokoll zwar eine deutlich höhere Verkehrslast, sie ist allerdings auch sehr robust gegen Knoten- und Verbindungsausfälle. SLP kann dynamisch zwischen den beiden Vermittlungsmodi wechseln und so die Robustheit des dezentralen Modus mit der Effizienz des zentralen Modus verbinden. Darüber hinaus sorgen viele weitere Details wie eine flexible Anfragesprache und frei verfügbare Implementierungen dafür, dass SLP als Grundlage für das Dienstvermittlungssystem gewählt wurde.

Auch wenn die SLP-Vermittlungsmechanismen für Ad-hoc-Netzwerke geeignet erscheinen, erfordert die hohe Dynamik solcher Netzwerke tiefgreifende Anpassun-

gen und Erweiterungen. Hierzu zählt vor allem die Vermittlung von Dienstattributen, die zeitlichen Schwankungen unterlegen sind (z.B. die aktuelle Auslastung des Dienstanbieters). Werden solche Attribute von einem Directory-Agent über einen längeren Zeitraum zwischengespeichert, verlieren sie an Relevanz. Aus diesem Grund wurde im Rahmen des Projekts eine standardkompatible Erweiterung für SLP konzipiert, mit der zusätzlich zu den Attributwerten auch deren Alter vermittelt wird. Directory Agents und Service Agents passen diese Alterswerte laufend an, so dass der User Agent das tatsächliche Alter der Attribute erfährt. Anhand des Alters kann er dann entscheiden, ob eine direkte Anforderung aktueller Werte beim jeweiligen Service-Agent notwendig ist.

Die hohe Netzwerkdynamik führt außerdem dazu, dass die ständige Erreichbarkeit eines Dienstanbieters nicht garantiert werden kann. Dienste sollten daher nach Möglichkeit so ausgelegt sein, dass ein Wechsel zu einem alternativen Dienstanbieter ohne Unterbrechung der Dienstleistung durchgeführt werden kann. Hierfür ist es erforderlich, dass der Dienstnutzer ständig über die aktuelle Anbietersituation informiert bleibt und so potenzielle Ersatzanbieter identifizieren und einen eventuellen Anbieterwechsel vorbereiten kann. Außerdem wird es so möglich, freiwillige Anbieterwechsel zu initiieren, wenn dem Netzwerk ein besser geeigneter Dienstanbieter beitritt.

Generell haben die Übertragungseigenschaften des Netzwerks einen erheblichen Einfluss auf die Nutzungsqualität eines Dienstes. Daher sollten bei der Dienstanbieterauswahl nicht nur die Eigenschaften des Dienstanbieters, sondern auch Qualitätsmerkmale der zugehörigen Verbindung und Informationen über die Netzwerkstruktur berücksichtigt werden. Auf diese Informationsquellen wird in den Abschnitten 3. und 4. noch detaillierter eingegangen, in der ersten Projektphase wurde jedoch bereits die konzeptionellen Voraussetzungen geschaffen, diese Informationsquellen in die Dienstvermittlung einbeziehen zu können.

Aus diesem Grund wurde der SLP-User-Agent zu einem komplexen Dienstinformationssystem erweitert, dessen Gesamtstruktur in Abbildung 2 zu sehen ist [6, 8]. Die Grundidee dieses Konzeptes ist die Schaffung einer zentralen Instanz auf dem Benutzergerät, bei der alle Applikationen, aber auch Middlewaresysteme (z.B. adaptive Routing- oder Transportprotokolle) Informationen zu Dienstanbietern und Netzwerkeigenschaften über eine gemeinsame Schnittstelle anfragen können. Die Dienstinformationen werden dabei in einem Zwischenspeicher verwaltet und periodisch aktualisiert. Der Zwischenspeicher stellt also ein aktuelles Abbild der Anbietersituation im Netzwerk dar, wobei aus Effizienzgründen nur die Dienstinformationen verwaltet werden, die von den Benutzerapplikationen zuvor angefragt wurden. Über einen Zeitmechanismus werden Einträge, die längere Zeit nicht angefragt wurden, aus dem Zwischenspeicher entfernt und nicht weiter aktualisiert. Da alle

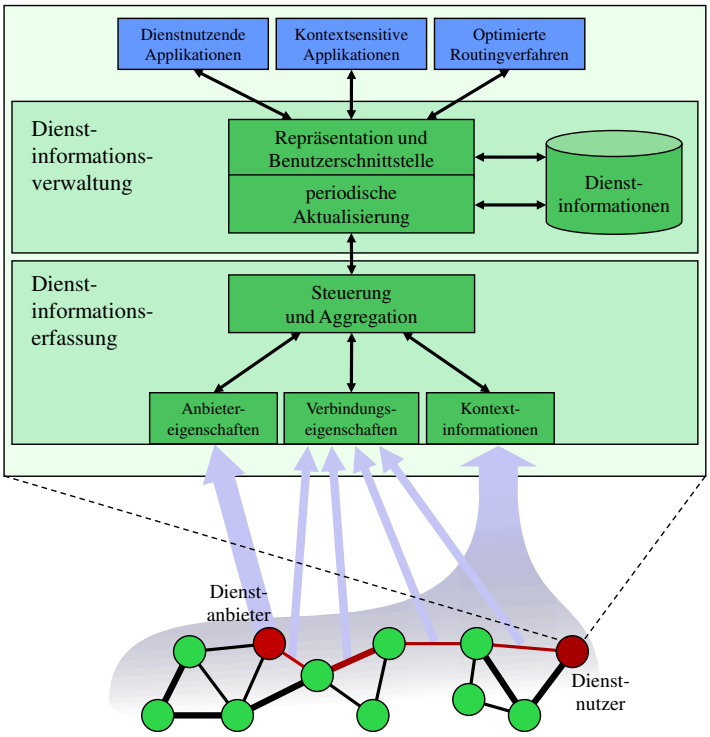


Abb. 2: Gesamtkonzept des Dienstinformationssystems

Applikationen auf den gleichen Informationsspeicher zugreifen, werden redundante Dienstanfragen eingespart und das Netzwerk entlastet.

Die Dienstinformationsverwaltung greift auf ein modulares Informationserfassungssystem zu, das für die eigentliche Ermittlung der verschiedenen Dienstinformationen verantwortlich ist. Für jede Informationsquelle ist dabei ein eigenes Modul verantwortlich. Für die Ermittlung der direkten Anbieterinformationen ist so zum Beispiel ein SLP-Modul zuständig. Eine zentrale Steuerungskomponente analysiert die Anfragen und ruft diejenigen Module auf, die zur Bearbeitung der Anfrage benötigt werden. Die eintreffenden Dienstinformationen werden in einer Antwortnachricht zusammengefasst und an die Informationsverwaltung zurückgeliefert. Die Aufteilung in verschiedene Informationsquellen bleibt dabei vor der Benutzerapplikation verborgen. Diese können also alle relevanten Informationen komfortabel über eine einzige Schnittstelle abfragen.

3. Messung von Dienstgütemerkmalen

Es wurde bereits mehrfach angedeutet, dass die Eigenschaften der Netzwerkverbindungen zwischen dem Dienstnutzer und den jeweiligen Dienstanbietern eine

entscheidende Rolle bei der Verwendung des Dienstes, aber auch schon bei der Anbieterauswahl spielen. Einige dieser Parameter (z.B. die Übertragungsverzögerung oder die Lokalität, d.h. der topologische Abstand zwischen den Endpunkten der Verbindung) lassen sich ohne signifikanten Aufwand leicht bestimmen. Andere, wie z.B. die Stabilität der Verbindung oder die verfügbare Bandbreite, erfordern hingegen komplexe, aber dennoch verkehrseffiziente Messverfahren. Im Rahmen der zweiten Projektphase wurden solche Verfahren entworfen und untersucht.

3.1. Prognose der Verbindungsstabilität

Einige Dienste erfordern eine mittel- oder langfristig bestehende Verbindung zwischen Dienstnutzer und Dienstanbieter. Dies kann jedoch in mobilen Ad-hoc-Netzwerken grundsätzlich nicht garantiert werden. Es ist allerdings möglich, die Stabilität einer Verbindung anhand von messbaren Informationen abzuschätzen. Ein solches Stabilitätsmaß sollte sich auf die Erreichbarkeit eines Dienstanbieters während eines dem Messzeitpunkt folgenden Zeitintervall beziehen, um die Dienstanbieterauswahl unterstützen zu können.

In der Literatur wurde bereits eine Vielzahl von Systemen zur Bestimmung von Stabilitätsparametern vorgeschlagen (u.a. [5, 7, 15]). Diese beziehen sich jedoch ausschließlich auf die Stabilität von einzelnen Links zwischen Nachbarknoten oder von konkreten Netzwerkpfaden. Moderne Routing-Verfahren für mobile Ad-hoc-Netzwerke sind jedoch in der Lage, den Verkehr einer Verbindung umzuleiten, wenn der aktuelle Pfad gestört ist, die Topologie aber einen alternativen Pfad zulässt. Aus Sicht des Dienstnutzers ist daher die Stabilität von Links oder konkreten Pfaden weniger von Bedeutung als die Verfügbarkeit eines *beliebigen* Pfades zwischen Dienstnutzer und -anbieter, unabhängig von zwischenzeitlichen Änderungen des konkreten Pfadverlaufs. Aus diesem Grund wurde ein abstraktes Stabilitätsmaß entwickelt, das die Verfügbarkeit eines Dienstanbieters im folgenden Zeitintervall quantifiziert [9, 14]. Dieses Maß wird durch eine Heuristik bestimmt, die auf zwei relevanten Beobachtungen beruht.

Die erste Beobachtung wird von detaillierten Simulationsstudien gestützt, bei denen die Abhängigkeit der Verbindungsstabilität (d.h. des zeitlichen Anteils des Messzeitraums, während dessen der Dienstanbieter erreicht werden konnte) von der aktuellen Länge des Pfades untersucht wurde. In Abbildung 3 ist nur ein kleiner Ausschnitt der Simulationsergebnisse zu sehen. Bei diesem Experiment wurden 30 Knoten auf der in der Legende angegeben Fläche verteilt und mit der ebenfalls dort angegebenen Geschwindigkeit gemäß dem Random-Waypoint-Mobilitätsmodell bewegt. Das Diagramm zeigt, dass die in diesen Szenarien gemessene Stabilität linear mit der Pfadlänge abnimmt. Die Länge des aktuellen Pfades ist also ein wichtiger Indikator für die Verbindungsstabilität.

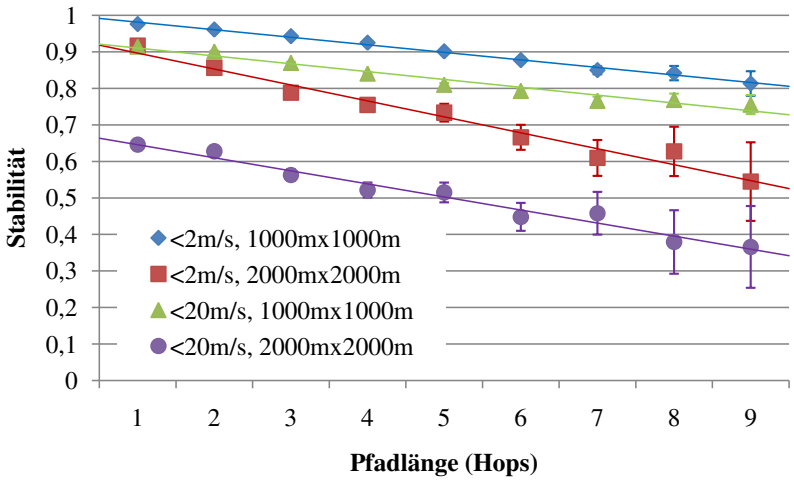


Abb. 3: Stabilität bei wachsender Pfadlänge

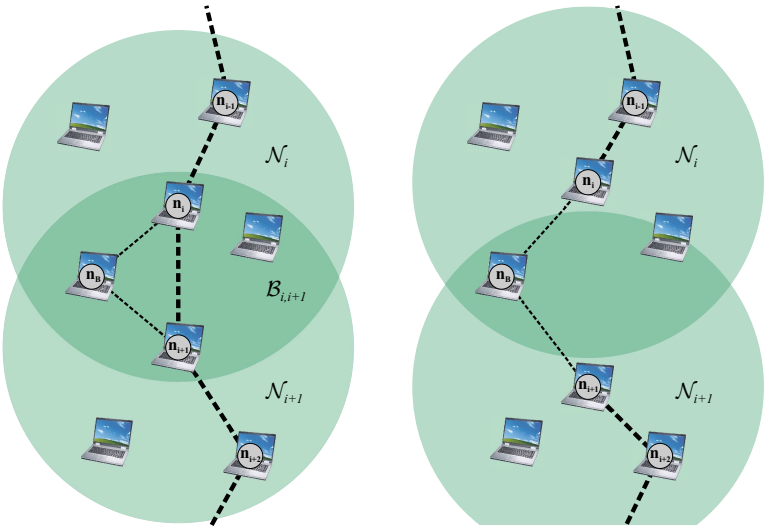
Neben der Pfadlänge verfügen die Knoten des Pfades auch über Nachbarschaftsinformationen. Sie kennen also die Knoten, die sich in ihrem direkten Übertragungsbereich befinden. Auch diese Informationen haben einen direkten Einfluss auf die Verbindungsstabilität, wie die in Abbildung 4 dargestellte Situation belegt. In Teil (a) ist ein Pfad mit zwei benachbarten Knoten n_i und n_{i+1} und ihren Nachbarschaften $\mathcal{N}_i$ und $\mathcal{N}_{i+1}$ abgebildet. Für die Stabilitätsprognose sind hierbei insbesondere die Knoten relevant, die sich in der Schnittmenge $\mathcal{B}_{i,i+1} = \mathcal{N}_i \cap \mathcal{N}_{i+1}$ befinden. Wenn sich die Knoten n_i und n_{i+1} wie in Abbildung 4(b) zu sehen von einander entfernen, können diese Knoten in der Regel die entstandenen Lücke überbrücken. Aus diesem Grund werden diese Knoten im Folgenden *Brückenknoten* genannt. Simulationsergebnisse haben gezeigt, dass die Stabilität einer Verbindung mit der Anzahl der Brückenknoten pro Link zunimmt.

Die *Brückenknotenheuristik* $\mathcal{S}(n_0, n_l)$ zur Prognose der Verbindungsstabilität berechnet sich daher aus der Gesamtzahl der Brückenknoten des aktuellen Pfades $n_0, \ldots, n_l$ und dessen Länge l:

$$\mathcal{S}(n_0, n_l) = \frac{1}{l^2} \sum_{i=0}^{l-1} |\mathcal{B}_{i,i+1}|$$

Die Pfadlänge wird quadratisch berücksichtigt, um zum einen die Anzahl der Brückenknoten auf die Anzahl der Links zu normieren und zum anderen um der bei steigender Pfadlänge abnehmenden Stabilität gerecht zu werden.

Zur Evaluation des Bewertungsverfahrens wurden umfangreiche Simulationsexperimente durchgeführt. Abbildung 5 zeigt die durchschnittliche Steigerung der Stabilität gegenüber der durchschnittlichen Verfügbarkeit, wenn das in der Legen-



(a) Vor Link-Abbruch (b) Nach Link-Abbruch

Abb. 4: Prinzip der Brückenknoten

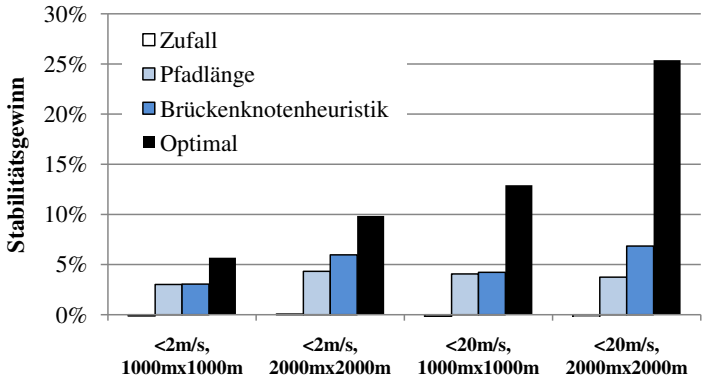


Abb. 5: Stabilitätsgewinn in verschiedenen Szenarien

de angegebene Kriterium zur Anbieterauswahl verwendet wurde. Die zufällige Auswahl eines Anbieters ergab demnach wie erwartet keinen Stabilitätsgewinn, während die Auswahl eines Anbieters anhand der Pfadlänge schon signifikante Gewinne erzeugt. Gerade in großen, also dünn besiedelten Szenarien kann die Brückenknotenheuristik diesen Gewinn noch deutlich steigern. Generell erscheinen die von diesen beiden Auswahlkriterien erzeugten Gewinne recht gering zu sein. Setzt man diese Ergebnisse in Relation zu den ebenfalls in der Abbildung dargestellten maximalen Gewinnen eines (hypothetischen) optimalen Verfahrens, sind die erzielen Ergebnisse jedoch trotz der geringen Informationsbasis beachtlich.

3.2. Messung verfügbarer Bandbreiten

Viele Dienste erfordern für eine erfolgreiche Ausführung eine bestimmte Mindestdatenrate. Die verfügbare Bandbreite, d.h. die maximale Datenrate, die ohne Störung fremder Datenflüsse über eine Verbindung zwischen Dienstanbieter und Dienstnutzer übertragen werden kann, ist daher ein entscheidendes Auswahlkriterium bei der Dienstsuche.

Zur Messung dieser verfügbaren Bandbreite wurden bereits zahlreiche Ansätze vorgeschlagen (z.B. [2, 3]). Die meisten dieser Verfahren wurden jedoch für drahtgebundene Netzwerke entworfen und sind wegen der völlig unterschiedlichen Übertragungseigenschaften nicht direkt auf drahtlose Netzwerke übertragbar.

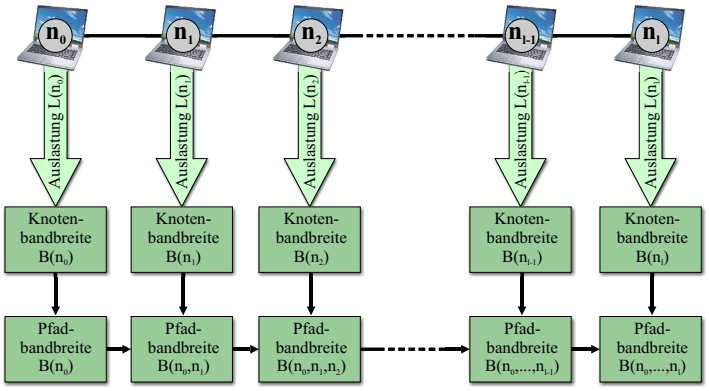


Abb. 6: Konzept der knotenbasierten Bandbreitenmessung

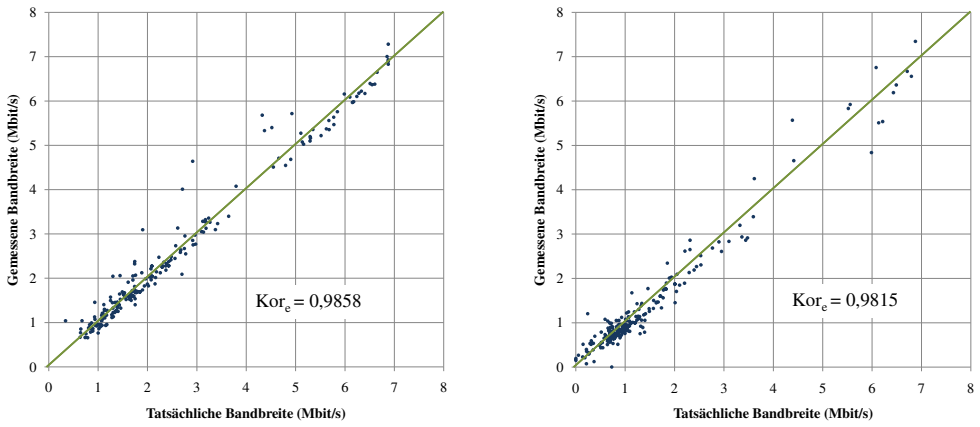
Die so genannten *knotenbasierten Messverfahren* haben sich jedoch auch für Ad-hoc-Netzwerke als geeignet erwiesen. Das Grundprinzip dieser Verfahren zeit Abbildung 6. Dabei wird vorausgesetzt, dass alle Knoten den Übertragungskanal permanent überwachen. So können sie die lokale Auslastung, also den Zeitanteil, in dem der Kanal an ihrer Position belegt ist, bestimmen. Unter Berücksichtigung ver-

schiedener technischer und physikalischer Faktoren kann der Knoten aus diesen Auslastungswerten eine *Knotenbandbreite* berechnen, die der Datenrate entspricht, die er einem zusätzlichen Datenfluss zur Verfügung stellen kann.

Zur Messung der verfügbaren Bandbreite des gesamten Pfades wird über diesen ein Messpaket übertragen, das unter anderem die verfügbare Bandbreite des bisher zurückgelegten Teilpfades enthält. Jeder Zwischenknoten berechnet aus diesem Wert und aus der eigenen Knotenbandbreite eine aktualisierte Teilpfadbandbreite und leitet diese an den nächsten Knoten weiter, der genauso verfährt. Der Zielknoten kann schließlich die endgültige *Pfadbandbreite* bestimmen.

Im Rahmen der zweiten Projektphase wurde ein solches Messverfahren analysiert und für die Verwendung in mobilen Ad-hoc-Netzwerken optimiert [12, 13]. Insbesondere wurde dabei auf die explizite Modellierung der *flussinternen Konkurrenz* eingegangen. Dieses Phänomen verhindert, dass zwei (oder mehr) benachbarte Knoten gleichzeitig senden können, da sich Funksignale gegenseitig stören würden. Die flussinterne Konkurrenz reduziert die verfügbare Bandbreite signifikant und muss bei der Berechnung der Pfadbandbreite explizit berücksichtigt werden.

Durch ausführliche Untersuchungen dieses Phänomens und der Entwicklung eines geeigneten Berechnungsverfahrens konnte die Genauigkeit der Bandbreitenmessung deutlich verbessert werden, wie die Streudiagramme in Abbildung 7 belegen. In diesem Diagrammen stellt jeder Punkt die gemessene Bandbreite der tatsächlichen Bandbreite in einem einzelnen Simulationslauf gegenüber. Die Punkte gruppieren sich offensichtlich nah an der eingezeichneten Idealllinie. Auch die angeführten Korrelationskoeffizienten von über 0,98 zeigen, dass dieses Messverfahren brauchbare Ergebnisse liefert.



(a) 30 Knoten auf $1000\mathrm{m} \times 1000\mathrm{m}$ (b) 30 Knoten auf $2000\mathrm{m} \times 2000\mathrm{m}$

Abb. 7: Genauigkeit der Bandbreitenmessung

Die Dynamik eines mobilen Ad-hoc-Netzwerks beeinflusst auch maßgeblich die verfügbaren Bandbreiten, so dass diese erheblichen Schwankungen unterlegen ist. Wenn für eine erfolgreiche Dienstleistung eine bestimmte Mindestbandbreite benötigt wird, ist die Wahrscheinlichkeit, dass diese Bandbreite im folgenden Zeitraum zur Verfügung steht, ein wichtiges Kriterium bei der Anbieterauswahl.

Im Rahmen der zweiten Projektphase wurde daher ein Prognoseverfahren entwickelt, das diese Verfügbarkeitswahrscheinlichkeit aus der statistischen Analyse von Auslastungswerten bestimmt. Dabei folgt das Verfahren grob den Schritten zur Messung der aktuellen Bandbreite. Hierfür wurde die Gesamtwahrscheinlichkeit zunächst in knotenbasierte Elementarwahrscheinlichkeiten zerlegt, d.h. jeder Knoten bestimmt zunächst, mit welcher Wahrscheinlichkeit er die erforderliche Bandbreite in der Folgeperiode bereitstellen kann. Simulationsexperimente haben gezeigt, dass diese Wahrscheinlichkeit gut durch eine auf das Intervall $[0, 1]$ normierte Laplacesche Doppelexponentialverteilung modelliert werden kann, deren Modalwert der aktuell verfügbaren Bandbreite entspricht und deren Varianz empirisch aus in der Vergangenheit gemessenen Auslastungswerten ermittelt werden kann.

Zusätzlich zu diesen Bandbreitenwahrscheinlichkeiten muss bei der Prognose auch die Wahrscheinlichkeit berücksichtigt werden, dass der betrachtete Pfad im Folgeintervall verbunden bleibt. Auch hierfür kann die Gesamtwahrscheinlichkeit in Elementarwahrscheinlichkeiten zerlegt werden, die sich auf einzelne Links beziehen und von den angrenzenden Knoten eigenständig bestimmt werden können.

Wie bei der Messung der aktuellen Bandbreite werden die von den Pfadknoten bestimmten Elementarwahrscheinlichkeiten bei der Übertragung eines Messpakets gesammelt und zu einer Gesamtwahrscheinlichkeit verrechnet. Simulationsexperimente haben gezeigt, dass die Wahrscheinlichkeit nicht in allen Szenarien exakt prognostiziert werden konnte, aber die prognostizierte Werte waren stets mit der im Folgeintervall gemessenen relativen Häufigkeit stark korreliert. Dies qualifiziert die gemessenen Werte als geeignetes Kriterium für die bandbreitenbezogene Anbieterauswahl.

3.3. Integration in das Dienstinformationssystem

Die in den vorangegangen Abschnitten vorgestellten Verfahren wurden so konzipiert, dass sie einen möglichst geringen Verkehrsaufwand erzeugen. Sowohl für die Messung der Brückenknotenheuristik als auch für die Bestimmung der Bandbreitenparameter wurden iterative Erfassungsverfahren entworfen, die die von Knoten zu Knoten übertragenen Informationen auf wenige Bytes reduzieren. Durch die Integration dieser Verfahren in die Pfadsuchemechanismen des *Dynamic Source Routing Protocols (DSR)* konnte die Effizienz noch weiter gesteigert werden.

Um aus dem Dienstinformationssystem auf diese Erfassungsverfahren zugreifen zu können, werden Module verwendet, die als Informationsquelle ins Dienstinformationssystem eingebunden werden und über eine entsprechende Schnittstelle die jeweiligen Erfassungsverfahren im Routing-Protokoll aktivieren. Diese durchgehende Integration gewährleistet, dass die Messung der Qualitätsparameter völlig gekapselt ist und die Benutzerapplikation nicht direkt auf sie zugreifen muss.

4. Identifizierung stabiler Teilstrukturen

In den letzten Jahren finden drahtlose *Mesh-Netzwerke* immer weitere Verbreitung. Diese Unterform der Ad-hoc-Netzwerke zeichnet sich dadurch aus, dass einzelne Knoten fest montiert sind (z.B. auf Hausdächern) und über Richtfunkstrecken verbunden sein können, die sehr stabil und leistungsfähig sind. Die Netzwerkstruktur ist also sehr heterogen - auf der einen Seite gibt es mobile Knoten mit problematischen Übertragungseigenschaften und auf der anderen Seite stationäre Knoten mit einer recht hohen Übertragungsqualität.

Bei der Auswahl von Dienstanbietern, aber auch bei Routing-Entscheidungen sollte diese Heterogenität berücksichtigt werden, indem Verbindungen über stabile Teilstrukturen bevorzugt werden. Diese Teilstrukturen sind jedoch in der Regel nicht allgemein bekannt, so dass sie zur Laufzeit ermittelt werden müssen. Hierfür wurde in der dritten Projektphase ein Verfahren entworfen, das stabile Knoten durch die statistische Analyse von Paketübertragungen identifiziert. Dieses Verfahren beruht lediglich auf der Beobachtung des regulären Netzwerkverkehrs und kommt somit ohne explizite Messübertragungen aus.

Das Dynamic Source Routing Protocol überträgt Pakete anhand einer im Paketkopf angegebenen Folge von Knotenadressen. Jeder Zwischenknoten sucht seine eigene Adresse in dieser Folge und leitet das Paket an den nächstgenannten Knoten weiter. Zum Aufbau dieser Knotenfolge wird ein Pfadanfragepaket über einen Broadcast-Mechanismus an alle erreichbaren Knoten verteilt und dabei die bislang zurückgelegte Knotenfolge im Paketkopf aufgezeichnet. Der Zielknoten sendet über den so ermittelten Pfad ein Antwortpaket mit der gesuchten Knotenfolge zurück.

Erkennt ein Knoten während einer Datenübertragung, dass er ein Paket nicht an den im Paketkopf angegebenen Nachbarknoten weiterleiten kann (z.B. weil dieser Nachbarknoten den Übertragungsbereich verlassen hat), sendet er eine Fehlernachricht an den Startknoten zurück, der daraufhin eine erneute Pfadsuche veranlassen kann. Das DSR-Protokoll sieht noch eine Reihe weiterer Kontrollnachrichten vor, um eine möglichst robuste und zügige Datenübertragung zu gewährleisten. All diese Nachrichten enthalten wertvolle Informationen über den Status einzelner Knoten. Wenn ein Knoten ein Datenpaket empfängt oder auch nur passiv mithört, kann

er daraus schließen, dass die bisher durchlaufenen Knoten erreichbar sind. Ein Fehlerpaket deutet hingegen darauf hin, dass ein Knoten nicht mehr erreicht werden kann.

Die grundlegende Idee ist nun, dass der interessierte Knoten die von ihm mitgehörten Topologieinformationen statistisch auswertet, um so auf der einen Seite einen Überblick über die aktuelle Netzwerkstruktur zu gewinnen und zum anderen jeden ihm bekannten Knoten bezüglich seiner Stabilität zu bewerten. Dazu verwaltet der interessierte Knoten für jeden anderen Knoten n_i eine Datenstruktur, in der er folgende Informationen ablegt:

- eine Liste der benachbarten Knoten von n_i,
- den Zeitpunkt t_i^f, zu dem n_i zum ersten mal genannt wurde,
- den Zeitpunkt t_i^l, zu dem n_i zum letzten mal genannt wurde,
- die Häufigkeit c_i der Nennungen in diesem Zeitraum.

Immer, wenn der interessierte Knoten eine DSR-Nachricht empfängt, aktualisiert er den Eintrag für die im Nachrichtenkopf genannten Knoten und Links, d.h. der Zeitpunkt t_i^l wird für alle im Paketkopf genannten Knoten auf den aktuellen Zeitpunkt gesetzt, die Häufigkeit um eins erhöht und die Liste der benachbarten Knoten gemäß der enthaltenen Pfadangaben erweitert. Im Falle eines Fehlerpakets wird der als fehlerhaft erkannte Link aus den Nachbarschaftsinformationen gelöscht.

Die Stabilität eines Knotens ist umso größer, je öfter er in einem möglichst langen Zeitraum genannt wird. Daher werden die Knoten zunächst anhand des Existenzzeitraums $t_i^l - t_i^f$ und der Aktualisierungsrate $u_i = {}^{c_i}/_{(t_i^l - t_i^f)}$ bewertet. Um stabile von instabilen Knoten zu unterscheiden, werden anschließend Knoten mit besonders langer Lebensdauer und hoher Aktualisierungsrate statistisch identifiziert.

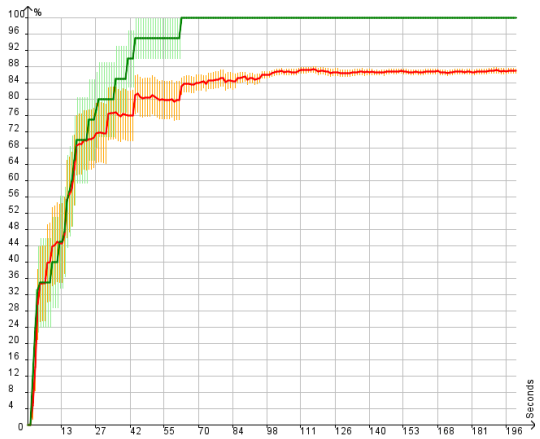


Abb. 8: Entwicklung der Erkennungsrate

Mit zunehmender Anzahl empfangener Pakete unterscheiden sich die Werte für stationäre und für mobile Knoten immer deutlicher, so dass die Klassifikation weniger fehleranfällig ist. Abbildung 8 zeigt die Entwicklung der Erkennungsrate in einem Simulationsexperiment mit fünf stationären und 45 mobilen Knoten auf einer $1000\,\mathrm{m} \times 1000\,\mathrm{m}$ großen Fläche. Die obere Linie markiert dabei den durchschnittlichen Prozentsatz der richtig erkannten fixen Knoten. Offensichtlich steigert sich diese Rate sehr schnell auf 100%, es werden also alle fixen Knoten richtig erkannt. Problematisch ist allerdings die untere Kurve, die den Prozentsatz der fälschlicherweise als fix detektierten Knoten darstellt. Dieser Prozentsatz ist sehr hoch, lässt sich aber durch die relativ geringe Knotengeschwindigkeit von maximal $2\,\mathrm{m/s}$ erklären. Bei derart geringen Geschwindigkeiten ist die Erreichbarkeit aller Knoten recht hoch, so dass sich mobile Knoten kaum von stationären Knoten unterscheiden.

Insgesamt kann das Verfahren also stabile Strukturen im Netzwerk recht zuverlässig bestimmen und gibt somit dem Dienstinformationssystem die Möglichkeit, die Stabilitätsbewertungen von einzelnen Knoten bei der Auswahl von Dienstanbietern berücksichtigen zu können.

5. Begleitende Tätigkeiten

Während der gesamten Projektlaufzeit wurde darauf geachtet, dass die einzelnen Komponenten zu einem Gesamtsystem zusammengesetzt werden können. Da der Aufbau und die Konfiguration eines realen Ad-hoc-Netzwerks realistischer Größe sehr kostenintensiv und aufwändig ist, wurde für Funktions- und Leistungstests eine Emulationsumgebung geschaffen, in der alle Komponenten integriert werden konnten. Die Basis hierfür bildet der Netzwerksimulator *ns-2*, in dessen DSR-Implementierung die Messung der Qualitätsparameter und die Erfassung stabiler Strukturen integriert wurde. Das Dienstinformationssystem selbst wurde hingegen außerhalb der Simulationsumgebung für reale Geräte implementiert.

Um trotz der verschiedenen Plattformen das Gesamtkonzept untersuchen und demonstrieren zu können, wurde das von ns-2 bereits unterstützte Konzept zur Netzwerkemulation weiterentwickelt. Bei diesem Konzept wird das Ad-hoc-Netzwerk mit allen Knotenbewegungen und Paketübertragungen in Echtzeit simuliert. Einige der simulierten Knoten sind realen Geräten zugeordnet, auf denen die Dienstapplikationen sowie die Dienstvermittlung laufen und die mit dem Simulationsrechner über eine schnelles drahtgebundenes Netzwerk verbunden sind. Diese Geräte sind so konfiguriert, dass sie alle ausgehenden Pakete zum Simulationsrechner weiterleiten. Dieser simuliert die Netzwerkübertragung in Echtzeit und leitet das Paket bei erfolgreicher Zustellung in der Simulation an das entsprechende reale Gerät weiter. Für die realen Geräte hat das Netzwerk demnach die Eigenschaften des simulierten Ad-hoc-Netzwerks.

Neben vielen kleineren technischen Problemstellungen stellte die Emulation von Multicast-Übertragungen die größte Herausforderung dar. Bei diesen Übertragungen können die Pakete nach der Simulation nicht einfach an die Multicast-Adresse weitergeleitet werden, weil die Netzwerkknoten in der Simulation die Pakete zu verschiedenen Zeitpunkten empfangen, im realen Netzwerk ein Multicast-Paket jedoch gleichzeitig an alle Empfänger zugestellt wird. Daher musste ein neues Adressierungkonzept entwickelt werden, bei dem der Simulator die Multicast-Pakete per Unicast an die endgültigen Empfänger weiterleitet [10].

Zu Demonstrationszwecken [11] wurde für das Gesamtsystem außerdem ein Internet-Gateway als Beispieldienst entwickelt. Ein solcher Dienst kann von Knoten angeboten werden, die neben der Funkschnittstelle ins Ad-hoc-Netzwerk über eine weitere Netzwerkverbindung ins Internet verfügen (z.B. über Kabel, Satellit, UMTS). Mobile Knoten können sich über einen virtuellen Netzwerktunnel mit diesem Knoten verbinden und so auf das Internet zugreifen.

Schon zu Beginn der zweiten Projektphase wurde das Dienstinformationssystem in einem projektübergreifenden Konzept eingesetzt [1]. Bei diesem Konzept bildete das Dienstinformationssystem das Bindeglied zwischen den semantischen Dienstsuchemechanismen des DIANE-Projekts (Universität Jena) und dem Multicast-Routing des MAMAS-Projekts (Universität Karlsruhe). Das Dienstinformationssystem war dabei dafür zuständig, die semantischen Dienstbeschreibungen zu vermitteln. Auf der anderen Seite profitierte die Dienstvermittlung von der Effizienz des MAMAS-Multicastings.

6. Zusammenfassung

Im Rahmen des Projekts *„Entwicklung eines Ad-hoc Service Managements für selbstorganisierende vernetzte mobile Systeme“* wurde ein umfassendes Erfassungs- und Verwaltungssystem für verschiedenartige Dienstinformationen konzipiert, implementiert und analysiert. Mithilfe dieses Dienstinformationssystems können Benutzer nicht nur Zugriffsinformationen und Eigenschaften geeigneter Dienstanbieter ermitteln, sondern auch detaillierte Informationen über die Verbindungsqualität und der Netzwerkstruktur erfassen, um sie bei der Dienstanbieterauswahl zu berücksichtigen. Bei der Konzeption aller Teilsysteme wurde darauf geachtet, dass sie zum einen robust genug für eine Verwendung in mobilen Ad-hoc-Netzwerken sind und dass sie zum anderen die knappen Netzwerkressourcen möglichst wenig beanspruchen. Sowohl die detaillierten Simulationsexperimente als auch die Funktionstests in der Emulationsumgebung haben erwiesen, dass diese Verfahren effizient und praktikabel einsetzbar sind. Somit bildet das Dienstinformationssystem eine universelle Grundlage für zukünftige Erweiterungen und Optimierungen.

Literaturverzeichnis

[1] BAUMUNG, P., S. PENZ und M. KLEIN: *P2P-Based Semantic Service Management in Mobile Ad-hoc Networks*. In: *7th ACM International Conference on Mobile Data Management (MDM 2006)*, Nara, Japan, Mai 2006.

[2] CHEN, L. und W.B. HEINZELMAN: *QoS-Aware Routing Based on Bandwidth Estimation for Mobile Ad Hoc Networks*. IEEE Journal on Selected Areas in Communications, 23(3):561–572, März 2005.

[3] DOVROLIS, CONSTANTINOS, PARAMESWARAN RAMANATHAN und DAVID MOORE: *Packet-Dispersion Techniques and a Capacity-Estimation Methodology*. IEEE/ACM Transactions on Networking, 12(6):963–977, Dezember 2004.

[4] GUTTMAN, E., C. PERKINS, J. VEIZADES und M. DAY: *Service Location Protocol, Version 2* . RFC 2608 (Proposed Standard), Juni 1999.

[5] JIANG, S., D. HE und J. RAO: *A Prediction-based Link Availability Estimation for Routing Metrics in MANETs*. IEEE Transactions on Networking, 13(6):1302–1312, 2005.

[6] KRITZNER, J., S. PENZ und O. SPANIOL: *Adaptive Services in Heterogeneous Networks*. Information Technologies and Control Journal, 3(3):2–12, Mai 2005.

[7] LEE, M.-G. und S. LEE: *A Link Stability Model and Stable Routing for Mobile Ad-Hoc Networks*. In: *2006 IFIP International Conference on Embedded And Ubiquitous Computing*, Seiten 904–913, Seoual, Südkorea, August 2006.

[8] PENZ, S.: *SLP-based Service Management for Dynamic Ad-hoc Networks*. In: *3rd Workshop on Middleware for Pervasive and Ad Hoc Computing (MPAC '05)*, Grenoble, Frankreich, November 2005.

[9] PENZ, S.: *A DSR Extension for Connection Stability Assessment in Mobile Ad-hoc Networks*. In: *3rd IEEE Workshop on Pervasive Wireless Networking (PWN'07)*, Seiten 509–513, White Plains, New York, USA, März 2007.

[10] PENZ, S.: *Wireless Multicast Support for the NS-2 Emulation Environment*. In: *15th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS'07)*, Seiten 267–273, Istanbul, Türkei, Oktober 2007.

[11] PENZ, S.: *SLP-based Service Management with QoS Server Selection in Mobile Ad-hoc Networks*. In: *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'08)*, Newport Beach, Kalifornien, USA, Juni 2008.

[12] PENZ, S. und A. ACHTERFELD: *Efficient Assessment of Available Bandwidth During DSR Route Discovery*. In: *5th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS'08)*, Seiten 49–56, Garmisch-Partenkirchen, Deutschland, Januar 2008.

[13] PENZ, S. und A. ACHTERFELD: *Probabilistic Prediction of Available Bandwidth in Multi-Hop Ad-Hoc Networks*. In: *2008 IEEE International Conference on Communications (ICC'08)*, Peking, China, Mai 2008.

[14] PENZ, S. und M. WENIG: *Bridging Nodes Density: A Connection Stability Heuristic for Dynamic Ad-hoc Networks*. In: *IFIP International Conference on Embedded and Ubiquitous Computing 2006 (EUC'06)*, Seiten 672–681, Seoul, Südkorea, August 2006.

[15] ZHANG, H. und Y.-N. DONG: *A Novel Path Stability Computation Model for Wireless Ad Hoc Networks*. IEEE Signal Processing Letters, 14(12):928–931, Dezember 2007.

Kooperative schichtenübergreifende Überlastkontrolle für mobile Ad-Hoc-Netzwerke

Björn Scheuermann, Christian Lochert und Martin Mauve

Institut für Informatik, Heinrich-Heine-Universität Düsseldorf

Zusammenfassung Die oft beobachteten Schwierigkeiten beim Einsatz gängiger und – in anderen Umgebungen – lange bewährter Netzwerkprotokolle in drahtlosen Multihop-Netzwerken sind eine direkte Folge der besonderen Eigenschaften dieser speziellen Umgebung. Bisherige Arbeiten haben diese Eigenschaften in der Regel als Nachteil bzw. ausschließlich als Quelle von Problemen angesehen. Hier werden die Besonderheiten der Umgebung hingegen aus einem anderen Blickwinkel betrachtet und konstruktiv als Grundlage für alternative Ansätze genutzt. Dabei zeigt sich, dass gerade diese Eigenschaften häufig als Fundament für unkonventionelle und gleichzeitig sehr effektive Lösungen dienen können.

1. Einführung

Kooperative drahtlose Multihop-Netzwerke erlauben ihren Nutzern die Kommunikation ohne feste Infrastruktur auch deutlich über die Funkreichweite eines einzelnen Gerätes hinaus. Die Einsatzmöglichkeiten solcher Netzwerke unterscheiden sich dabei ebenso wie die Fähigkeiten der eingesetzten Geräte: Die Spanne reicht von ressourcenarmen Sensorknoten, die ausschließlich und speziell für das Erfassen, Weiterleiten und Versenden von Messungen gebaut sind, bis hin zu durchaus leistungsstarken Systemen mit umfangreichen Speicher- und Prozessorkapazitäten.

Diskutiert werden als Einsatzgebiete von leistungsstarken Geräten beispielsweise die Vernetzung von Fahrzeugen auf der Straße zur Verbesserung von Verkehrssicherheit und Fahrkomfort, Mesh-Netzwerke, die die Reichweite von Internetzugangspunkten vergrößern, oder die Kommunikation in Katastrophenfällen, wenn keine Infrastruktur verfügbar ist. Bei den eher leistungsschwächeren Geräten stehen Überwachungs-, Kontroll- und Alarmierungsanwendungen im Mittelpunkt der Überlegungen. Gleichwohl nutzen alle diese Systeme Kommunikationsformen, die deutliche Parallelen aufweisen: Sie basieren allesamt auf drahtloser Kommunikation über mehrere Hops; im Weiteren werden wir das breite Spektrum von Geräten und Anwendungen daher unter dem Begriff *drahtlose Multihop-Netzwerke* zusammenfassen.

Drahtlose Multihop-Netzwerke unterscheiden sich in einigen Punkten fundamental von anderen Netzwerkarchitekturen wie dem Internet und weisen viele einzigartige und interessante Eigenschaften auf. Die vielleicht bemerkenswerteste ist, dass beim Einsatz omnidirektionaler Antennen Übertragungen nicht nur vom angesprochenen Empfänger, sondern auch von anderen Netzwerkknoten in der Umgebung empfangen werden können. Darüber hinaus serialisiert das Medium Übertragungen lokal: Innerhalb einer beschränkten Umgebung können nicht mehrere Übertragungen gleichzeitig erfolgreich stattfinden. Wegen der sich daraus ergebenden Besonderheiten stellen drahtlose Multihop-Netzwerke existierende Kommunikationsprotokolle vor große Schwierigkeiten. Die Eigenschaften dieser Netze wurden daher bislang nahezu ausschließlich als Nachteil gesehen und auch als solcher behandelt.

Im Rahmen des Projekts *Kooperative schichtenübergreifende Überlastkontrolle für mobile Ad-Hoc-Netzwerke* werden die besonderen Eigenschaften eines drahtlosen Multihop-Netzwerkes aus einer anderen Perspektive betrachtet. Wie sich dabei herausstellt, können diese oft als Basis für neue, manchmal ungewöhnliche Lösungswege dienen. Statt existierende Ansätze so zu verändern, dass ihre schlechte Eignung für den Einsatz in drahtlosen Multihop-Umgebungen weniger deutlich in Erscheinung tritt, werden hier die Eigenschaften der Netzwerke für alternative Herangehensweisen gezielt ausgenutzt. Für eine Reihe von fundamentalen Problemen in drahtlosen Multihop-Netzwerken werden solche neuartigen Wege aufgezeigt. Dabei ist der Schlüsselgedanke stets derselbe: Die Eigenschaften des Übertragungsmediums lassen sich nutzen, um *implizit* Information zu gewinnen oder Vorgänge zu koordinieren, ohne dabei explizit Daten austauschen zu müssen.

Der Hauptfokus liegt hier auf der Funktionalität, die im Internet-Protokollstapel der Sicherungs- bzw. Medienzugriffsschicht und der Transportschicht zugeordnet ist. Alle vorgestellten Ansätze haben gemein, dass sie die Medieneigenschaften bereits in ihrem Ansatz berücksichtigen, und so in einem schichtenübergreifenden Ansatz auf einer gemeinsamen Grundlage zu Lösungen für verschiedenste Probleme führen. Zunächst werden die zentralen Funktionen des Transmission Control Protocol (TCP) aus neuen Blickwinkeln betrachtet. TCP realisiert im Internet-Protokollstapel Überlastkontrolle und zuverlässigen Ende-zu-Ende-Datentransport. Der im nachfolgenden Abschnitt vorgestellte Ansatz zur Überlastkontrolle namens *CXCC* resultiert aus diesen Überlegungen. CXCC wird durch den Zuverlässigkeitsmechanismus *BarRel* in Abschnitt 3. zu einer vollständigen TCP-Alternative ergänzt. Das in Abschnitt 4. diskutierte *BMCC* stellt eine Verallgemeinerung von CXCC auf den Fall der Multicast-Kommunikation dar. Anschließend wird in Abschnitt 5. aufgezeigt, wie durch die Ausnutzung der Medieneigenschaften und basierend auf einer Reinterpretation der zuvor eingeführten Ansätze die verfügbare Übertragungskapazität in bestimmten Situationen massiv gesteigert werden

kann. Dabei werden mehrere Paketübertragungen auf dem Medium im Inneren des Netzwerkes mittels Network Coding kombiniert. Nach der Diskussion dieser Protokolle für unterschiedliche Aufgaben in drahtlosen Multihop-Netzwerken werden in Abschnitt 6. Techniken zur Evaluation und Ergebnisinterpretation sowohl für simulative Untersuchungen als auch für Realwelt-Experimente vorgestellt, die im Rahmen der Entwicklung der genannten Protokolle entstanden sind und die für ihre Implementation und Untersuchung zum Einsatz kamen. Schließlich stellt Abschnitt 7. laufende Arbeiten in Richtung eines formalen Beweises der Korrektheit der Protokolle und Protokollimplementationen vor.

2. Implizite Überlastkontrolle: CXCC

Die erste Fragestellung, die hier betrachtet werden soll, ist die Überlastkontrolle. Ein Überlastkontrollmechanismus regelt die Datenrate, mit der ein Sender Daten in ein Netzwerk übertragen darf, um zu vermeiden dass die Rate der eingespeisten Daten die Kapazität des Netzwerkes übersteigt. Im Internet ist der Grund für den Verlust eines Paketes in der Regel ein Pufferüberlauf vor einer überlasteten Verbindung. Der Überlastkontrollmechanismus von TCP macht sich dies zunutze: TCP schließt aus dem Auftreten von Verlusten auf das Vorhandensein von Überlast und reduziert seine Senderate. Dies hat sich in drahtlosen Multihop-Netzwerken jedoch als problematisch erwiesen [5, 10]. Überlasteffekte äußern sich in drahtlosen Multihop-Netzwerken grundlegend anders. Wegen der lokal beschränkten Bandbreite sind in solchen Netzwerken nicht einzelne Verbindungen, sondern ganze (geografische) Bereiche des Netzwerks überlastet. Außerdem treten häufig spontane Paketverluste auf, beispielsweise wegen Kollisionen oder aufgrund variierender Eigenschaften des Übertragungsmediums. Schon die Grundannahme der TCP-Überlastkontrolle, dass Paketverluste durch Pufferüberläufe entstehen und somit eine direkte Folge von Überlasterscheinungen sind, ist in drahtlosen Multihop-Netzwerken also falsch.

Deshalb ist es wichtig, alternative Möglichkeiten zur Überlastkontrolle zu betrachten. Die hier eingeführte *implizite schrittweise Überlastkontrolle* ist ein solcher Ansatz. Sie basiert auf dem Aufbau von Rückstau im Netzwerk mittels sehr kurzer Paketwarteschlangen. Die Grundidee ist dabei äußerst einfach: Die Übertragung eines Paketes an den Nachfolgeknoten entlang der Route zum Ziel wird verhindert, solange dieser Nachfolgeknoten nicht das vorangegangene Paket weitergeleitet hat. So wird ein Zufluss von Daten, der die verfügbare Transportkapazität übersteigt, vermieden. Umgesetzt und evaluiert wird dieses Konzept im Protokoll *Cooperative Cross-layer Congestion Control* (CXCC), das in [15] vorgestellt wurde.

Die zentralen Faktoren, die einen solchen Ansatz in drahtlosen Multihop-Netzwerken nicht nur ermöglichen, sondern ihn sogar zu einer besonders guten Wahl

machen, sind die speziellen Eigenschaften des Mediums. Um dies zu verstehen, ist es hilfreich, das konkrete Vorgehen des Protokolls beim Weiterleiten von Daten einer Verbindung zu betrachten. Schematisch ist es in Abbildung 1 dargestellt, die den Ablauf einer Datenübertragung von einer Quelle links über zwei Zwischenstationen zu einem Zielknoten rechts zeigt.

Nach dem Weiterleiten eines Paketes muss ein Knoten warten (er ist „blockiert"), bis er erfährt, dass sein Nachfolger das Paket abermals weitergeleitet hat. Er kann dies erfahren, indem er das Weiterleiten seines Nachfolgers mithört. Dieses Mithören wird durch das Übertragungsmedium ermöglicht. Es liefert einerseits ein *implizites Acknowledgment*, also eine indirekte Bestätigung, dass die vorherige Übertragung zum Nachfolger erfolgreich war, ohne dass ein separates Bestätigungspaket nötig wäre. Gleichzeitig löst es die Blockierung und erlaubt die Übertragung eines Folgepaketes. Nur der endgültige Zielknoten leitet das Paket nicht weiter und muss deshalb seinem unmittelbaren Vorgänger den Empfang explizit bestätigen.

Noch zentraler tritt die Berücksichtigung der Medieneigenschaften in einem anderen Zusammenhang hervor: Während der Übertragung eines Folgepaketes in einen Knoten hinein kann das Medium nicht gleichzeitig dafür genutzt werden, ein früher eingetroffenes Paket aus diesem Knoten heraus weiterzuleiten. Im Gegensatz zu kabelgebundenen Netzen ist es in drahtlosen Multihop-Umgebungen deshalb *nicht* sinnvoll, mehr als ein Paket in jeder Station entlang der Route zwischenzuspeichern. CXCC berücksichtigt diese besondere Eigenschaft des Mediums: Der beschriebene Mechanismus gibt einem Knoten nach dem Erhalt eines Paketes stets Gelegenheit, dieses auch weiterzuleiten, bevor sein Vorgänger ihm ein Folgepaket zusenden darf. Dies hat auch zur Folge, dass jeder Zwischenknoten maximal ein Paket puffert.

Durch den einfachen Mechanismus des Blockierens baut sich ein Rückstau in Richtung Quelle auf, wenn die Übertragung an einer Stelle im Netz verzögert ist. Letztlich kann damit auch der Quellknoten Pakete nicht schneller ins Netzwerk einspeisen, als diese es passieren können – das Resultat ist also ein Überlastkontrollmechanismus. Das Besondere dieser Herangehensweise ist, dass die Überlastkontrolle rein implizit erfolgt: Es gibt keinen dedizierten Mechanismus oder Algorithmus zur Festlegung der Senderate, kein Knoten muss explizit Übertragungsraten schätzen und keine Komponente des Netzwerks muss Rückmeldung über den aktuellen Lastzustand geben. Die einfache Regel des Abwartens nach der Übertragung eines Paketes bis zum Mithören des Weiterleitens bringt ein selbstregulierendes System hervor.

Selbstverständlich ist von diesem Ansatz bis zu einem vollständigen Protokoll noch ein weiter Weg zu gehen. Insbesondere muss die Möglichkeit von (bei drahtlosen Übertragungen durchaus häufigen) Übertragungsfehlern oder nicht mehr er-

reichbaren Nachfolgeknoten (beispielsweise in mobilen Netzen aufgrund der Bewegung der Knoten) in Betracht gezogen werden. Natürlich darf ein Knoten nach einer fehlgeschlagenen Übertragung nicht unbegrenzt darauf warten, dass sein Nachfolger ein Paket weiterleitet, das dieser nie erhalten hat.

Um mit beiden Problemen effektiv und effizient umzugehen kommt bei CXCC ein Mechanismus namens *Request for Acknowledgment* (RFA) zum Einsatz. Mit einem RFA-Packet kann ein Knoten nach Versand eines Paketes seinen Nachfolger entlang der Route fragen, ob dieser eine vorangegangene Übertragung erhalten hatte. Beispielhaft ist das in Abbildung 2 zu sehen, wo der Fall ein verlorengegangenes implizites Acknowledgment durch einen RFA-ACK-Handshake aufgelöst wird. Trotz – oder gerade wegen – der Vielfalt von Möglichkeiten, wie Verluste und Verzögerungen beim Weiterleiten in einer so komplexen und unberechenbaren Umgebung wie einem drahtlosen Multihop-Netzwerk zusammenspielen können, zeigt ein RFA-basierter Ansatz im Vergleich zur gängigen Lösung unmittelbarer Neuübertragungen beim zu langen Ausbleiben von (impliziten) Bestätigungen eine deutlich bessere Performance im Zusammenspiel mit impliziter schrittweiser Überlastkontrolle. Abbildung 3 demonstriert die Wirksamkeit der CXCC-Überlastkontrolle in einem sehr einfachen ns-2-Simulationsszenario mit einer 10-Hop-Kettentopologie und bidirektionalem UDP-Datenverkehr. Wie zu erwarten bricht der erreichte Durchsatz bei Verwendung von IEEE 802.11 ohne Überlastkontrollmechanismus rapide ein, sobald ein kritischer Wert der durch die Anwendungen in den beiden Endknoten angebotenen Last überschritten wird. Mit Überlastkontrolle mittels CXCC wird hingegen – bei Verwendung von RFA – die Quelldatenrate durch Rückstau zu den Quellknoten so geregelt, dass der maximal mögliche Durchsatz unabhängig von der angebotenen Last realisiert werden kann. In weitergehenden Vergleichsuntersuchungen zeigt sich eine klare Überlegenheit gegenüber TCP-basierten Ansätzen zur Überlastkontrolle in drahtlosen Multihop-Umgebungen sowohl in Hinblick auf den erreichten Durchsatz als auch bezüglich anderer wichtiger Eigenschaften wie Übertragungsverzögerungen und Protokoll-Overhead. Eine tiefergehende Diskussion der Ergebnisse und ihrer Zusammenhänge sowie Details zu CXCC finden sich in [15].

3. Implizite Zuverlässigkeit: BarRel

Einer der Gründe für die überragenden Ergebnisse von CXCC im Vergleich zu anderen Ansätzen ist die Tatsache, dass für die Überlastkontrolle keine Kontrollpakete notwendig sind, die wie die Bestätigungspakete von TCP vom Ziel zurück zur Quelle laufen. Solche Pakete wirken sich in drahtlosen Multihop-Netzwerken sehr nachteilig aus [4], unter anderem weil sie mit den Datenpaketen um das Medium konkurrieren und häufig die Ursache von Kollisionen und damit Paket-

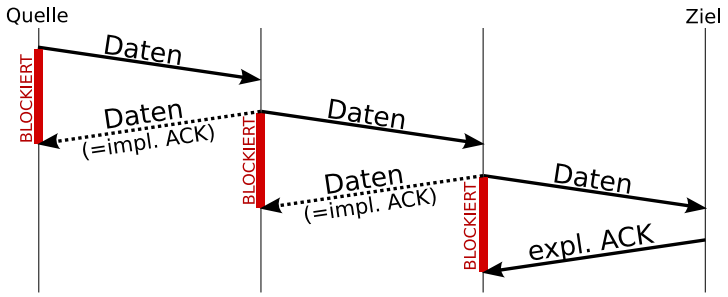


Abbildung 1. Paketweiterleitung in CXCC.

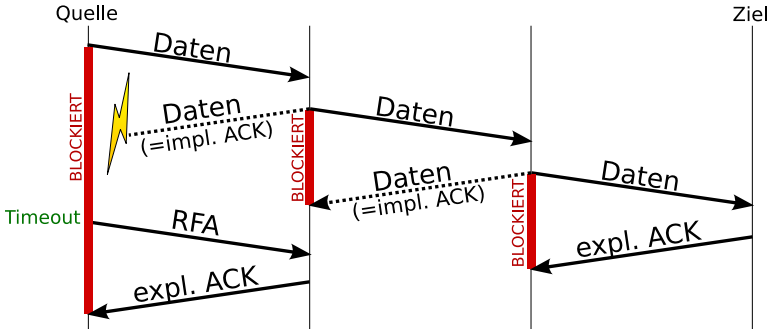


Abbildung 2. Erholung vom Verlust eines impliziten Acknowledgment mittels RFA.

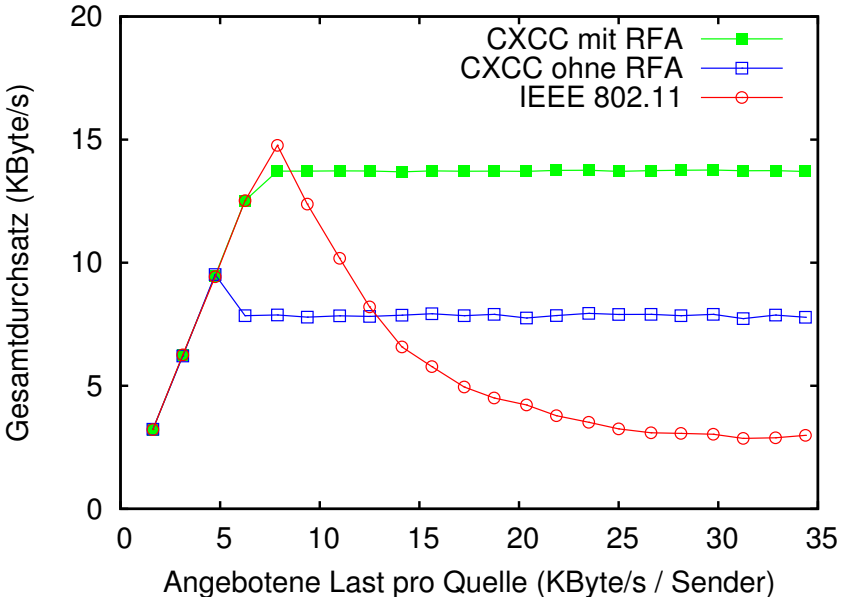


Abbildung 3. Durchsatz mit CXCC in bidirektionaler Kettentopologie.

verlusten sind. Allerdings vermeidet CXCC zwar Ende-zu-Ende-Kontrolldatenverkehr für die Überlastkontrolle, gewährleistet dabei aber keine TCP-äquivalente Ende-zu-Ende-Zuverlässigkeit. Hierfür könnten wiederum explizite Ende-zu-Ende-Bestätigungspakete notwendig sein. In der Tat wurde bislang allgemein angenommen, dass sie für eine zuverlässige Ende-zu-Ende-Datenübertragung unvermeidlich sind.

Mit dem Protokoll *Backpressure Reliability* (BarRel), dessen Details in [11] diskutiert werden, konnte konstruktiv gezeigt werden, dass dies nicht zutrifft. BarRel verfolgt den mit CXCC eingeschlagenen Weg weiter und ergänzt die implizite Überlastkontrolle durch einen Mechanismus für die zuverlässige Ende-zu-Ende-Übertragung von Daten. Es nutzt Wissen über die Funktionsweise der CXCC-Überlastkontrolle, um implizit auf die erfolgreiche Zustellung von Datenpaketen an einen weiter entfernten Zielknoten zu schließen. Im Gegensatz zu existierenden TCP-äquivalenten Transportprotokollen benötigt BarRel keinen kontinuierlichen Strom von Bestätigungspaketen und vermeidet dadurch den problematischen gegenläufigen Kontrolldatenverkehr.

Die Idee, die BarRel zugrunde liegt, nutzt CXCCs Limitierung der Zahl von zwischengespeicherten Paketen in den Knoten entlang der Route. Wie zuvor gesehen, beschränkt CXCC die Länge der Paketwarteschlange in jedem Zwischenknoten auf ein Paket. Wir nehmen nun an, dass die Routenlänge n beim Quellknoten bekannt ist; dies ist mit vielen Routingansätzen problemlos realisierbar. Verschickt der Quellknoten dann das $i+n$-te Paket, so kann er daraus schließen, dass das i-te Paket beim Zielknoten eingetroffen sein muss – andernfalls hätte CXCC den Versand dieses Paketes nicht erlaubt!

Auch hier stellen sich wieder eine ganze Reihe von praktischen Herausforderungen, wenn die Idee in einem lauffähigen Protokoll umgesetzt werden soll. So stellt sich etwa die Frage, wie fehlgeschlagene Paketübertragungen und Änderungen der Route zum Ziel behandelt werden sollen. Es zeigt sich jedoch, dass diese Probleme lösbar sind und dass es möglich ist, TCP vollständig und transparent durch CXCC und BarRel zu ersetzen. Die wahrscheinlich interessanteste Frage im Zusammenhang mit bestätigungsfreier Ende-zu-Ende-Zuverlässigkeit ist die nach dem Verhalten beim (vorübergehenden oder endgültigen) Ende einer Datenübertragung. Denn wenn der Versand des $i+n$-ten Paketes dazu dient, indirekt den Erhalt des i-ten Paketes zu bestätigen, dann muss selbstverständlich auch der Fall berücksichtigt werden, dass es kein $i+n$-tes Paket gibt.

Der erste und offensichtliche Ansatz ist der Versand eines einzelnen Bestätigungspaketes vom Ziel zurück zur Quelle für das letzte Datenpaket. Dies löst das Problem, widerspricht aber dem eigentlich rein impliziten Ansatz. Eine überraschend einfache, pragmatische Lösung entschärft das Problem des fehlenden $i+n$-ten Paketes auf andere Weise: Wenn die Anwendung keine weiteren Daten

mehr liefert, die durch ihren Versand für die implizite Bestätigung früher versandter Pakete dienen können, so kann das Protokoll schlicht den Sendepuffer mit n *leeren* Paketen – in BarRel *Capacity-Refill-Pakete* (CaRe-Pakete) genannt – auffüllen. Wurde das letzte CaRe-Paket verschickt, so muss zuvor das letzte „wichtige" Datenpaket den Zielknoten erreicht haben. Damit wird zuverlässige Datenübertragung ohne jeglichen Kontrollpaketfluss in Gegenrichtung möglich.

Der Versand von n zusätzlichen CaRe-Paketen am Übertragungsende wirkt auf den ersten Blick ineffizient, weist aber bei genauerer Betrachtung durchaus Vorteile auf. Insbesondere ist es bei der Verwendung von CaRe-Paketen nicht notwendig, eine Zeit festzulegen, die maximal auf ein Bestätigungspaket gewartet werden soll, bevor ein Paketverlust angenommen wird. Die Festlegung dieser Wartezeit ist in zuverlässigen Datenübertragungsprotokollen stets ein großes Problem, und wurde in der Vergangenheit intensiv erforscht. Dennoch bleibt sie ein potentieller wunder Punkt eines jeden Protokollentwurfs, der Bestätigungspakete einsetzt – einschließlich TCP und der BarRel-Variante mit einem einzelnen Bestätigungspaket am Übertragungsende. Bei Verwendung von CaRe-Paketen ist die Wahl einer solchen Wartezeit nicht nötig, wodurch das Problem vollständig vermieden wird.

4. Rückdruckbasierte Multicast-Überlastkontrolle: BMCC

Die implizite schrittweise Überlastkontrolle kann auch auf Multicast-Datenübertragungen angewendet werden, also auf Kommunikationsszenarien, in denen ein Sender identische Daten zeitgleich an mehrere Empfänger versenden möchte. Dies wird im Protokoll *Backpressure Multicast Congestion Control* (BMCC) umgesetzt, dessen Details in [16] beschrieben werden. BMCC erzielt eine effektive Regelung der Quelldatenrate bei geringen Paketlaufzeiten und minimalem Kontrolldatenaufkommen.

Für die Zustellung an eine Gruppe von Empfängern statt eines einzelnen Zielknotens werden die Daten entlang einer Baumstruktur weitergeleitet, deren Wurzel der Quellknoten ist und deren Blätter von Zielknoten gebildet werden. Ein weiterleitender Knoten muss die Daten also an mehr als einen Nachfolger weiterreichen, wenn sich bei ihm der Multicast-Baum verzweigt. Wird das Rückdruckprinzip der impliziten schrittweisen Überlastkontrolle auf diese Situation erweitert, so müssen zunächst die Mechanismen für implizite und explizite Bestätigungen, zur Erkennung und Behebung von Übertragungsfehlern und zum Identifizieren von nicht mehr erreichbaren Nachfolgeknoten entsprechend verallgemeinert werden. Im konkreten Fall wurde dies in Zusammenarbeit mit der Universität Mannheim für das dort entwickelte geographische Multicast-Routingprotokoll Scalable Position-Based Multicast (SPBM) [17] implementiert und untersucht. Da bei geographischen Ansätzen zum Multicast-Routing dem Quellknoten die einzelnen Gruppenmitglie-

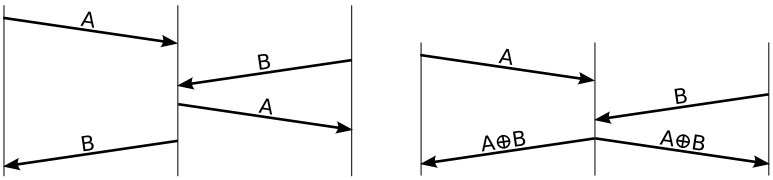
der ebenso wenig bekannt sind wie deren genaue Anzahl, ist das Durchführen von Überlastkontrolle hier besonders anspruchsvoll.

Eine naheliegende Erweiterung des CXCC-Rückdruckmechanismus erlaubt einem Knoten die Übertragung eines Folgepaketes, sobald *alle* seine Nachfolger das vorangegangene Paket weitergeleitet haben. Wenn dies jeder Knoten im Multicast-Baum so handhabt, dann wird sich Rückdruck entsprechend der „engsten" Stelle im gesamten Baum aufbauen. Die Senderate der Quelle wird sich also nach dem langsamsten aller Empfänger richten. Dies kann durchaus erwünscht sein, wenn es wichtig ist, allen Empfängern alle Daten (oder zumindest einen Großteil) zu übermitteln. Ist jedoch auch der Empfang eines mehr oder weniger großen Bruchteils der Daten für einen Empfänger nützlich – etwa, weil entsprechende Codierungsverfahren zum Einsatz kommen –, dann bleibt gegebenenfalls viel Medienkapazität auf dem Weg zu besser erreichbaren Empfängern ungenutzt. Eine alternative Version von BMCC erlaubt deshalb, basierend auf einer Relaxation des Rückstauprinzips, die Zustellung mit höheren Datenraten an besser erreichbare Empfänger, ohne dass hierfür die langsamen Empfängern zur Verfügung stehende Medienkapazität beeinträchtigt wird.

5. Koordiniertes Network Coding: noCoCo

Network Coding kombiniert – in der Regel mittels Linearkombinationen – im Inneren eines Netzwerkes mehrere Pakete in eine einzige Übertragung. Ursprünglich wurde dies in [2] vorgeschlagen; viele theoretische Arbeiten zeigen, dass dadurch große Kapazitätsgewinne möglich sind. Ein einfaches Beispiel in drahtlosen Netzwerken zeigt Abbildung 4. In Abbildung 4(a) werden zwei Pakete übertragen, Paket A vom linken zum rechten und Paket B vom rechten zum linken Knoten. Beide werden vom mittleren Netzwerkknoten weitergeleitet. Insgesamt sind somit vier Übertragungen notwendig. In Abbildung 4(b) werden die gleichen Pakete ausgetauscht. Statt jedoch A und B einzeln weiterzuleiten, überträgt der Knoten in der Mitte das bitweise Exklusiv-Oder $A \oplus B$. Da jeder der Endknoten das von ihm selbst erzeugte Paket kennt, können beide die Operation umkehren und die für sie bestimmten Pakete zurückgewinnen. Dadurch wird eine der vier Übertragungen eingespart. Dies lässt sich auf komplexere Situationen verallgemeinern.

Von Katti et al. wurde Network Coding im Protokoll COPE [8] erstmals praktisch in drahtlosen Multihop-Netzwerken angewendet. Dort geschieht dies rein *opportunistisch*: Kombinierbare Übertragungen werden zusammengefasst, falls sich Gelegenheiten hierfür spontan ergeben. Dies nutzt die Möglichkeiten jedoch nicht vollständig aus. Gerade in kritischen Situationen stellen sich Gelegenheiten zum Network Coding nur vergleichsweise selten spontan ein. Der in Kooperation mit der Universität Cambridge entwickelte Ansatz *Near-Optimal Co-ordinated Coding*



(a) Ohne Network Coding. (b) Mit Network Coding.

Abbildung 4. Paketaustausch mit und ohne Network Coding.

(noCoCo) [14] ersetzt das rein opportunistische Vorgehen von COPE in Situationen, in denen gegenläufiger Datenverkehr dieselben Zwischenstationen passiert. Es stellt sich heraus, dass dies ideal mit der impliziten schrittweisen Überlastkontrolle in CXCC kombiniert werden kann. CXCC wird deshalb als Basis benutzt und die zugrunde liegende Idee der impliziten Koordination wird weiterentwickelt, um das Zusammenspiel von Übertragungen benachbarter Netzwerkknoten zu steuern. So wird es mit noCoCo möglich, Gelegenheiten zum Network Coding nicht nur spontan zu nutzen, sondern ihre Existenz für bidirektionalen Datenverkehr zu *garantieren*. Dadurch kann der höchstmögliche Gewinn tatsächlich praktisch realisiert werden.

Analytisch lassen sich Zusammenhänge zwischen der Zahl im Netzwerk gepufferter Pakete und dem mit Network Coding in bidirektionalem Datenverkehr realisierbaren Gewinn herleiten; es zeigt sich, dass noCoCo dem theoretischen Optimum extrem nahe kommt. Vergleichende Untersuchungen mit COPE zeigen, dass sich der Datendurchsatz mit koordiniertem Network Coding oft dramatisch steigern lässt, und dass gleichzeitig die Paketlaufzeit und der durch Kontrolldaten und Übertragungswiederholungen verursachte Overhead stark vermindert werden. Abbildung 5 zeigt im gleichen einfachen Simulationsszenario wie zuvor in der Diskussion von CXCC, dass sich durch opportunistisches Network Coding mit COPE der Durchsatz (gleichgültig ob mit oder ohne Einsatz von CXCC) zwar geringfügig steigern lässt, die Vorteile jedoch dramatisch höher sind, wenn Network Coding mit noCoCo koordiniert wird. Abbildung 6 analysiert dasselbe Szenario aus einem anderen Blickwinkel und betrachtet die Zahl an Bytes, die durchschnittlich über das drahtlose Medium übertragen werden, um ein Byte Nutzlast einen Hop näher an sein Ziel zu bringen – einschließlich aller Kontrollnachrichten, Header, Übertragungswiederholungen etc.. Besonders erstaunlich ist hierbei, dass aufgrund der hohen Gewinne durch das koordinierte Network Coding mit noCoCo im Mittel *weniger* als ein Byte auf dem Medium übertragen werden muss, um ein Byte Nutzlast weiterzutransportieren. Eine tiefergehende Diskussion der Ergebnisse findet sich in [14].

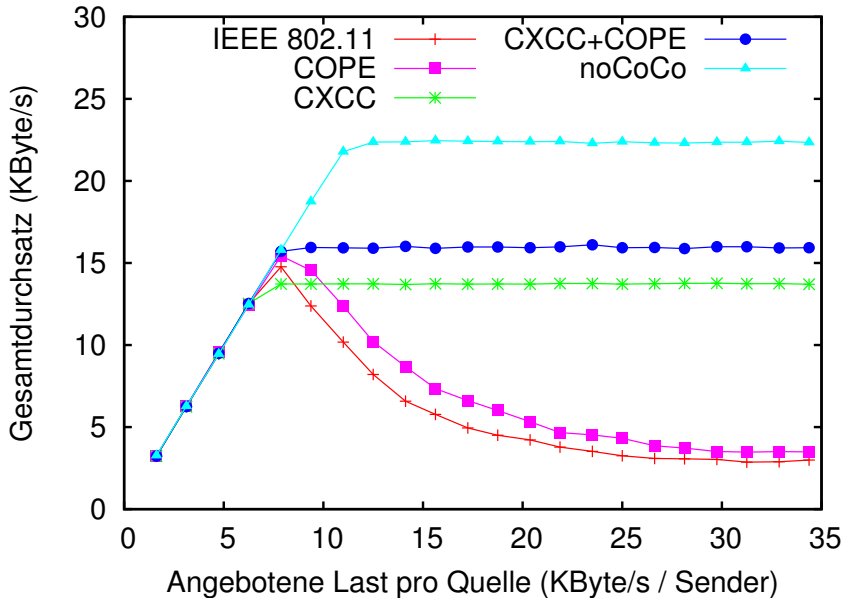


Abbildung 5. Durchsatz mit noCoCo in bidirektionaler Kettentopologie.

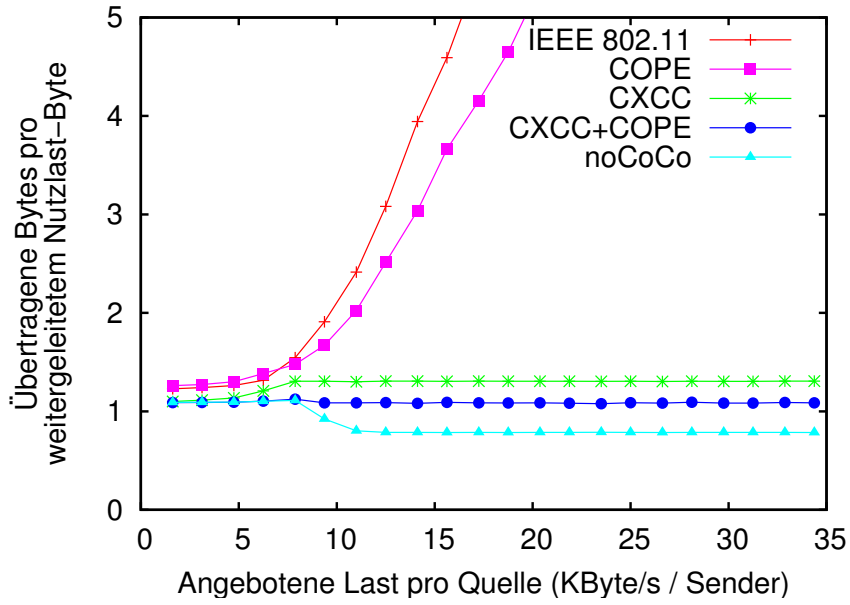


Abbildung 6. Protokolloverhead mit noCoCo in bidirektionaler Kettentopologie.

6. Evaluation schichtübergreifender Protokolle

Die Untersuchung der vorgestellten Protokolle erfolgte einerseits durch ausführliche Simulationsstudien, darüber hinaus jedoch auch durch Realwelt-Messungen auf Basis mehrerer unabhängiger Implementationen. Bereits im Falle der Untersuchungen mit Netzwerksimulatoren und noch deutlicher bei den realen Implementationen ergeben sich dabei praktische Schwierigkeiten aus der schichtübergreifenden Architektur der Protokolle. Die an mehreren Stellen notwendig werdenden Eingriffe sind eine Herausforderung bei der Implementation in einem Netzwerksimulator ebenso wie beim Einbinden in die Protokollarchitektur realer Systeme. Bei der Auswertung von Simulationen und Messungen wird erneut deutlich, dass die existierenden Werkzeuge und Vorgehensweisen die Untersuchung von schichtübergreifenden Ansätzen nur eingeschränkt unterstützen.

Im Falle der Simulationen ist das Verfolgen von Vorgängen über mehrere Schichten und Knoten hinweg essentiell wichtig für das Verständnis von Protokollverhalten und -fehlverhalten. Genau dieses Verfolgen von Ereignisabfolgen ist jedoch mit den üblicherweise bereitgestellten Analysemöglichkeiten unnötig schwierig. In Kooperation mit der Universität Mannheim wurde deshalb ein Werkzeug entwickelt, dass die interaktive Analyse von Simulationsläufen des Netzwerksimulators ns-2 auch in komplexen Netzwerkumgebungen unterstützt. Das entwickelte Werkzeug namens Huginn [12, 13] bietet eine Vielzahl von verschiedenen Darstellungsmöglichkeiten über in die Visualisierung des Netzwerkgeschehens eingebettete dynamisch berechnete Auswertungen. Es erlaubt dadurch dem Entwickler eines Protokolls, das Geschehen im Netzwerk interaktiv zu erkunden und bis hinab zu einzelnen Übertragungsvorgängen zu analysieren.

Um Untersuchungen von Realwelt-Implementationen schichtübergreifender Protokolle bis hinab zur Medienzugriffsschicht zu ermöglichen ist zunächst eine Testplattform notwendig, in der sich die entsprechenden Bestandteile des Protokollstapels überhaupt modifizieren lassen. Auf Basis gängiger WLAN-Hardware sind Modifikationen, wie sie beispielsweise für das Protokoll CXCC notwendig sind, nicht ohne weiteres möglich. Für eine Implementation von CXCC und entsprechende Messungen kamen daher die ESB-Sensorknoten der Freien Universität Berlin zum Einsatz. Um die Durchführung von Experimenten auf diesen Geräten und die Auswertung der Resultate zu ermöglichen, wurden ein umfassender Satz von Protokollen, Mechanismen und Werkzeugen entwickelt, die wir in [6, 7] näher vorstellen. Beispielsweise unterstützt das dort beschriebene ESB-basierte Cross-Layer-Testbett die Analyse und gezielte Beeinflussung von Netzwerktopologie und Routing-Tabellen sowie die speicherplatz- und rechenzeiteffiziente Aufzeichnung von Logdaten während der Durchführung von Messläufen.

7. Formale Verifikation der Protokolle

Die hier bislang diskutierten Protokolle für drahtlose Multihop-Netzwerke haben ihre Funktion und Performance in Simulationen und teilweise auch in realen Implementationen unter Beweis gestellt. Die Korrektheit der Protokollspezifikation einerseits und der konkreten Implementationen andererseits lassen sich jedoch mit solchen Untersuchungen alleine niemals zweifelsfrei belegen. Insbesondere in Anwendungsfällen, die entweder inhärent sicherheitskritisch sind (etwa im Bereich Fahrzeug-zu-Fahrzeug-Kommunikation oder beim Einsatz von drahtlosen Multihop-Netzwerken in Katatrophenszenarien) oder in denen Modifikationen am einmal ausgebrachten System schwierig und aufwändig sind (etwa in Sensornetzen mit weit verstreuten, schwer zugänglichen Netzwerkknoten) wäre jedoch eine solche zweifelsfreie Untermauerung der Korrektheit höchst wünschenswert.

Es existieren eine ganze Reihe von Methoden zur formalen Verifikation von Softwaresystemen, die prinzipiell auch für Netzwerkprotokolle zum Einsatz kommen können. Es stellt sich jedoch bei genauerer Betrachtung heraus, dass gerade drahtlose Multihop-Netzwerke aufgrund ihres hohen Grades an Parallelität und der vielen nichtdeterministischen Elemente in ihrem Verhalten solche Ansätze vor große Herausforderungen stellen. Die formale Modellierung von Protokolle in drahtlosen Multihop-Umgebungen wurde noch nicht in signifikanter Tiefe wissenschaftlich untersucht, entsprechende Ansätze befinden sich noch in einem sehr frühen Stadium.

Um Erfahrungen mit der Anwendung formaler Methoden für drahtlose Multihop-Kommunikationsprotokolle zu sammeln und dabei gleichzeitig weitere Erkenntnisse über das Verhalten der entworfenen Protokolle zu gewinnen wurden deshalb in Kooperation mit der Arbeitsgruppe für Softwaretechnik der Universität Düsseldorf erste Schritte hin zu einem formalen Modell der Protokolle CXCC und BarRel unternommen. Da die Grundstruktur dieser Protokolle trotz ihrer großen Effizienz auf wenigen, einfachen Prinzipien und Regeln beruht, erscheinen sie als besonders gut geeignet, um trotz vollständiger Abbildung der Funktionalität die Größe und Komplexität der Modelle beherrschbar zu halten.

Einerseits wird dabei auf einen Beweis dafür hingearbeitet, dass die Kombination aus CXCC und BarRel in ihrer Funktionalität äquivalent zu TCP ist, andererseits zeichnet sich daran anschließend die Möglichkeit der Herleitung einer beweisbar korrekten Protokollimplementation ab. Für die entsprechenden Arbeiten kommen Event-B [1] als formale Modellierungsmethode und das Werkzeug ProB [9] zum Einsatz. Die entsprechenden Arbeiten dauern noch nach, erste Zwischenergebnisse wurden in [3] vorgestellt.

8. Zusammenfassung

Die vorgestellten Protokolle und Lösungsansätze für verschiedenartige Probleme in drahtlosen Multihop-Netzwerken haben in ihrem Kern alle gemeinsam, dass sie die Eigenschaften des zugrundeliegenden Systems aus einem neuen Blickwinkel betrachten. Statt des Versuchs, die von anderen Netzen stark abweichenden Faktoren durch Modifikationen zu kompensieren oder zu kaschieren, werden die Besonderheiten der spezifischen Umgebung als etwas Positives aufgefasst, das selbst zu effektiven Lösungen beitragen kann.

Vorgeschlagen wurden Protokolle und Mechanismen zur Überlastkontrolle mittels impliziten Rückdrucks in CXCC, BarRel für die zuverlässige, TCP-äquivalente Ende-zu-Ende-Datenübertragung ohne kontinuierlichen Strom von Ende-zu-Ende-Acknowledgments, eine Verallgemeinerung der CXCC-Überlastkontrolle für Multicast-Datenverkehr namens BMCC und die Koordination lokaler Network-Coding-Operationen auf Basis der CXCC-Grundprinzipien in noCoCo. Jeder der vorgeschlagenen Ansätze nutzt Wissen über die Eigenschaften des Netzwerks, um Informationen implizit zu gewinnen, und alle zeigen, dass dadurch neue und manchmal unkonventionelle, doch höchst effektive Lösungen möglich werden. Darüber hinaus wurden Werkzeuge und Vorgehensweisen entwickelt, die die Untersuchung von schichtübergreifenden Protokollen sowohl simulativ als auch mit realen Experimenten signifikant erleichtern oder gar erst ermöglichen. Schließlich zeigen erste Ergebnisse aus der Modellierung und Untersuchung der vorgeschlagenen Protokolle mittels formaler Methoden, dass durch die gezielte Weiterentwicklung und den konsequenten Einsatz entsprechender Techniken beweisbar korrektes Verhalten auch in komplexen Umgebungen wie drahtlosen Multihop-Netzwerken durchaus erreichbar sein kann.

Literatur

[1] ABRIAL, JEAN-RAYMOND, DOMINIQUE CANSELL und DOMINIQUE MÉRY: *Refinement and Reachability in Event$_B$*. In: *ZB '05: Proceedings of the 4th International Conference of Z and B Users*, Seiten 222–241, April 2005.

[2] AHLSWEDE, RUDOLF, NING CAI, SHUO-YEN LI und RAYMOND YEUNG: *Network Information Flow*. IEEE Transactions on Information Theory, 46(4):1204–1216, 2000.

[3] BENDISPOSTO, JENS, MICHAEL JASTRAM, MICHAEL LEUSCHEL, CHRISTIAN LOCHERT, BJÖRN SCHEUERMANN und INGO WEIGELT: *Validating Wireless Congestion Control and Realiability Protocols using ProB and Rodin*. In: *FMWS '08: Proceedings of the International Workshop on Formal Methods for Wireless Systems*, August 2008.

[4] DE OLIVEIRA, RUY und TORSTEN BRAUN: *A Smart TCP Acknowledgment Approach for Multihop Wireless Networks*. IEEE Transactions on Mobile Computing, 6(2):192–205, Februar 2007.

[5] FU, ZHENGHUA, XIAOQIAO MENG und SONGWU LU: *How Bad TCP Can Perform In Mobile Ad Hoc Networks*. In: *ISCC '02: Proceedings of the 7th IEEE International Symposium on Computers and Communication*, Seiten 298–303, Juli 2002.

[6] JERSCHOW, YVES IGOR, BJÖRN SCHEUERMANN, CHRISTIAN LOCHERT und MARTIN MAUVE: *A Cross-Layer Protocol Evaluation Framework on ESB Nodes (Demo)*. In: *REALMAN '06: Proceedings of the 2nd International Workshop on Multi-hop Ad Hoc Networks: from Theory to Reality*, Seiten 104–106, Mai 2006.

[7] JERSCHOW, YVES IGOR, BJÖRN SCHEUERMANN, CHRISTIAN LOCHERT und MARTIN MAUVE: *A Real-World Framework to Evaluate Cross-Layer Protocols for Wireless Multihop Networks*. In: *REALMAN '06: Proceedings of the 2nd International Workshop on Multi-hop Ad Hoc Networks: from Theory to Reality*, Seiten 1–6, Mai 2006.

[8] KATTI, SACHIN, HARIHARAN RAHUL, WENJUN HU, DINA KATABI, MURIEL MEDARD und JON CROWCROFT: *XORs in The Air: Practical Wireless Network Coding*. In: *SIGCOMM '06: Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, Seiten 243–254, September 2006.

[9] LEUSCHEL, MICHAEL und MICHAEL BUTLER: *ProB: A Model Checker for B*. In: *FM '03: Proceedings of the 12th International FME Symposium*, Seiten 855–874, September 2003.

[10] RAGHUNATHAN, VIVEK und PANGANAMALA R. KUMAR: *A Counterexample in Congestion Control of Wireless Networks*. Elsevier Performance Evaluation, 64:399–418, Juni 2006.

[11] SCHEUERMANN, BJÖRN: *Reading Between the Packets – Implicit Feedback in Wireless Multihop Networks*. Doktorarbeit, Heinrich-Heine-Universität, Düsseldorf, Oktober 2007.

[12] SCHEUERMANN, BJÖRN, HOLGER FÜSSLER, MATTHIAS TRANSIER, MARCEL BUSSE, MARTIN MAUVE und WOLFGANG EFFELSBERG: *Huginn: A 3D Visualizer for Wireless ns-2 Traces*. In: *MSWiM '05: Proceedings of the 8th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Seiten 143–150, Oktober 2005.

[13] SCHEUERMANN, BJÖRN, HOLGER FÜSSLER, MATTHIAS TRANSIER, MARTIN MAUVE und WOLFGANG EFFELSBERG: *Visualizing Wireless ns-2 Traces in 3D*. In: *MobiCom '05: The 11th Annual ACM International Conference on Mobile Computing and Networking, Demo Session*, September 2005.

[14] SCHEUERMANN, BJÖRN, WENJUN HU und JON CROWCROFT: *Near-Optimal Coordinated Coding in Wireless Multihop Networks*. In: *CoNEXT '07: Proceedings of the 3rd International Conference on Emerging Networking Experiments and Technologies*, Dezember 2007.

[15] SCHEUERMANN, BJÖRN, CHRISTIAN LOCHERT und MARTIN MAUVE: *Implicit Hop-by-Hop Congestion Control in Wireless Multihop Networks*. Elsevier Ad Hoc Networks, 6(2):260–286, April 2008.

[16] SCHEUERMANN, BJÖRN, MATTHIAS TRANSIER, CHRISTIAN LOCHERT, MARTIN MAUVE und WOLFGANG EFFELSBERG: *Backpressure Multicast Congestion Control in Mobile Ad-Hoc Networks*. In: *CoNEXT '07: Proceedings of the 3rd International Conference on Emerging Networking Experiments and Technologies*, Dezember 2007.

[17] TRANSIER, MATTHIAS, HOLGER FÜSSLER, JÖRG WIDMER, MARTIN MAUVE und WOLFGANG EFFELSBERG: *A Hierarchical Approach to Position-Based Multicast for Mobile Ad-hoc Networks*. Springer Wireless Networks, 13(4):447–460, August 2007.

MAMAS – Mobility aware Multicast for ad-hoc groups in self organizing networks

Peter Baumung und Martina Zitterbart

Institut für Telematik, Universität Karlsruhe (TH)

Zusammenfassung – Das Projekt MAMAS beschäftigte sich mit der Konzeption eines P2P-basierten Gruppendienstes für mobile Ad-hoc-Netze. Um die Anforderungen unterschiedlicher Anwendungen bestmöglich unterstützen zu können, wurde eine modulare Architektur konzipiert, anhand welcher der Gruppendienst durch Kombination optimierter Protokollbausteine flexibel komponiert werden kann. Es wurden unterschiedliche Protokolle in die Architektur integriert, darunter eine hocheffiziente Overlay-Topologie sowie generische Konzepte zur Verbesserung der Skalierbarkeit oder zur Reaktion auf Gruppenmitgliedsausfälle. Durch Kapselung der System-Abhängigkeiten ist der Gruppendienst Plattform-unabhängig und konnte somit sowohl in Simulationsumgebungen evaluiert als auch in reale Anwendungen integriert werden.

1. Einführung und Projektziele

Ziel des Projekts war die Konzipierung und Umsetzung von *adaptiven und mobilitätsbewussten Gruppendiensten* in mobilen Ad-hoc-Netzen (MANETs). Letztere entstehen durch direkte Kommunikation WLAN-fähiger Endgeräte, welche zusätzlich zu ihrer Rolle als Endsysteme eine Weiterleitungsfunktion übernehmen. Da MANETs keine fest installierte Infrastruktur erfordern, zeigen sie eine hohe Flexibilität. Gruppendienste sind in MANETs von besonderer Relevanz, da viele Anwendungsszenarien eine Kommunikation von Gruppen erfordern: So lassen sich anhand von MANETs beispielsweise die Einsätze von Rettungs- und militärischen Truppen koordinieren oder Veranstaltungen in verteilten Lehrbetrieben, wie dem in Abbildung 1 gezeigten Campusszenario, unterstützen.

1.1. Anforderungen

An den in diesem Projekt zu konzipierenden Gruppendienst werden im Wesentlichen zwei getrennt betrachtbare Anforderungen gestellt:

1. Der Gruppendienst soll unter Berücksichtigung der *Eigenschaften von MANETs* eine *hohe Effizienz* aufweisen. In direktem Zusammenhang zu dieser Anforderung stehen die Forderungen nach einer möglichst sparsamen Ressourcen-Nutzung, der Berücksichtigung der potenziellen Mobilität der Endgeräte oder auch

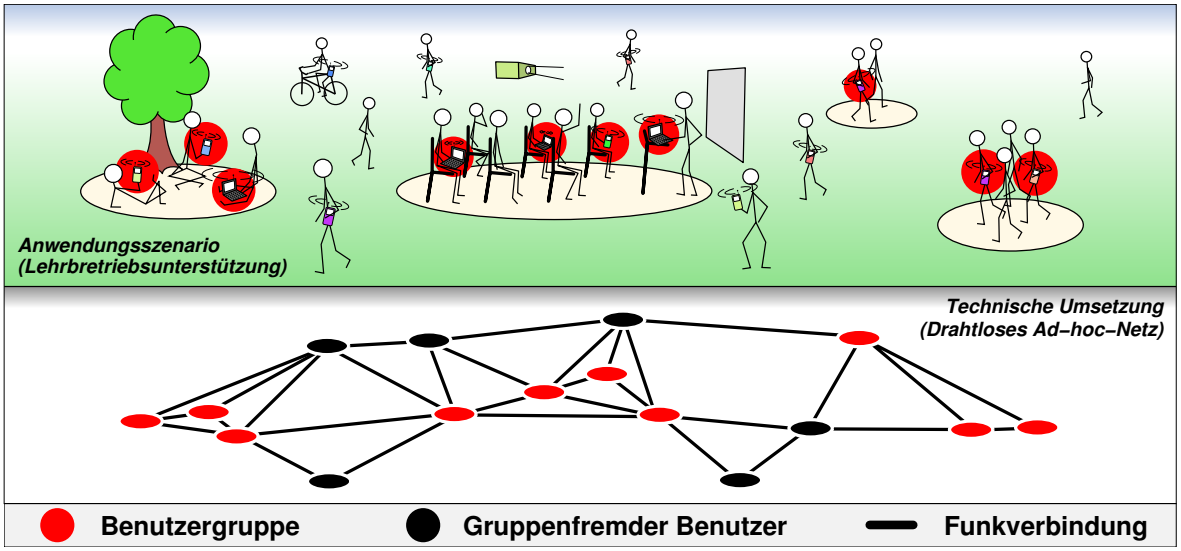


Abb. 1. Ad-hoc-Netz-basierte Gruppenkommunikation zur Lehrbetriebsunterstützung

der Behandlung eventueller Gruppenmitgliedsausfälle. Von ebenso großer Bedeutung ist die Berücksichtigung des in MANETs vorhandenen geteilten Mediums bzw. der bestmöglichen Nutzung von dessen Broadcast-Eigenschaft.

2. Der Dienst soll nicht auf den Einsatz mit bestimmten Anwendungen beschränkt sein. In der Tat haben die über Gruppendiensten betriebenen Anwendungen oftmals stark variierende Anforderungen, beispielsweise an die gebotene Zuverlässigkeit des Datenaustauschs. Um in diesem Zusammenhang unterschiedliche Anwendungen unterstützen zu können, muss der Gruppendienst flexibel sein und darf folglich keinerlei Einschränkungen bzgl. der Integration von Mechanismen, etwa hinsichtlich der genannten Zuverlässigkeit, vornehmen.

Hinsichtlich der gewünschten Flexibilität bilden sogenannte *P2P-basierte Protokolle* einen viel versprechenden Ansatz. Diese realisieren die Funktionalität des Dienstes ausschließlich in den Gruppenmitgliedern, so dass sie einfach auf bestimmte Anwendungen zugeschnitten werden kann. Die Gruppenmitglieder sind dabei über ein sogenanntes *Overlay-Netz* verbunden, über welches die Kommunikation erfolgt. Endgeräte, welche nicht an der Gruppenkommunikation teilnehmen, werden somit nicht durch Gruppen-spezifische Aufgaben belastet.

1.2. Parameter zur simulativen Protokollauswertung

Die in diesem Bericht präsentierten Ergebnisse wurden mit der Netzwerksimulationsumgebung GloMoSim [24] gewonnen. Dabei wurden Szenarien betrachtet, in welchen sich 100 Endgeräte auf einer Fläche von 1000 x 1000 m^2 mit Fußgängerge-

schwindigkeiten zwischen 1 und 2 *m/s* zufällig auf Basis des *Random Direction Mobility Model* bewegen. Diesem Grundnetz werden in Abhängigkeit des untersuchten Szenarios bis zu 50 weitere Endgeräte hinzugefügt, welche die letztendlich kommunizierende Gruppe bilden. Die Gruppenmitglieder bewegen sich gemäß des *Reference Point Group Mobility* Modells [17] in kleinen Verbänden aus bis zu 5 Gruppenmitgliedern mit einer Geschwindigkeit zwischen 0,5 und 1,5 *m/s*. Alle Endgeräte tätigen ihren Medienzugriff über das IEEE 802.11b Protokoll mit einer Bandbreite von 2 *Mbit/s*. Sofern nicht anders erwähnt, beträgt die Übertragungsreichweite der Endgeräte 175 *m*. Zur Herstellung von Multi-hop-Verbindungen wird das reaktive Routing-Protokoll AODV [21] verwendet.

2. Modulare P2P-basierte Gruppendienste

In diesem Abschnitt werden ausgewählte Konzepte, die im Rahmen des MAMAS-Projekts zur Erfüllung der gestellten Anforderungen entwickelt wurden, zusammen mit entsprechenden Auswertungen kurz vorgestellt. Weiterführende Informationen können den angeführten Referenzen entnommen werden.

2.1. Die Modulare Architektur für Application-Layer Multicast

Die *Modulare Architektur für Application-Layer Multicast* (kurz MAAM Architektur [10], vgl. Abbildung 2) zerlegt einen P2P-basierten Gruppendienst in unterschiedliche Protokollkomponenten. Diese werden in *austauschbaren und miteinander kombinierbaren Modulen* integriert, so dass der gebotene Gruppendienst eine hohe Flexibilität zur Erfüllung unterschiedlicher Anwendungsanforderungen aufweist. Während der genaue Aufbau der Architekturmodule sowie deren Schnittstellen in [4, 7] genau erläutert wird, soll an dieser Stelle lediglich eine kurze Beschreibung der Aufgabenbereiche der einzelnen Komponenten erfolgen.

- *Data Processing Algorithms*: Diese Komponente integriert Daten-verarbeitende Algorithmen einer Anwendung, wie etwa (De)Kompressionsverfahren für digitalisierte Sprachdaten. Benutzerinteraktionen über graphische Oberflächen sind nach oben durch eine lokale Socket-Schnittstelle gekapselt. Diese Auftrennung ist für den System-unabhängigen Betrieb der Architektur interessant: Kompressionsverfahren können nicht nur durch reale Anwendungen angesteuert, sondern auch innerhalb von Simulationsumgebungen betrieben werden, so dass sich Untersuchungen zur erzielten Sprachqualität einfach durchführen lassen.
- *Transport Base*: Diese Komponente kapselt die Zuverlässigkeitsmechanismen, welche zur Erkennung und Behebung von Paketverlusten herangezogen werden. Diese Zwischenschicht ist in der Regel auf eine bestimmte Anwendung zugeschnitten und umfasst Regeln für Paketformate sowie ARQ-Verfahren.

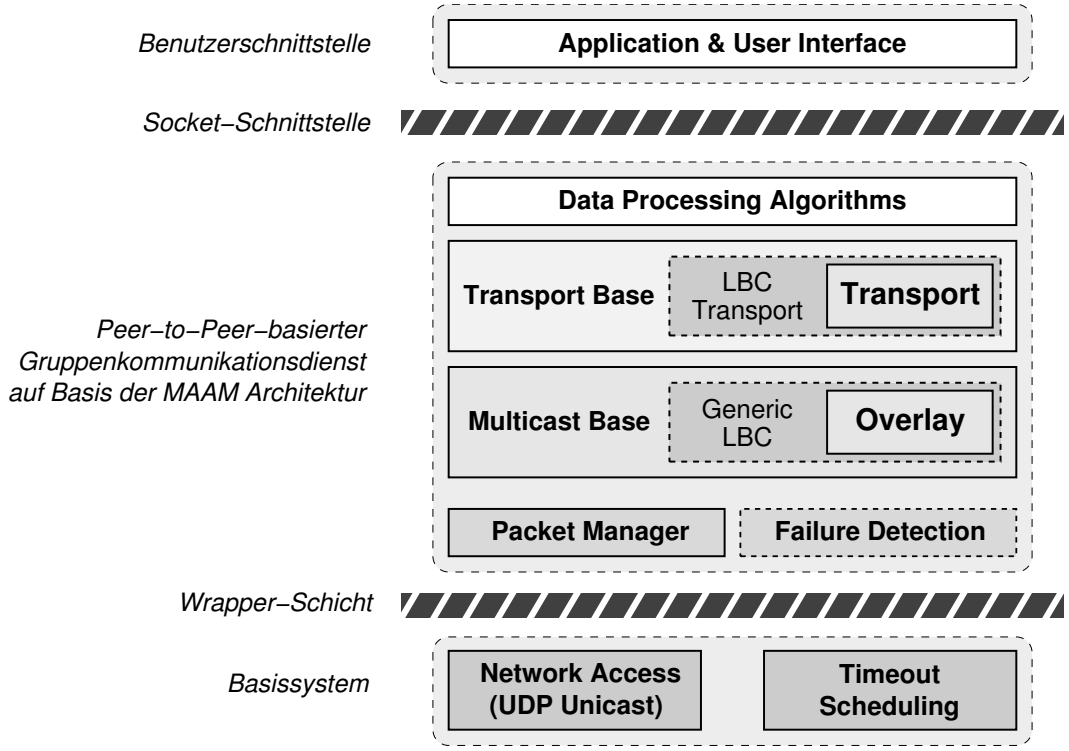


Abb. 2. Die Modulare Architektur für Application-Layer Multicast

- *Multicast Base*: Diese Komponente integriert Methodiken zur Gruppenverwaltung bzw. sorgt für die Konnektivität der Mitglieder einer Gruppe. Hauptbestandteil sind Overlay-Strukturen bzw. die Integration von Overlay-Multicast-Algorithmen und deren Routing-Mechanismen.
- *Packet & Traffic Manager*: Diese Komponente beinhaltet Verfahren zur Verwaltung von Paketen. So können unterschiedliche Paketwarteschlangen integriert werden, die zur Einsparung von Medienzugriffen eine für die übrigen Protokollkomponenten transparente Aggregation von Paketen mit gleicher Zieladresse vornehmen, wie sie beispielsweise in Abschnitt 2.4. zum Einsatz kommt.

Über eine Wrapper-Schicht wird die MAAM Architektur an ein sogenanntes Basissystem angebunden. Darunter wird eine Grundstruktur verstanden, welche der Architektur sowohl einen Netzwerkzugang (*Network Access*) in Form von UDP Unicast, als auch die Möglichkeit zur Verwaltung von Zeitgebern (*Timeout Scheduling*) bietet. Durch Kapselung der Systemabhängigkeiten innerhalb der Wrapper-Schicht erlangt die Architektur und ihre Komponenten selbst Systemunabhängigkeit, so dass sie nicht nur auf realen Betriebssystemen sondern auch innerhalb Ereignisbasierter Netzwerksimulationsumgebungen (wie beispielsweise GloMosim, ns2, OMNET, etc.) betrieben werden kann.

2.2. TrAM - Eine effiziente Overlay-Topologie für Ad-hoc-Netze

Bei TrAM (*Tree-based Overlay Architecture for MANETs*) handelt es sich um ein Protokoll zum Aufbau einer Baum-förmigen Overlay-Topologie [5, 18], welche von allen Gruppenmitgliedern als Verteilbaum genutzt werden kann. Durch die resultierende, geringe Anzahl erforderlicher Transportverbindungen ($n-1$ bei n Gruppenmitgliedern) wird besonders in Kombination mit reaktiven Routing-Protokollen, deren Overhead direkt von der Anzahl verwendeter Transportverbindungen abhängt, eine beträchtliche Entlastung des Netzes erreicht.

TrAMs Baumstruktur entsteht, indem Gruppenmitglieder im Zuge ihres Gruppenbeitritts ein bereits beigetretenes Gruppenmitglied ausfindig machen. Durch Aufbau einer Transportverbindung zu diesem Mitglied (welches somit zum *Elternknoten* des beitretenden Gruppenmitglieds wird), integriert sich das neue Gruppenmitglied in die Baumstruktur. Die Suche nach einem Elternknoten wird von jedem Gruppenmitglied periodisch wiederholt, so dass die Overlay-Topologie eine eventuelle Mobilität von Endgeräten berücksichtigt.

Die Besonderheit des TrAM-Protokolls liegt in dem zur Bestimmung des Elternknotens eingesetzten Verfahren. Herkömmliche Protokolle bewerten einzelne Gruppenmitglieder durch Abstandsmessungen anhand expliziter Ping-Pakete [1, 15]. Letztere wirken sich in Kombination mit reaktiven Routing-Protokollen nachteilig aus, da eine Route zum Zielgerät zunächst aufgebaut und das Netz hierfür geflutet werden muss. Zum Finden eines guten Elternknotens müssen üblicherweise Abstände zu mehreren Gruppenmitgliedern bestimmt werden, so dass der Prozess eine starke Belastung des Netzes erfordert. Bei TrAM flutet jedes Gruppenmitglied periodisch seinen derzeitigen Verbindungsstatus mit begrenzter Lebenszeit: Dieser besteht aus dem Hop-Abstand zum aktuellen Elternknoten und die Anzahl an Overlay-Hops zur Wurzel des TrAM-Baums. Ein Gruppenmitglied A, welches den Verbindungsstatus empfängt, kann aufgrund der erhaltenen Information und der verbleibenden Lebenszeit das sendende Gruppenmitglied in doppelter Hinsicht bewerten: Einerseits kann das sendende Gruppenmitglied evtl. als besserer Elternknoten für A identifiziert werden, oder andererseits kann A sich selbst als besseren Elternknoten für das sendende Gruppenmitglied erkennen. Erst in einem dieser Fälle erfolgt ein direkter Nachrichtenaustausch, so dass ein Routen-Aufbau nur bei einer mit Sicherheit stattfindenden Anpassung der Overlay-Topologie vollzogen wird. Zusätzlich können durch das Fluten des Verbindungsstatus viele Gruppenmitglieder auf einmal erreicht werden, so dass die zahlreichen, einzelnen Messungen und die damit implizierten Netzbelastungen entfallen.

Die Leistungsfähigkeit des TrAM-Protokolls im Vergleich zu den P2P-Multicast-Protokollen Narada [15], NICE [1], PAST-DM [16] und n*Unicast [6] ist in Abbildung 3 dargestellt. Die oberen Diagramme zeigen den von den Protokollen jeweils

implizierten Aufwand im Sinne von *pro Sekunde durch das Routing-Protokoll aufgebauter Routen* (links) und *pro Sekunde erfolgte Medienzugriffszeit innerhalb des gesamten Ad-hoc-Netzes* (rechts). Es ist zu erkennen, dass TrAM bei steigender Gruppenmitgliedszahl aufgrund seiner optimierten Protokollmechanismen lediglich einen Bruchteil des Aufwands von anderen Protokollen verursacht. Dies birgt insbesondere den Vorteil, dass die eingesparte Bandbreite zur Vermittlung zusätzlicher Daten oder zur Verwaltung größerer Gruppen eingesetzt werden kann. Die beiden unteren Diagramme, in welchen die Leistungsfähigkeit der Protokolle im Kontext unterschiedlicher Anwendungen (eine *CBR-Anwendung* links und eine *Chat-Anwendung* rechts) dargestellt ist, zeigen dies deutlich: So erreicht das TrAM-Protokoll eine im Vergleich zu bisherigen Protokollen stark erhöhte Paketzustellrate.

2.3. Generische Protokollverbesserungen

In diesem Abschnitt werden gesonderte Protokollmechanismen beschrieben, welche innerhalb des MAMAS-Projekts zur Leistungssteigerung von Overlay-Protokollen entwickelt wurden und einen generischen Charakter aufweisen: Sie sind folglich mit beliebigen solchen kombinierbar.

2.3.1. Das Lokale Broadcast-Clustering

In Netzarealen mit hoher Gruppenmitgliedsdichte zeigt sich eine Datenverteilung auf Basis eines Overlay als äußerst ineffizient. Dies ergibt sich aus der Tatsache, dass ein Datenpaket über das Overlay von Gruppenmitglied zu Gruppenmitglied geleitet wird. Mit steigender Gruppenmitgliedsdichte wird folglich ein steigendes Maß an Bandbreite benötigt, so dass ein solches Netzareal zu einem Flaschenhals wird und den erzielbaren Durchsatz reduziert. Um dieser Problematik entgegen zu wirken, bietet es sich an, die Broadcast-Fähigkeit des geteilten Mediums gezielt zur Datenweiterleitung zu nutzen, wie beispielsweise anhand des in diesem Projekt entwickelten Konzepts *Lokaler Broadcast Cluster* (LBCs [2, 13]).

LBCs entstehen, indem jedes dem Overlay beigetretene Gruppenmitglied periodisch *Heartbeat*-Nachrichten als Broadcast versendet. Es wird hierdurch zum Repräsentanten seines LBCs. Ein Gruppenmitglied A, des sich in Übertragungsreichweite eines LBC-Repräsentants befindet, empfängt dessen Heartbeats und nimmt hierdurch dessen Anwesenheit wahr. A tritt dem Overlay nicht bei, sondern verbleibt als *lokal beigetretenes Gruppenmitglied* innerhalb des detektierten LBCs.

Während die Verteilung von Multicast-Daten im Overlay wie gewohnt vollzogen wird, erfolgt die Weiterleitung innerhalb eines LBCs durch die jeweiligen LBC-Repräsentanten als Broadcast: Zur Übertragung eines Datenpakets an die Mitglieder eines LBCs wird folglich nur ein einziger Medienzugriff benötigt. Handelt es sich bei einer Multicast-Quelle um ein lokal beigetretenes Gruppenmitglied, so nutzt dieses

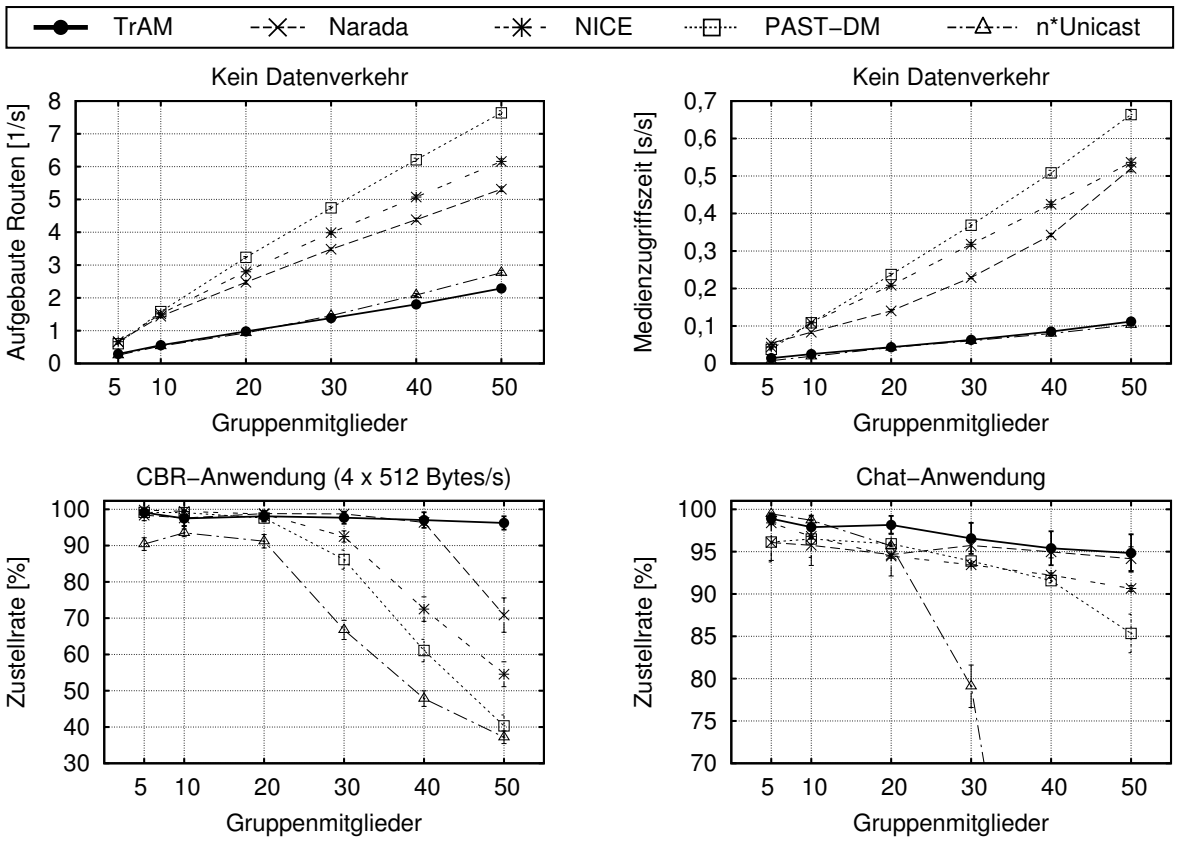


Abb. 3. Vergleich von TrAM mit anderen Overlay-Protokollen

seinen LBC-Repräsentant, um Daten an die übrigen Gruppenmitglieder zu versenden. Hierfür leitet der Repräsentant das von der lokalen Quelle empfangene Paket sowohl über seine Overlay-Verbindungen als auch in seinem LBC weiter.

Aufgrund von Mobilität kann ein lokales Gruppenmitglied seinen LBC verlassen, was sich durch das Ausbleiben der periodischen Heartbeat-Nachrichten äußert. Um Daten weiterhin beziehen zu können, ist es erforderlich, dass sich das betroffene Gruppenmitglied in das Overlay eingliedert und hierdurch selbst zu dem Repräsentant eines neuen LBCs wird. Analog hierzu kann auch die Overlay-Ausgliederung eines Gruppenmitglieds definiert werden: Befinden sich zwei LBC-Repräsentanten in gegenseitiger Übertragungsreichweite, empfangen sie gegenseitig ihre Heartbeats. Da dies wiederum einer redundanten Datenverteilung entspricht, zieht sich eines der beiden Gruppenmitglieder aus dem Overlay zurück und gliedert sich als lokales Gruppenmitglied in den verbleibenden LBC ein.

Die Auswirkungen des LBC-Konzepts auf die Leistungsfähigkeit der Overlay-Topologien sind in Abbildung 4 dargestellt. Um Vergleichsmöglichkeiten zu bieten,

wurden die Diagramme unter ansonsten gleichen Bedingungen wie für Abbildung 3 gewonnen. Im Hinblick auf den verursachten Aufwand (obere Diagramme) ist festzustellen, dass sowohl die Anzahl der aufgebauten Routen als auch die verursachte Medienzugriffszeit stark reduziert wird [3, 12, 13]. Mit steigender Gruppengröße werden die Ausmaße der erzielten Verbesserungen deutlicher, da das betrachtete Simulationsgebiet zunehmend von LBCs abgedeckt wird, und somit die Wahrscheinlichkeit lokaler Gruppenbeitritte steigt.

Im Kontext von Anwendungen (untere Diagramme) zeigt sich das LBC-Konzept vor allem bei einer CBR-Anwendung (linkes Diagramm) als gewinnbringend. So fallen die in Abbildung 3 bei steigender Gruppengröße beobachteten Einbrüche für alle Protokolle weitaus geringer aus. Aufgrund der Fehleranfälligkeit der Broadcast-Übertragungen zeigt das LBC-Konzept bei einer Chat-Anwendung (rechtes Diagramm) hingegen negative Auswirkungen. Die Übertragungswiederholungen des Medienzugriffsprotokolls, welche bei einer CBR-Anwendung zusätzliche Paketkollisionen implizierten, sorgen bei einer Chat-Anwendung hingegen für eine höhere Zuverlässigkeit. Da sich das LBC-Konzept folglich nicht in jedem Anwendungsszenario als vorteilhaft erweist [6], wurde es lediglich als *transparenter* und *optionaler* Bestandteil in die MAAM-Architektur integriert (vgl. Abbildung 2): Die letztendliche Entscheidung über die Nutzung des generischen LBC-Moduls und eines optimierten LBC-Transportmoduls bleibt der Anwendung überlassen.

2.3.2. Ausfallerkennung von Gruppenmitgliedern

Insbesondere in mobilen Umgebungen ist mit temporären Unerreichbarkeiten von Geräten oder Totalausfällen aufgrund erschöpfter Energiereserven zu rechnen. Zur Reaktion auf derartige Ereignisse integrieren Overlay-Protokolle Mechanismen zur Detektion ausgefallener Gruppenmitglieder und zur Reparatur der Overlay-Topologie. Die Detektionsmechanismen basieren auf dem konsekutiven Verlust periodisch erwarteter Kontrollnachrichten. Zur Reduktion der Netzbelastung beläuft sich die Periodizität der Kontrollnachrichten üblicherweise auf 10 bis 20 Sekunden, während die Anzahl für einen vermuteten Ausfall auszubleibender Nachrichten 4 bis 6 beträgt. Aus diesen Größen resultiert oftmals eine Detektionsdauer im Minutenbereich, während welcher die Datenverteilung über die Overlay-Topologie gestört sein kann. Da in Abhängigkeit der Anwendung Datenpakete weitaus häufiger versendet werden als die zur Ausfalldetektion eingesetzten Kontrollnachrichten, wurden zur Beschleunigung der Ausfalldetektion entsprechende Protokollmechanismen entwickelt, welche Ausfälle ebenfalls aufgrund erwarteter aber ausbleibender Datenpakete vermuten [20].

Die entwickelten Protokollmechanismen überwachen Gruppenmitglieder, von welchen derzeit Datenpakete empfangen werden. Im Zuge dieser Überwachung wird ein Verkehrsprofil erstellt, aufgrund dessen künftige Datenpakete erwartet

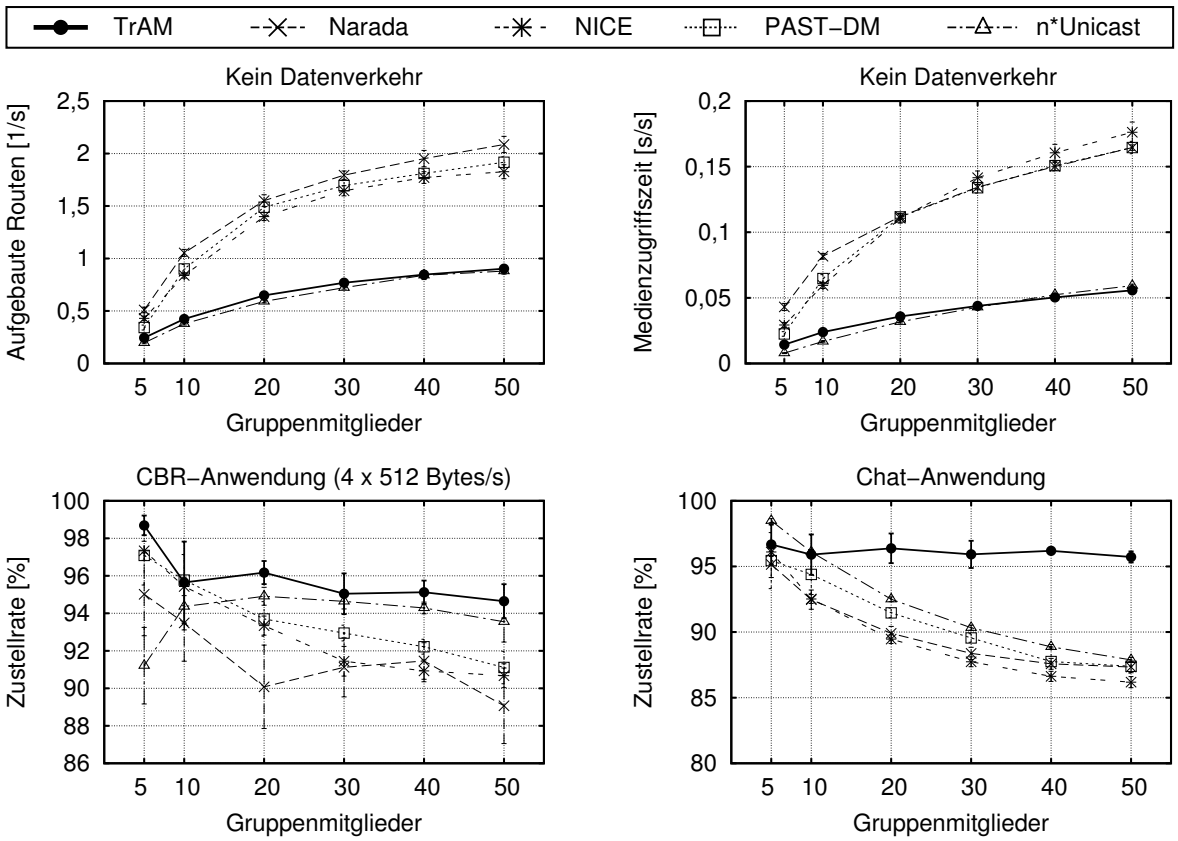


Abb. 4. Auswirkungen des LBC-Konzepts auf Overlay-Protokolle

werden. Bleiben diese Datenpakete aus, wird ein aktiver Gruppenmitgliedstest eingeleitet, bei welchem die Reaktivität des als ausgefallen vermuteten Gruppenmitglieds mittels expliziter Ping-Pakete getestet wird. Reagiert das Gruppenmitglied durch Versenden eines Pong-Pakets, wird das mit diesem Gruppenmitglied assoziierte Verkehrsprofil entsprechend angepasst. Bleiben hingegen die Ping-Anfragen fortlaufend unbeantwortet, wird das Gruppenmitglied als ausgefallen deklariert und die Reparatur der Overlay-Topologie eingeleitet.

Die Auswirkungen der erweiterten Ausfalldetektion sind für die Protokolle TrAM und Narada im Kontext einer CBR- und einer Chat-Anwendung beispielhaft in Abbildung 5 aufgezeigt. Dabei tragen die Diagramme auf der x-Achse die Reparaturdauer und auf der y-Achse den Anteil der bis zu diesem Zeitpunkt beobachteten Overlay-Reparaturen auf. Es ist zu erkennen, dass aufgrund der erweiterten Ausfalldetektion Reparaturen von Overlay-Topologien weitaus zügiger initiiert werden können. Dies gilt nicht nur für CBR-Anwendungen, die mehrere Datenpakete pro Sekunde versenden, sondern auch für Chat-Anwendungen, welche üblicherweise einen sehr unregelmäßigen Verkehr aufweisen [5, 6]. Während durch

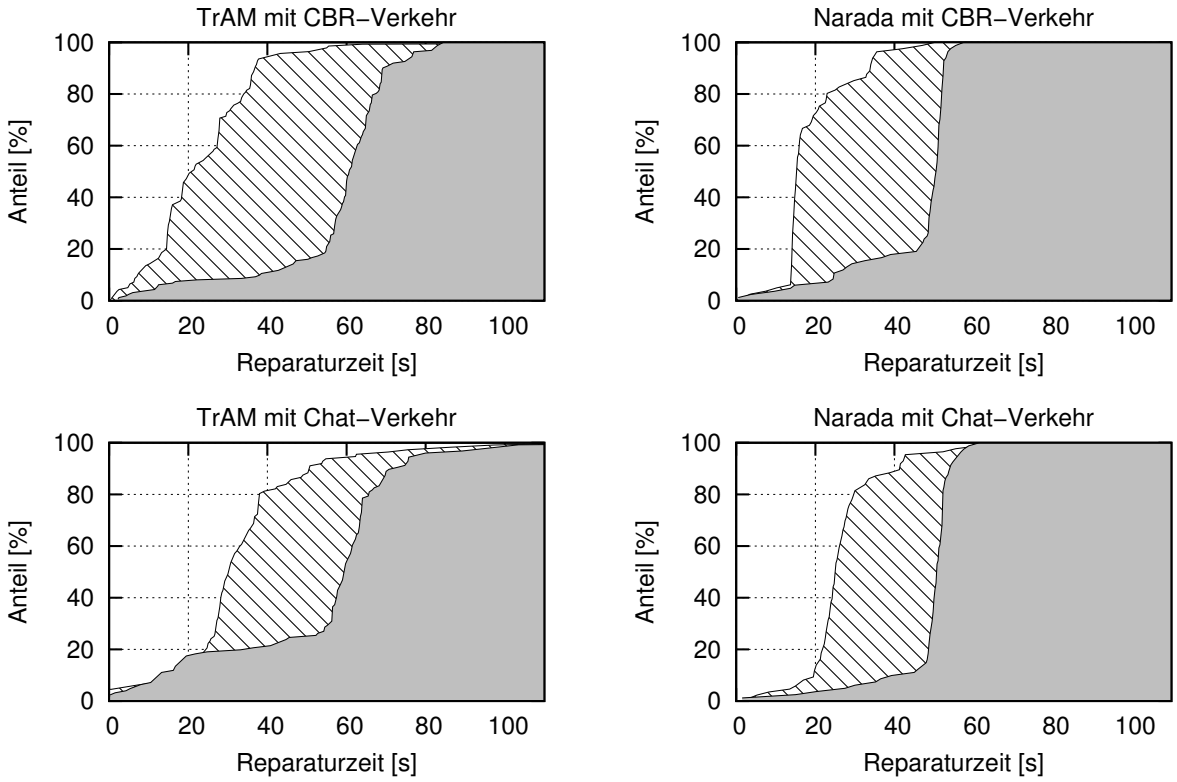


Abb. 5. Overlay-Reparaturdauer mit (▨) und ohne (▬) erweiterter Ausfalldetektion

eine (hier nicht weiter dargestellte) Verkürzung des aktiven Gruppenmitgliedstests die Detektions- und Reparaturdauer weiter reduziert werden kann, impliziert dies jedoch auch einen Anstieg der falsch erkannten Gruppenmitgliedsausfälle.

Aufgrund der zügigeren Ausfalldetektion und Overlay-Reparaturen erholt sich die für Anwendungen als kritisch erweisende Paketzustellrate nach einem Gruppenmitgliedsaufall schneller auf ein akzeptables Niveau. Dies ist in Abbildung 6 zu erkennen, in welcher im Kontext einer CBR-Anwendung Messungen mit dem TrAM-Protokoll durchgeführt wurden. Die zu den Zeitpunkten $400s$, $500s$, $600s$ und $700s$ simulierten Ausfälle lassen die Zustellrate auf knapp 40% einbrechen. Bei Einsatz der erweiterten Ausfalldetektion kann jedoch ein weitaus zügigeres Ansteigen der Zustellrate als für die herkömmliche Ausfalldetektion beobachtet werden.

2.4. Multi-Player Gaming Transport

Der modulare Aufbau der MAAM-Architektur ermöglicht die einfache Integration spezialisierter Transportprotokolle, wie [3, 13, 22], in den Gruppendienst, um unterschiedliche Anwendungen zu unterstützen. Aufgrund des in MANETs

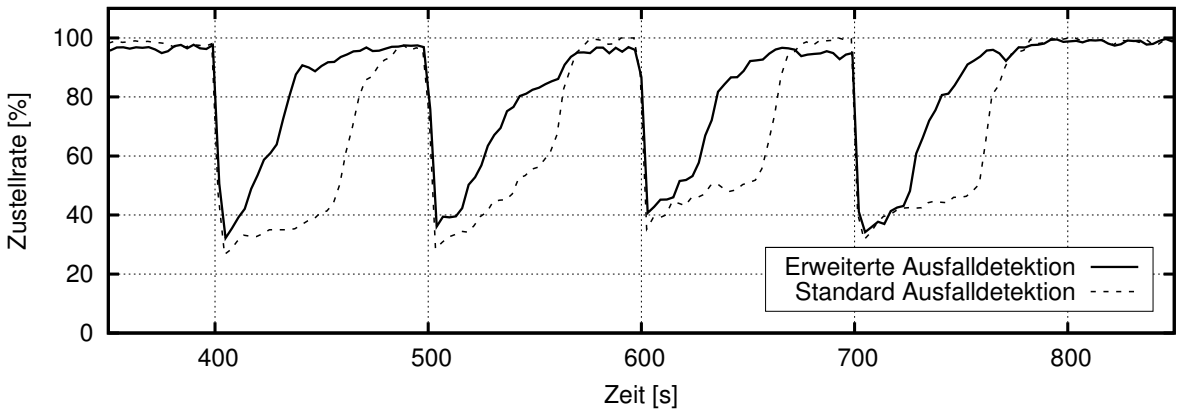


Abb. 6. Verlauf einer CBR-Zustellrate für TrAM bei mehreren Knotenausfällen

verwendeten, geteilten Mediums stellen dabei Echtzeit-basierte Anwendungen eine besondere Herausforderung dar. Um in diesem Zusammenhang die Eignung des in MAMAS konzipierten Gruppendienstes zu untersuchen, wurde beispielhaft ein *Echtzeit-basiertes Mehrbenutzerspiel* betrachtet. Dieses charakterisiert sich durch seinen typischen Datenverkehr, welcher aus von jedem Spieler alle 40 *ms* versendeter Positions- und Geschwindigkeitsinformation in einem virtuellen Raum besteht. Zusätzlich umfasst der Datenverkehr gesonderte Ereignisse, welche etwa das Abfeuern einer Waffe oder das Einsammeln eines Gegenstands beschreiben. Um einen angenehmen Spielfluss zu garantieren, wird üblicherweise gefordert, dass die Verzögerung zwischen dem Aussenden und dem Eintreffen eines Datenpakets einen Wert von ca. 150 *ms* nicht überschreiten darf [14].

Zur Unterstützung des Spiels wurde ein spezielles Transportprotokoll konzipiert und in die MAAM Architektur integriert. Das *Multi-Player Gaming Transport Protocol* (MPGT-Protokoll) berücksichtigt die Tatsache, dass Positions- und Geschwindigkeitsinformation unzuverlässig übertragen werden kann, da der sendende Spieler in der Regel aktuellere Information versenden wird, bevor ein empfangender Spieler die fehlenden Daten erkannt hat. Ereignisse werden dahin gegen zuverlässig übertragen, da sie oftmals Spiel-entscheidende Aktionen kodieren. Um die für MANETs kritische Anzahl von Medienzugriffen zu reduzieren, integriert das Protokoll Mechanismen zur Aggregation von Paketen mit gleicher Zieladresse. Diese sehen einerseits vor, dass Ereignisse nicht direkt versendet sondern bis zur nächsten periodischen Positionsinformation gepuffert und dann mit dieser zusammengefasst werden. Andererseits werden die von einem Spieler empfangenen Daten nicht direkt weitergeleitet sondern bis zum Versenden des eigenen, nächsten Pakets gepuffert und dann mit diesem zusammengefasst. Dabei werden die durch die Pufferung implizierten Verzögerungen den jeweiligen Daten angehängt, um der Anwendung

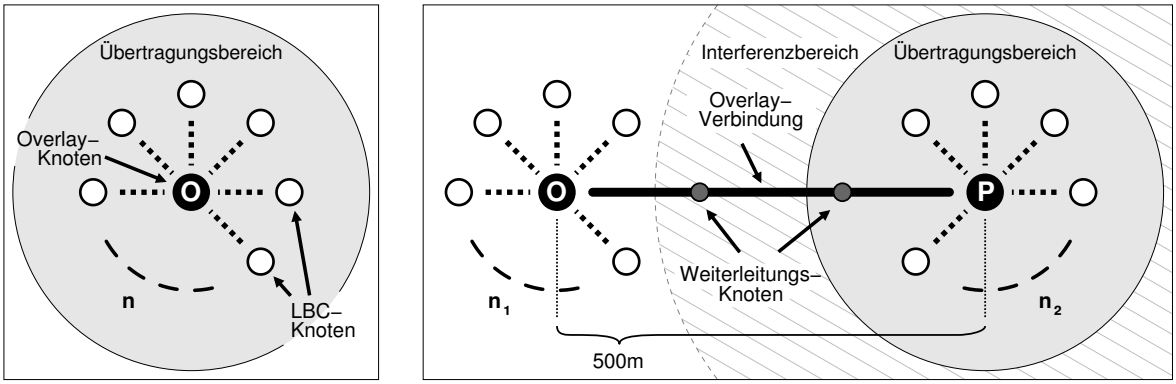


Abb. 7. Modellierte Szenarien zur Untersuchung von Mehrbenutzerspielen

eine zeitliche Extrapolation von Bewegungen zu erleichtern. Das MPGT-Protokoll sieht zusätzlich explizit eine Unterstützung des LBC-Konzepts vor, indem die Paketaggregation über der Broadcast-Adresse erfolgt.

Die Leistungsfähigkeit des MPGT-Protokolls wurde mit den in Abbildung 7 dargestellten räumlichen Spielerverteilungen untersucht. Das linke (Single-hop) Szenario bildet eine klassische LAN-Party nach, bei welcher sich die Spieler nahe beieinander befinden. Aufgrund des LBC-Konzepts bildet sich hier ein einziger Overlay-Knoten O aus, welcher einen lokalen Beitritt der übrigen Spieler ermöglicht. Bei diesem Szenario wird erwartet, dass sich mit steigender Spielerzahl n die Verzögerung der versendeten Daten erhöht und schließlich die Grenze von 150 *ms* übersteigt. Zur Unterstützung zusätzlicher Spieler kann sich eine räumliche Verteilung der Spieler (rechtes Szenario) als vorteilhaft erweisen: Hier befinden sich zwei Spielergruppen räumlich weit genug voneinander entfernt, um unter Berücksichtigung des Interferenzbereichs der Funksignale von 353 *m* ihre Kommunikation gegenseitig nicht zu beeinflussen. Die deshalb 500 *m* voneinander platzierten LBCs sind über eine Overlay-Verbindung verbunden, über welche die von O und P aggregierten Daten ausgetauscht werden. Für den Aufbau der Overlay-Verbindung werden zwei Weiterleitungs-Knoten eingesetzt, welche selbst nicht an dem Spiel teilnehmen.

Die für die Szenarien gewonnenen Messungen sind in Abbildung 8 dargestellt, wobei der Anteil der nicht rechtzeitig ausgelieferten Daten in Abhängigkeit der Spielerzahl aufgetragen ist. Dabei kann festgestellt werden, dass sich eine räumliche Verteilung der Spieler negativ auf die Anzahl unterstützbarer Spieler auswirkt. So können innerhalb des Single-hop Szenarios bis zu 21 Spieler unterstützt werden, bevor der Anteil verspäteter Daten nennenswerte Ausmaße annimmt. Dahin gegen weist das verteilte Szenario bereits ab 13 Spielern deutliche Verzögerungen auf, welche auf das Problem versteckter Endgeräte zurückzuführen sind: So hem-

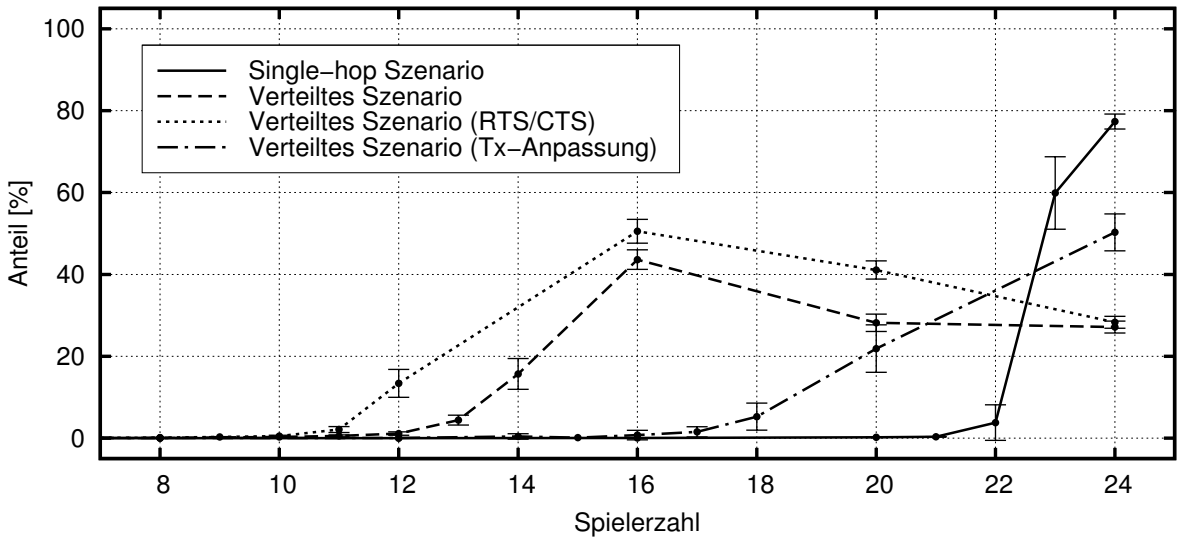


Abb. 8. Anteil der nicht rechtzeitig ausgelieferten Ereignis- und Positionsinformation

men hier die aus der Kommunikation in den LBCs resultierenden Interferenzen die Weiterleitung der aggregierten Daten über die Overlay-Verbindung. Das Aktivieren der RTS/CTS-Erweiterung des Medienzugriffsprotokolls bewirkt aufgrund zusätzlicher Medienzugriffe stärkere Interferenzen und somit ein weiteres Absinken der unterstützbaren Spielerzahl. Zur Reduktion des Ausmaßes der Interferenzen, kann die Sendeleistung Tx von LBC-Knoten auf ein Maß herabgesetzt werden, so dass der nahe gelegene LBC-Repräsentant gerade noch erreicht werden kann. Obwohl die unterstützbare Spielerzahl hierdurch ansteigt, bleibt sie dennoch deutlich hinter dem Single-hop Szenario zurück [8].

2.5. Praktische Umsetzungen und Demonstratoren

Auf Basis der in den vorherigen Abschnitten genannten Protokolle und insbesondere anhand der Plattformunabhängigkeit der MAAM-Architektur konnten innerhalb des MAMAS-Projekts unterschiedliche Demonstratoren nicht nur selbst sondern auch in Zusammenarbeit mit anderen Projekten entwickelt werden. Zur Demonstration des Projekt-intern entwickelten Gruppendienstes wurden Anwendungen selbst entwickelt (vgl. Abbildung 9) und vorgestellt [9]. Zu diesen zählen:

– Eine *grafische Chat-Anwendung* zur Text-basierten Interaktion zwischen den Gruppenmitgliedern. Gruppenmitglieder können sich nach der Wahl eines Avatars und Nicknames mittels Texteingabe unterhalten. Durch Betätigung der Cursor-Tasten können sie sich zusätzlich innerhalb eines virtuellen Chat-Raums bewegen und sich dort begegnen.

Abb. 9. Screenshot und Vorführung des MAMAS-Gruppendienstes auf der MobiSys 2007

- Eine *Slideshow-Anwendung* anhand welcher Vorlesungs- oder Seminarpräsentationen durchgeführt werden können. Die Anwendung überträgt die von einzelnen Gruppenmitgliedern über einen Dateiauswahlsdialog bestimmten Foliensätze zu den übrigen Gruppenmitgliedern. Anhand entsprechender Pfeiltasten können Benutzer ihre Präsentation abhalten, indem sie durch Drücken der Pfeiltasten zwischen Folien hin- und herwechseln.
- Eine *Voice-over-IP-Anwendung* zur Sprach-gestützten Unterhaltung von Gruppenmitgliedern. Zur Kompression der Sprachdaten sind in die Anwendung unterschiedliche, frei verfügbare Kompressionsverfahren integriert. Beim Start der Anwendung kann der Benutzer das zu verwendende Verfahren über eine Dialogbox auswählen.
- Ein *Echtzeit-Mehrbenutzerspiel*, bei welchem die Gruppenmitglieder in einem virtuellen Sonnensystem anhand von Raumschiffen gegeneinander antreten. Aufgrund seiner Echtzeit-Anforderungen zeigt sich diese Anwendung als sehr fordernd gegenüber der zwischen Gruppenmitgliedern stattfindenden Kommunikation. Es basiert daher auf dem speziellen, in Abschnitt 2.4. beschriebenen Transportprotokoll.

In Kooperation mit den ebenfalls im Schwerpunktprogramm geförderten Projekten DIANE [19] und AHSM [23] konnte eine Anwendung zur P2P-basierten Verwaltung und Nutzung semantisch beschriebener Dienste entwickelt werden [11]. Die Software vereint die Konzepte der einzelnen Projekte wie die semantische Dienstbeschreibung, die Diensteverwaltung und die P2P-basierte Anfragevermittlung. Sie ermöglicht dem Benutzer somit, Dienste anderen Benutzern anbieten zu können, oder wiederum von anderen Benutzern angebotene Dienste lokalisieren und nutzen zu können. Die beispielhaft umgesetzte Anwendung ermöglicht den Tausch semantisch beschriebener Dokumente in einem verteilten Campus-Szenario.

3. Zusammenfassung

Innerhalb des Forschungsprojekts MAMAS wurde ein P2P-basierter Gruppendienst für MANETs konzipiert. Er unterstützt die Anforderungen unterschiedlicher Anwendungen, indem er auf einer modularen Architektur basiert, welche eine flexible Dienstkomposition aus optimierten Protokollbausteinen gestattet. Auf diese Weise können auf eine bestimmte Anwendung zugeschnittene Protokolle gezielt in den Dienst integriert und genutzt werden. Zur Verteilung von Daten innerhalb des Gruppendienstes wurde unter Berücksichtigung der Eigenschaften von MANETs eine Overlay-Topologie entwickelt, welche die Effizienz anderer Protokolle deutlich übertrifft. Darüber hinaus konnten generische Protokollmechanismen zur Steigerung der Leistungsfähigkeit des Dienstes entwickelt werden. Durch Abstraktion des Betriebssystems zeigt sich der konzipierte Gruppendienst als Plattformunabhängig, so dass er in Netzwerksimulationsumgebungen aber auch auf realen Betriebssystemen betrieben werden kann. So konnten unterschiedliche Anwendungen auf Basis des Dienstes entwickelt und vorgestellt werden. Insbesondere konnte auch in Kooperation mit anderen Projekten des Schwerpunktprogramms ein Demonstrator zur P2P-basierten, semantischen Dienstnutzung entwickelt werden.

Literaturverzeichnis

[1] BANERJEE, S., B. BHATTACHARJEE und C. KOMMAREDDY: *Scalable Application Layer Multicast*. In: *ACM SIGCOMM 2002*, Pittsburgh, PA, USA, Juni 2002.

[2] BAUMUNG, P.: *Implementierung und Evaluation eines hierarchischen Endsystem-Multicast-Protokolls für mobile Ad-hoc-Netze*. Diplomarbeit, Universität Karlsruhe (TH), August 2003.

[3] BAUMUNG, P.: *Stable, Congestion-Controlled Application-Layer Multicasting in Pedestrian Ad-hoc Networks*. In: *Proceedings of the IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, Seiten 57–64, Taormina, Italy, Juni 2005.

[4] BAUMUNG, P.: *On the Modular Composition of Scalable Application-Layer Multicast Services for Mobile Ad-hoc Networks*. In: *Proceedings of the 2006 International Workshop on Wireless Ad-hoc and Sensor Networks (IWWAN 2006)*, New York, USA, Juni 2006.

[5] BAUMUNG, P.: *TrAM: Cross-Layer Efficient Application-Layer Multicast in Mobile Ad-hoc Networks*. In: *Proceedings of the IEEE Wireless Communications and Networking Conference 2007 (WCNC 2007)*, Hong Kong, China, März 2007.

[6] BAUMUNG, P.: *Application-Layer Multicast in MANETs: To Broadcast or not to Broadcast?* In: *Proceedings of the Fifth Annual Conference on Wireless On demand Network Systems and Services*, Seiten 133–140, Garmisch-Partenkirchen, Germany, Januar 2008.

[7] BAUMUNG, P.: *P2P-basierte Gruppenkommunikation in drahtlosen Ad-hoc-Netzen*. Doktorarbeit, Institut für Telematik, Universität Karlsruhe (TH), ISBN 978-3-86644-268-9, Universitätsverlag Karlsruhe, 2008.

[8] BAUMUNG, P.: *Banishing Patch-Cables from WLAN Parties – Using P2P Multicast as a Substitute?* In: *Kommunikation in Verteilten Systemen*, Kassel, Germany, März 2009.

[9] BAUMUNG, P. et al.: *Modular P2P Multicast in Wireless Ad-hoc Networks*. http://www.sigmobile.org/mobisys/2007/poster.html, *(Überprüft im Februar 2008)*.

[10] BAUMUNG, P. et al.: *The Modular Architecture for Application-Layer Multicast*. http://maam.pcb-net.org, *(Überprüft im Februar 2008)*.

[11] BAUMUNG, P., S. PENZ und M. KLEIN: *P2P-Based Semantic Service Management in Mobile Ad-hoc Networks*. In: *Proceedings of the Seventh International Conference on Mobile Data Management (MDM 2006)*, Nara, Japan, Mai 2006.

[12] BAUMUNG, P. und M. ZITTERBART: *Effiziente Endsystem-basierte Multicast-Dienste in mobilen Ad-hoc Netzen*. In: *Lecture Notes in Informatics*, 18. DFN-Arbeitstagung, Düsseldorf, Germany, Juni 2004.

[13] BAUMUNG, P., M. ZITTERBART und K. KUTZNER: *Improving Delivery Ratios for Application Layer Multicast in Mobile Ad-hoc Networks*. Elsevier Special Issue on Computer Communications, 28(14):1669–1679, 2005.

[14] BEIGBEDER, T., R. COUGHLAN, C. LUSHER, J. PLUNKETT, E. AGU und M. CLAYPOOL: *The Effects of Loss and Latency on User Performance in Unreal Tournament 2003*. In: *Proceedings of ACM Network and System Support for Games Workshop (NetGames)*, Portland, OG, USA, September 2004.

[15] CHU, Y., S. G. RAO und H. ZHANG: *A Case for End System Multicast*. In: *ACM SIGMETRICS 2000*, Santa Clara, California, USA, Juni 2000.

[16] GUI, C. und P. MOHAPATRA: *Efficient Overlay Multicast for Mobile Ad hoc Networks*. In: *The Wireless Communications and Networking Conference (WCNC)*, Seiten 1118–1123, New Orleans, Louisiana, USA, März 2003.

[17] HONG, X., M. GERLA, G. PEI und C. CHIANG: *A Group Mobility Model for Ad Hoc Wireless Networks*. In: *Proceedings of ACM International Workshop on Modelling and Simulation of Wireless and Mobile Systems*, Seiten 53–60, Seattle, USA, August 1999.

[18] KRÄMER, B.: *Entwicklung und Evaluation einer effizienten Overlay-Struktur zur Endsystembasierten Gruppenkommunikation in mobilen Ad-hoc Netzen*. Diplomarbeit, Universität Karlsruhe (TH), August 2005.

[19] KÖNIG-RIES, B. et al.: *http://hnsp.inf-bb.uni-jena.de/diane/*.

[20] MARTIN, D.: *Detektion und Behandlung von Knotenausfällen bei Overlay-Multicast in drahtlosen Ad-hoc-Netzen*. Diplomarbeit, Institut für Telematik, Universität Karlsruhe (TH), Januar 2008.

[21] PERKINS, C., E. BELDING-ROYER und S. DAS: *Ad hoc On-Demand Distance Vector (AODV) Routing*. RFC 3561 (Experimental), Juli 2003.

[22] SCHLAGER, T.: *Entwurf und Evaluation eines drahtlosen Ad-hoc-Transportprotokolls für Overlay-Multicast mit Unterstützung lokaler Broadcast-Cluster*. Diplomarbeit, Institut für Telematik, Universität Karlsruhe (TH), Januar 2008.

[23] SPANIOL, O. und S. PENZ: *http://www.nets.rwth-aachen.de/ahsm/*.

[24] ZENG, X., R. BAGRODIA und M. GERLA: *GloMoSim: a Library for Parallel Simulation of Large-scale Wireless Networks*. In: *Proceedings of the 12th Workshop on Parallel and Distributed Simulation*, Banff, Alberta, Canada, Mai 1999.

PBM – Positionsbasierter Multicast für mobile Ad-Hoc-Netze

Sascha Schnaufer, Matthias Transier und Wolfgang Effelsberg

Lehrstuhl für Praktische Informatik IV, Universität Mannheim

Zusammenfassung – Die spontane Vernetzung von mobilen Endgeräten bietet ein großes Potential für neuartige gruppenbasierte Anwendungen. Aufgrund der limitierten Bandbreite und den besonderen Eigenschaften eines solchen mobilen Netzwerkes ist hierfür ein spezielles Multicast-Protokoll notwendig. Deshalb war es das Ziel des PBM-Projekts, erstmals ein positionsbasiertes Multicast-Routing-Protokoll für mobile Ad-Hoc-Netze zu entwerfen und anschließend zu simulieren und zu evaluieren.

1. Einleitung

Mit diesem Bericht möchten wir einen kurzen Überblick über die wichtigsten Ergebnisse des Projektes geben. In Abschnitt 2. wird der erste Versuch beschrieben, bestehende positionsbasierte Unicast-Protokolle zu Multicast-Protokollen zu erweitern. Hierbei zeigte sich die Notwendigkeit, neue Verfahren zur positionsbasierten Gruppenverwaltung zu entwickeln, die auch bei hoher Mobilität und großen Gruppengrößen noch skalieren. Diese Ergebnisse führten zu dem in Abschnitt 3. beschriebenen Protokoll *Scalable Position-Based Multicast Routing* (SPBM). In einer umfangreichen Simulationsstudie konnte gezeigt werden, dass SPBM wesentlich bessere Skalierungseigenschaften mit sich bringt als das bis dato als Referenzprotokoll geltende ODMRP. In Kooperation mit der Universität Düsseldorf wurde die dort entwickelte Überlastkontrolle für den Multicast-Fall erweitert und in Zusammenarbeit mit SPBM evaluiert.

Bei SPBM entscheidet der Sender explizit, welche 1-Hop-Nachbarn das Paket in Richtung der Gruppenmitglieder weiterleiten sollen. In Abschnitt 4. wird mit *Contention-Based Multicast Forwarding* (CBMF) eine alternative Strategie vorgestellt. Hierbei wird der Weiterleiter durch einen verteilten Wettbewerb unter den 1-Hop-Nachbarn ausgewählt. Anders als bei den meisten anderen Verfahren wird bei diesem Wettbewerb die Position der direkten Nachbarn nicht benötigt. Hierdurch kann der regelmäßige Austausch von Positionsdaten, der SPBM und ähnliche Protokolle als Overhead belastet, eingespart werden. In Abschnitt 5. wird ein kurzer Überblick gegeben, welche Software während des Projekts entstanden ist. Besonders herauszuheben ist hierbei die Implementierung von SPBM und CBMF als Linux-Kernel-Module, die den realen Einsatz der Protokolle ermöglichen.

Alle positionsbasierten Routing-Protokolle, ob Unicast oder Multicast, versuchen, mit Hilfe von Positionen einen Pfad vom Sender zum Ziel zu finden. Allerdings wird durch Radiohindernisse der hierbei genutzte Zusammenhang zwischen räumlicher Nähe und Kommunikationsmöglichkeit stark reduziert. In Abschnitt 6. wird dieser Umstand näher untersucht und evaluiert. Es zeigte sich, dass die bisher etablierten Strategien in vielen Fällen durch Radiohindernisse stark beeinträchtigt werden. Wir entwickelten mit *Greedy Routing with Abstract Neighbor Table* (GRANT) ein neues Verfahren, um die Eignung eines potentiellen Weiterleiters besser zu bestimmen. Diese Strategie kann nun in Unicast-Protokollen oder in SPBM verwendet werden, um die Performance in Gebieten mit Radiohindernissen signifikant zu steigern.

2. Das Position-Based Multicast Routing Protocol

Das Ziel zu Beginn des Schwerpunktprogramms war es, erstmals ein rein positionsbasiertes Protokoll für Multicast-Routing in mobilen Ad-hoc-Netzwerken zu entwerfen. Dem ersten Ansatz lag eine Verallgemeinerung des positionsbasierten Unicast-Routings, wie es von *Greedy-Routing-Scheme* (GRS) [3] und FACE-2 [2] bekannt ist, zugrunde. Hieraus entstand das *Position-Based Multicast Routing Protocol* (PBM) [10] [11] [12]. Statt der Position eines einzelnen Empfängers trägt der Sender bei PBM die Adressen aller Empfänger in den Kopf der Datenpakete ein. Als Voraussetzung hierfür müssen dem Sender alle Empfänger sowie deren aktuelle Positionen bekannt sein. Die Verwaltung dieser Informationen ist nicht Bestandteil von PBM. Bei der Verallgemeinerung auf Multicast existieren zwei Kernprobleme, die zu lösen waren. Dies ist zum Einen die Frage, wann die Menge der Empfänger im Paketkopf geteilt und das Datenpaket dupliziert oder vervielfältigt werden soll, und zum Anderen das Problem der Recovery-Strategie für lokale Optima, die bei GRS auftreten können.

Der Zeitpunkt, zu dem der Pfad eines Pakets aufgespalten werden soll, lässt sich nach zwei Optimierungskriterien bewerten. Auf der einen Seite ist es das Ziel, möglichst kurze Pfade für jeden Empfänger zu erreichen. Dazu ist es notwendig, die Pfade zu den einzelnen Empfängern früh zu trennen und für jeden Empfänger den jeweils optimalen Nachbarn zur Weiterleitung auszuwählen. Auf der anderen Seite steht dem jedoch das Ziel entgegen, dass der Pfad eines Pakets möglichst lange für verschiedene Empfänger zusammen gehalten werden sollte, um die Anzahl der Einzelübertragungen zu minimieren und damit Bandbreite zu sparen. Im Ansatz des Mannheimer Protokolls PBM wurde eine Zielfunktion aufgestellt, die diese beiden Ziele miteinander kombiniert und so einen Mittelweg findet, der bei annehmbar kurzen Pfaden ausreichend Einzelübertragungen einspart. Die Formel der Zielfunktion und alle weiteren Einzelheiten hierzu finden sich in [10] [11] [12].

Für alle positionsbasierten Routing-Protokolle gilt, dass eine so genannte Recovery-Strategie definiert werden muss, die verwendet wird, wenn kein Nachbar vorhanden sein sollte, der sich näher am Ziel befindet als der aktuelle Knoten. Bei der Fragestellung nach einer geeigneten Recovery-Strategie für das positionsbasierte Multicast-Routing wurde der Ansatz verfolgt, bekannte Verfahren aus dem Bereich des positionsbasierten Unicast-Routings für die Anwendung bei Multicast zu verallgemeinern. Für Ziele im Paket, für die kein weiterleitender Nachbar mit geographischem Fortschritt gefunden werden konnte, wird der so genannte Perimeter-Modus [6] aktiviert. Dieser versucht, anhand des planarisierten Netzwerkgraphen mit der aus der Graphentheorie bekannten Rechte-Hand-Regel einen Weg um die geographische Region, in der sich keine geeigneten Weiterleitungsknoten aufhalten, zu finden. Für die hierbei benötigte Planarisierung kann die verteilte Berechnung des *Gabriel-Graph* (GG) [5] oder des *Relative-Neighborhood-Graph* (RNG) [18] verwendet werden. Sobald ein Paket im Perimeter-Modus näher an einem Ziel ist als der Knoten, der es in den Perimeter-Modus versetzt hat, wird es wieder auf die herkömmliche Art und Weise (GRS) weitergeleitet. Dabei ist es möglich, dass ein Paket gleichzeitig Ziele beider Klassen enthält, so dass ein weiterleitender Knoten bei der Entscheidung nach zwei unterschiedlichen Verfahren vorgehen muss.

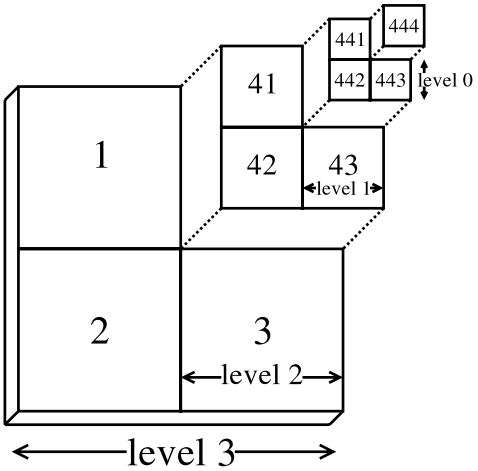
Die Entwicklung von PBM hat gezeigt, dass es möglich ist, Multicast auf Basis von Positionsinformationen durchzuführen. Die Schwierigkeit bei diesem Ansatz ist jedoch, dass ein Sender zum Einen alle Empfänger kennen und zum Anderen auch noch über deren aktuelle Position informiert sein muss. Des Weiteren müssen alle verbleibenden Empfängerpositionen im Kopf der Multicast-Pakete mitgeführt werden. Durch diese Umstände ist der Einsatz mit großen Multicast-Gruppen problematisch. Die bisherigen Erkenntnisse wurden genutzt, um ein fortschrittlicheres Protokoll zu entwickeln, welches im nächsten Abschnitt vorgestellt wird.

3. Scalable Position-Based Multicast Routing

Scalable Position-Based Multicast Routing (SPBM) [24] [25] setzt sich aus zwei Bestandteilen zusammen. Auf der einen Seite ist dies die aggregierte Verwaltung von Gruppenmitgliedschaften im Netz, und auf der anderen Seite ein hierarchischer Ansatz für die Weiterleitung von Multicast-Nachrichten.

Die Algorithmen zur Gruppenverwaltung und Weiterleitung nutzen die Struktur eines Quadtrees aus (Abbildung 1(a)). Das definierte Gebiet des Netzwerks wird dabei in vier gleiche Quadrate eingeteilt, diese jeweils wieder in vier Unterquadrate usw., bis die Größe eines Quadrats der Sendereichweite eines Knotens entspricht. Die Gruppeninformationen werden in diesen Gebieten in regelmäßigen Abständen verteilt und aggregiert, so dass bekannt ist, ob sich in einem Quadrat Empfänger einer bestimmten Multicast-Gruppe aufhalten. Die Idee dabei ist, dass ein Knoten

über Gruppenmitgliedschaften in entfernten Gebieten immer nur die aggregierten Informationen vorhalten muss. Somit benötigt ein Sender nicht mehr die Position jedes einzelnen Empfängers. Möchte eine Quelle nun Daten an eine Gruppe versenden, schlägt sie in ihrer Mitgliedschaftstabelle die Quadrate nach, in denen Knoten an Nachrichten dieser Gruppe interessiert sind, und trägt diese als Empfänger in das Datenpaket ein. In Abbildung 1(b) ist die Mitgliedschaftstabelle eines Knotens aus dem Quadrat vier dargestellt. Der Knoten kennt somit die Gruppenmitgliedschaften der Unterquadrate von Quadrate vier im Detail, die Mitgliedschaften in den Quadraten eins, zwei und drei aber nur in aggregierter Form. In diesem Beispiel wird jede der acht Gruppen durch ein Bit in der zweiten Spalte der Tabelle repräsentiert. Möchte der Knoten nun an die Gruppe fünf senden, so trägt er in den Paketkopf die Quadrate einundvierzig, zweiundvierzig und eins ein.



(a) Unterteilung in einen Quadtree

Quadrat	Gruppen
1	00011100
2	01000100
3	10100010
41	01010000
42	00010101
43	00100100
441	00000100
443	00010000
444	00100100

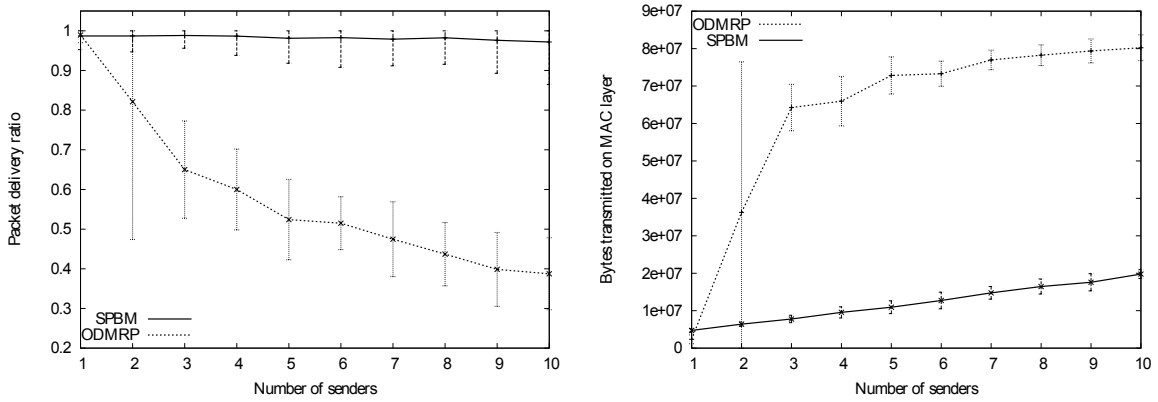
(b) Beispiel einer Mitgliedschaftstabelle

Abb. 1. Gruppenverwaltung und Weiterleitung nutzen einen Quadtree

Die Weiterleitung folgt dann dem durch GRS bekannten Prinzip. Für jedes in einem Paket enthaltene Ziel wird der am besten geeignete Nachbar als weiterleitender Knoten ausgewählt. Ziele für den gleichen Nachbarn werden zu einem Paket zusammengefasst und versendet. Die Ziele sind im Fall von SPBM nicht mehr einzelne Knoten, sondern Quadrate, die Empfänger enthalten. Erreicht ein Paket eines der Zielquadrate, so haben die Knoten dort dank der hierarchischen Gruppenverwaltung nähere Information über die Verteilung der Empfänger innerhalb dieses Quadrates und können die Zielinformationen im Paketkopf weiter verfeinern. Im weiter oben beschriebenen Beispiel würden dann die Knoten im Quadrat eins wis-

sen, in welchen Unterquadraten sich Empfänger der Gruppe fünf befinden und das Paket entsprechend weiterleiten.

In einer umfangreichen Simulationsstudie mit Hilfe des Netzwerksimulators ns-2 konnte gezeigt werden, dass SPBM wesentlich bessere Skalierungseigenschaften mit sich bringt als das bis dato als Referenzprotokoll geltende *On-Demand Multicast Routing Protocol* (ODMRP) [8]. Abbildung 2(a) zeigt ein Simulationsergebnis für variierende Anzahlen an Sendern in einer Multicast-Gruppe. Schon ab zwei gleichzeitigen Sendern sinkt die Zustellrate von ODMRP deutlich ab, während SPBM unter gleichen Bedingungen auch bei zehn Sendern nahezu alle Pakete erfolgreich ausliefern kann. In Abbildung 2(b) sind die Summen der insgesamt übertragenen Bytes auf der Sicherungsschicht der gleichen Simulation dargestellt. ODMRP generiert ab zwei Sendern eine ca. viermal höhere Netzwerklast als SPBM. Hierdurch wird das Netzwerk überlastet, und es kommt zu den in 2(a) gezeigten Paketverlusten. Weitere Ergebnisse und Details der Funktionsweise sind in [24], [25], [23], [22] und [19] nachzulesen.



(a) Zustellrate

(b) Summe der auf der Sicherungsschicht übertragenen Bytes

Abb. 2. Variable Anzahl von Sendern (10 Empfänger, 1 Pkt/s, 10 m/s, 100 Knoten)

Des Weiteren wurde mit SPBM-BC eine Variante von SPBM entwickelt, welche die Pakete nicht per 1-Hop-Unicast explizit an die nächsten Weiterleiter versendet, sondern die Pakete per 1-Hop-Broadcast an alle Nachbarn schickt. Die Empfänger können dann dem Paketkopf entnehmen, ob sie als Weiterleiter ausgewählt wurden. Die Knoten, die ausgewählt wurden, leiten das Paket nach dem bekannten Prinzip weiter, während die anderen Knoten das Paket ignorieren. Durch das erneute Versenden empfängt der ursprüngliche Sender ebenfalls das Paket und kann dies als implizite Bestätigung verwenden, dass der Weiterleiter das Paket ordnungsgemäß

empfangen hat. Sollte diese Bestätigung ausbleiben, überträgt der ursprüngliche Sender das Paket erneut per 1-Hop-Broadcast und löscht zuvor alle Weiterleiter, von denen er eine Bestätigung erhalten hat, aus dem Paketkopf. Durch dieses Verfahren konnte die Anzahl der Einzelübertragungen weiter gesenkt werden.

Die im Düsseldorfer Teilprojekt entwickelte Überlastkontrolle CXCC [15] wurde in Kooperation mit uns für den Multicast angepasst und in die 1-Hop-Broadcast-Variante von SPBM integriert. Die Idee der neu entstandenen *Backpressure Multicast Congestion Control* (BMCC) ist, dass ein Paket eines Datenstroms erst dann zum nächsten Hop gesendet werden darf, wenn das vorherige Paket von allen ausgewählten Knoten um einen weiteren Hop weitergeleitet wurde. Durch dieses Verfahren wird eine Überlastung Schritt für Schritt in Richtung des Senders des Datenstroms gemeldet, der dann die Datenrate reduzieren kann. Auf diese Weise passt sich die Bandbreite des Datenstroms an den Empfänger mit der schlechtesten Anbindung an. In manchen Fällen, je nachdem welche Daten in welcher Form übertragen werden, kann es sinnvoller sein die Paketverluste für einzelne Knoten in Kauf zu nehmen. Hierfür wurden Verfahren entwickelt, welche die Bandbreite für jeden Zweig des Multicast-Baums getrennt anpassen können. Weitere Details und ausführliche Simulationsergebnisse können der gemeinsamen Publikation [16] entnommen werden.

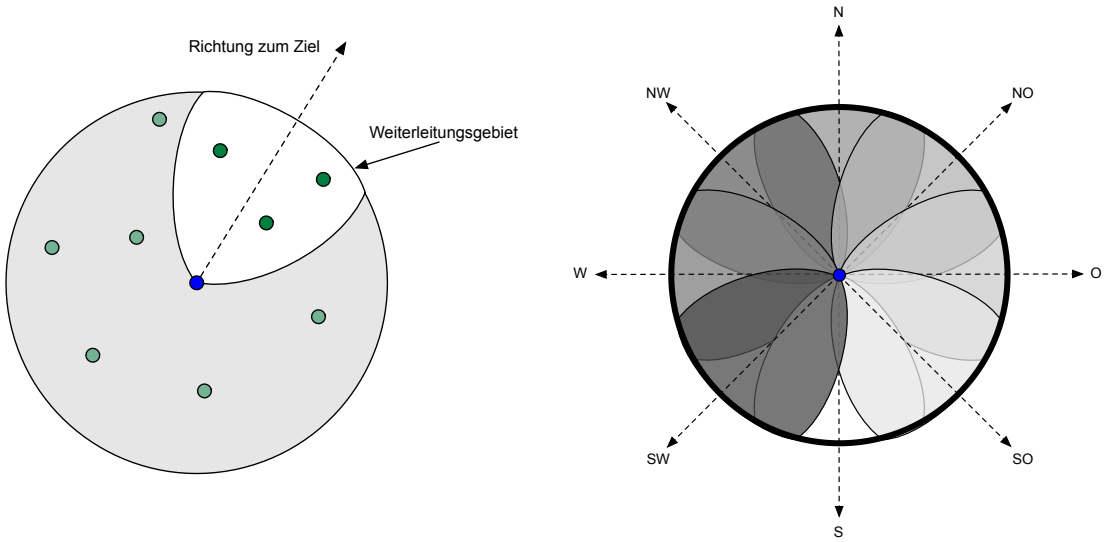
4. Contention-Based Multicast Forwarding

PBM und SPBM leiten beide die Pakete vom Sender zu den Empfängern in der Multicast-Gruppe per Unicast weiter. Dementsprechend wird für jedes verbleibende Ziel der am besten geeignete Nachbar als Weiterleiter explizit ausgewählt. Zu jedem der ausgewählten Nachbarn wird ein separates Paket per Unicast übertragen, welches im Paketkopf die dem jeweiligen Knoten zugeordneten Zieladressen (bzw. Zielquadrate) enthält. Auf der einen Seite können hierbei die automatischen Übertragungswiederholungen und Bestätigungen auf der Sicherungsschicht ausgenutzt werden, andererseits entsteht aber eine redundante Belegung des Mediums. Da alle Netzteilnehmer auf ein gemeinsames Medium zugreifen, können Unicast-Pakete von allen Knoten in Radioreichweite gehört werden. Entsprechend werden hierdurch auch die nicht adressierten Nachbarn am Senden oder Empfangen anderer Nachrichten gehindert. Durch diesen Umstand entsteht sehr häufig die Situation, dass ein Nachbar, der als Weiterleiter ausgewählt wurde, mehrere nahezu identische Pakete empfängt - einmal für sich selbst und einige Male für andere Weiterleiter in seiner Radioreichweite. Aus dieser Beobachtung entstand die Idee, solche Pakete nur einmal per 1-Hop Broadcast zu versenden und das Routing-Protokoll so anzupassen, dass die Weiterleiter durch einen verteilten Algorithmus implizit ermittelt

werden. Die Auswahl der Weiterleiter erfolgt dann also nicht mehr beim Sender, sondern bei den Empfängern.

Für den positionsbasierten Unicast existiert mit dem Mannheimer *Contention-Based Forwarding* (CBF) [4] ein Routing-Protokoll, welches die Auswahl der Weiterleiter den Empfängern überlässt und damit hervorragende Ergebnisse erzielt. CBF überträgt die Pakete per Broadcast an alle Nachbarn in Radioreichweite und bestimmt über einen verteilten Wettbewerb den am besten geeigneten Weiterleiter. Hierfür zieht jeder Empfänger eine Uhr auf, die umso schneller abläuft, je näher sich der Empfänger zum Zielknoten befindet. Beim Knoten mit dem kleinsten Abstand läuft die Uhr als Erstes ab, und der Knoten leitet das Paket per Broadcast weiter. Diese Weiterleitung führt dann zum Anhalten der Uhren auf den anderen Knoten des Wettbewerbs. Bei der expliziten Auswahl auf dem Senderknoten, werden die Positionen der Nachbarn benötigt, die mit Hilfe von regelmäßig gesendeten Beacon-Nachrichten ausgetauscht werden müssen. Die Genauigkeit der Positionsdaten, die bei der Auswahl verwendet werden können, hängt entsprechend stark von der Sendefrequenz dieser Beacon-Nachrichten ab. Im Gegensatz hierzu wird bei CBF nur die Position des Zieles, welche im Paketkopf gespeichert ist, und die eigene Position benötigt. Deshalb kann auf das Senden von Beacon-Nachrichten verzichtet werden. Hierdurch sind die verwendeten Positionen immer aktuell.

Diese Vorteile führten zu der Idee, diese Auswahlstrategie auch im positionsbasierten Multicast einzusetzen. Es entstand *Contention-Based Multicast Forwarding* (CBMF) [21]. Der wichtigste Unterschied liegt darin, dass beim Unicast genau ein geographischer Punkt adressiert wird, wohingegen beim Multicast mehrere Ziele existieren können. Als Grundlage für die neue Konzeption der Weiterleitungszonen dient das gebietsbasierte Schema, wie es in CBF definiert wurde: Während einer Wettbewerbsphase nehmen alle diejenigen Knoten an dem Wettbewerb teil, die sich innerhalb eines Reuleaux-Dreiecks befinden, dessen eine Ecke auf dem weiterleitenden Knoten und dessen Spiegelachse auf der direkten Verbindungslinie zwischen dem weiterleitenden Knoten und der endgültigen Zielposition liegt (siehe Abbildung 3(a)). Für den Multicast-Fall wurden nun acht dieser Reuleaux-Dreiecke fest definiert, die in die acht verschiedenen Himmelsrichtungen N, NW, W, SW, S, SO, O, NO ausgerichtet sind (siehe Abbildung 3(b)). Der weiterleitende Knoten legt nun fest, welches der Ziele über welchen dieser Sektoren erreicht werden soll, speichert diese Daten im Kopf des Pakets und sendet das Paket per Broadcast an seine Nachbarn. Nun wird in den angegebenen Sektoren nacheinander ein Wettbewerb ausgeführt, indem der für den jeweiligen Sektor am besten geeignete Knoten bestimmt wird. Dieser übernimmt die Weiterleitung des Pakets und beendet damit den Wettbewerb. Durch diese Übertragung wird ebenfalls dem Vorgänger implizit der korrekte Empfang des Paketes bestätigt.



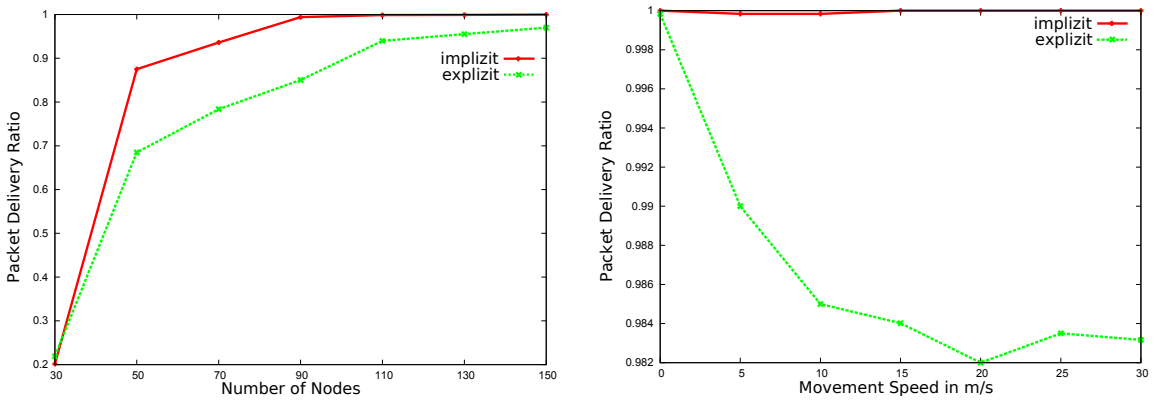
(a) CBF Unicast-Routing (b) CBMF Multicast-Routing

Abb. 3. Das Reuleaux-Dreieck als Wettbewerbsregion

Dieses Verfahren bringt Steigerungen in der Zustellrate vor allem für Szenarien, in denen eine niedrige Knotendichte herrscht oder in denen sich die Knoten schnell bewegen. Abbildung 4(a) zeigt die Paketzustellrate in Abhängigkeit von der Knotendichte im simulierten Szenario, bei einer moderaten Maximalgeschwindigkeit von 10 m/s. Vor allem in dünn besiedelten Szenarien ist der Zuwachs bei der Paketzustellrate enorm. Das rührt daher, dass in den Szenarien mit niedriger Knotendichte die durchschnittliche Entfernung zwischen den Knoten größer ist und damit auch die Wahrscheinlichkeit steigt, dass ein Knoten sich seit dem letzten empfangenen Beacon-Paket aus der Radioreichweite entfernt hat. Das implizite Verfahren CBMF nutzt stets die aktuellen Positionen und ist somit nicht auf die möglicherweise veralteten Nachbarschaftstabellen angewiesen. Auch im Fall von größerer Mobilität der Knoten zeigt Abbildung 4(b), dass CBMF höhere Zustellraten erreicht als das explizite Verfahren. Eine Veröffentlichung von detaillierten Ergebnissen aus dem Vergleich der verschiedenen Weiterleitungsmethoden ist in [21] zu finden.

5. Implementierung

Simulationen können die Funktionsfähigkeit und Korrektheit eines Protokolls zeigen, es ist jedoch unerlässlich, Protokolle für mobile Ad-hoc-Netzwerke auch in der Realität zu testen. Aus diesem Grund wurde eine Implementierung von SPBM und CBMF als Linux-Kernel-Module vorgenommen. Die Module lassen sich sowohl auf PCs mit der i386-Architektur als auch auf HP iPaq-Handheld-Geräten kompilie-



(a) Variierende Knotendichten (b) Variierende Mobilität

Abb. 4. Zustellraten der expliziten und impliziten Weiterleitung im Vergleich

ren und verwenden. Empfangene Pakete werden nur an Programme im Userspace ausgeliefert, wenn es sich um eine an der entsprechenden Multicast-Gruppe angemeldete Applikation handelt. Weitere Details zu der Implementierung können [20] entnommen werden.

Aufgrund ihrer handlichen Bauweise wurden die Linux-kompatiblen Handhelds von HP als Evaluierungsplattform ausgewählt. Mit Hilfe einer neu erstellten Anwendung werden die aktuellen Positionsdaten von einem angeschlossenen GPS-Empfänger ausgelesen und dem Kernel-Modul für das Routing zur Verfügung gestellt. Alternativ kann die Position auch manuell gesetzt werden. Des Weiteren wurde eine Software mit graphischer Benutzeroberfläche für die iPaqs entwickelt, die es erlaubt das SPBM- und CBMF-Kernel-Modul auf einfache Art und Weise zu konfigurieren und zu überwachen. Die Software erspart die Konfiguration über die Kommandozeile und bietet Funktionen wie das Ändern von Gruppenmitgliedschaften und das Anzeigen von Informationen über aktuelle Nachbarn.

Das Prinzip von CBMF legt es nahe, die Routing-Schicht mit der Medienzugriffskontrolle, die eine Schicht tiefer angesiedelt ist, zu verbinden, um eine exaktere Zeitsteuerung zu erreichen und Zugriff auf die Sendewarteschlagen zu erhalten. Um dies auf einem realen System zu ermöglichen, benötigt der Programmierer Zugriff auf die Implementierung der Medienzugriffskontrolle, welche aber bei herkömmlichen WLAN-Karten meist Bestandteil der Firmware ist. Diese liegt allerdings nicht als Quelltext vor, und es ist im Normalfall untersagt, diese zu verändern bzw. zu manipulieren. Ohne Zugriff auf die Timersteuerung und Sendewarteschlagen müssen aber für eine Implementierung auf PC-Hardware viel größere Timerwerte gewählt werden als theoretisch möglich, was die Performance beeinträchtigt und die Messungen verfälscht. Deshalb wurde CBMF außerdem auch noch auf den

ESB-Sensorknoten der FU Berlin implementiert, die einen selbst gesteuerten Medienzugriff erlauben und somit eine ideal geeignete Testplattform darstellen. Weitere Details und Messergebnisse können [7] entnommen werden.

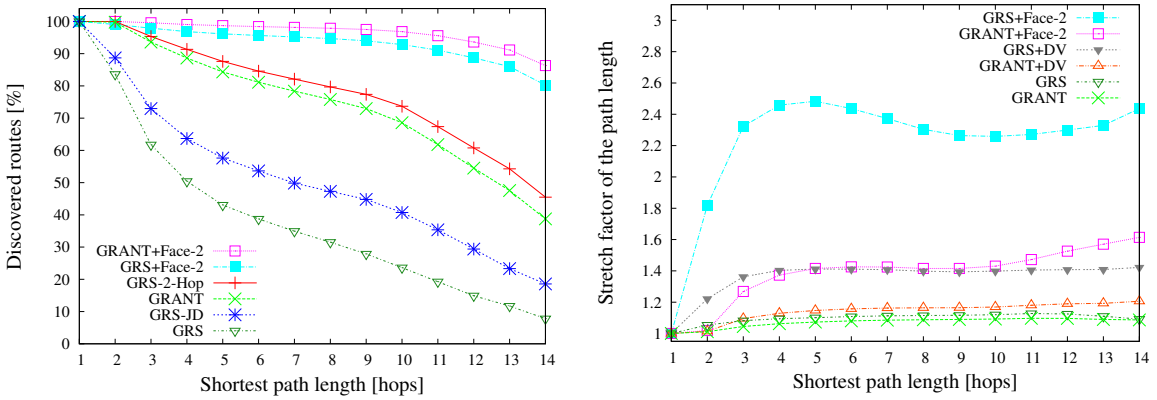
6. Greedy Routing with Abstract Neighbor Table

Alle positionsbasierten Routing-Protokolle, ob Unicast oder Multicast, versuchen mit Hilfe von Positionen der Knoten einen Pfad vom Sender zum Ziel zu finden. Hierbei wird davon ausgegangen, dass jedem Knoten die eigene Position und die der direkten Nachbarn in Funkreichweite bekannt sind. Nun wird anhand einer gegebenen Zielposition der Nachbarknoten mit dem kleinsten räumlichen Abstand zum Ziel als nächster Weiterleiter ausgewählt. Die Idee hierbei ist, dass sich bei jeden Weiterleiter die Distanz zum Ziel verringert und schließlich das Ziel erreicht wird.

Allerdings wird durch Radiohindernisse der Zusammenhang zwischen räumlicher Nähe und Kommunikationsmöglichkeit stark reduziert. Um diesen Umstand genauer zu untersuchen und nicht auf synthetische Daten zurückgreifen zu müssen, wurde der Fokus im weiteren Verlauf des SPP-Projekts auf eine möglichst realistische Modellierung eines Stadtgebietes gelegt. Das Stadtgebiet ist charakterisiert durch sehr viele Radiohindernisse und durch eine eingeschränkte Bewegungsmöglichkeit der Netzteilnehmer. Durch diesen Sachverhalt ist es für positionsbasierte Verfahren eine große Herausforderung, in dieser Umgebung einen Pfad zu finden. Anstatt eine synthetische Stadtkarte und die entsprechenden Radiohindernisse zu generieren, entschieden wir uns, einen real existierenden Innenstadtbereich nachzubilden. Hierfür suchten wir nach kostenlosen und frei verfügbaren Straßenkarten und wurden bei OpenStreetMap [13] fündig. Allerdings stimmten die Straßenverläufe der exportierten Karte nicht exakt mit den Satellitenbildern von Google Earth überein, welche wir für die Erfassung der Radiohindernisse verwenden wollten. Deshalb musste die exportierte Karte mit einer selbst entwickelten Anwendung nachjustiert werden. Nach dieser Anpassung wurden die Gebäudeumrisse in Google Earth nachgezeichnet, exportiert und mit einem Konverter in das Format des Netzwerksimulators umgewandelt. Hierdurch ist ein Simulationsszenario der gesamten Karlsruher Innenstadt mit einer Straßenlänge von 66 km und realen Gebäudeumrissen entstanden. Die Bewegungsdaten können nun beispielsweise mit Hilfe des quelltextoffenen Verkehrssimulators SUMO [1] generiert werden.

Wir verwendeten das erzeugte Szenario, um eine umfangreiche Simulationsstudie zur Analyse der Performance von positionsbasierten Routing-Protokollen im Innenstadtbereich durchzuführen. Der Schwerpunkt lag hierbei auf der Auswahl des Weiterleiters, die bei positionsbasierten Unicast- und Multicast-Protokollen nahezu identisch ist. Um die Anzahl der Einflussfaktoren (Gruppenzugehörigkeit,

Gridgröße usw.) zu reduzieren, simulierten wir deshalb nur die Unicast-Protokolle. Zwar wurden, wie in den Szenarien ohne Radiohindernisse, kurze Verzögerungszeiten bei einer geringen Bandbreitenauslastung erreicht, aber die Anzahl der Knoten, die keine Route zum Kommunikationspartner finden konnten, war inakzeptabel hoch. So konnten mit der Basisstrategie GRS nur zu ca. 44% aller theoretisch erreichbaren Knoten mit einem Abstand von fünf Hops eine Route gefunden werden (siehe Abbildung 5(a)). Um eine Route zu den restlichen 56% zu finden, muss GRS um eine Recovery-Strategien ergänzt werden. Im Fall des bekannten GPSR [6] ist dies der FACE-2-Algorithmus, der einen planaren Graphen voraussetzt. Zwar existieren mit dem Gabriel-Graph und dem Relative-Neighborhood-Graph zwei bekannte Verfahren, einen verbundenen planaren Graphen verteilt mit lokalem Wissen zu berechnen, aber beide Verfahren setzten voraus, dass sich zwei Knoten in Radioreichweite auch hören können. Die im Stadtgebiet zahlreich vorhandenen Radiohindernisse führen deshalb zu einer fehlerhaften Planarisierung und somit auch zu einer sehr schlechten Performance des FACE-2-Algorithmus. Zwar wurden mit GPSR im oben gewählten Beispiel 96% der Routen gefunden, aber die Pfadlänge hat sich durch die fehlerhafte Planarisierung auf das Vielfache von GRS erhöht (siehe Abbildung 5(b)).



(a) Das Verhältnis zwischen der Anzahl der gefundenen Routen und den theoretisch existierenden

(b) Streckungsfaktor der durchschnittlichen Pfadlänge

Abb. 5. Simulationsergebnisse

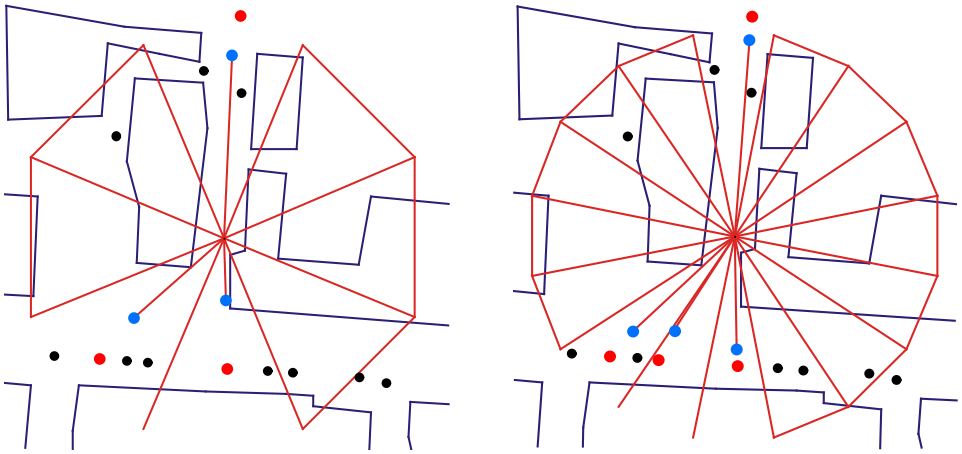
Wir haben deshalb eine auf Distanzvektoren basierende und an *Ad-Hoc On-Demand Distance Vector Routing* (AODV) [14] angelehnte Recovery-Strategie vorgeschlagen, die nahezu alle Routen findet und dabei die Pfadlänge nur geringfügig erhöht. Allerdings verwendet diese Strategie, genauso wie AODV, einen gesteuerten Flutmechanismus bei der Routenfindung. Es ist deshalb notwendig, den Basisalgo-

rithmus soweit wie möglich zu verbessern, um die *teuren* Recovery-Strategien so selten wie möglich einsetzen zu müssen. Durch unsere vorgeschlagene Kreuzungserkennung ohne Kartenkenntnis [9] konnte zu 58% aller theoretisch erreichbaren Knoten mit einem Abstand von fünf Hops eine Route gefunden werden. In Abbildung 5 wird diese erweiterte Basisstrategie als GRS-JD bezeichnet.

Da leider auch dieses Ergebnis hinter unseren Erwartungen zurück blieb, suchten wir nach weiteren Alternativen. Durch die regelmäßig gesendeten Beacons wird auf jedem Knoten eine Tabelle aller direkter Nachbarn und ihrer Positionen aufgebaut. Es liegt nahe, davon auszugehen, dass mehr Informationen zu einem korrekteren und vollständigeren Bild der Umgebung führen und somit auch das Routing verbessern können. Wir modifizierten unser Protokoll dann so, dass in den Beacon-Paketen zusätzlich alle direkten Nachbarn des Senders aufgelistet werden. Auf diese Weise kann eine Zwei-Hop-Nachbarschaftstabelle aufgebaut werden. Die Quote der ohne Recovery-Strategie gefunden Routen stieg auf sehr gute 88%. In der Abbildung 5 wird diese Variante als GRS-2-Hop bezeichnet.

Die Anzahl der Knoten in Radioreichweite kann in dichten Gebieten des Netzwerkes leicht auf über hundert steigen. Würde als Netzwerk-ID eine IPv4 Adresse verwendet werden und die Position wie üblich als Längen- und Breitengrade mit je 32 Bit kodiert werden, müssten in diesem Fall 1200 Byte zusätzlich in jedem Beacon-Paket übertragen werden. Es handelt sich hierbei also um einen eher theoretischen Ansatz. Wir nutzten allerdings diese Erkenntnisse, um die sogenannte *Abstract Neighbor Table* (ANT) zu entwickeln. Hierbei wird die Ebene um den Knoten in gleich große Gebiete aufgeteilt und pro Gebiet ein Repräsentant ausgewählt und dessen Position geschickt kodiert. In Abbildung 6 sind zwei Varianten einer ANT exemplarisch visualisiert. Ausgehend vom Knoten in der Mitte wird die Ebene in acht bzw. sechzehn gleich große Gebiete aufgeteilt. Aus der Menge von Nachbarn in einem Gebiet wird ein Repräsentant dieses Gebietes ausgewählt. Im Beispiel wurde jeweils der Nachbar als Repräsentant, rot gekennzeichnet, ausgewählt, der den größten räumlichen Abstand zu dem aktuell betrachteten Knoten besitzt. Die Positionen der Repräsentanten werden verlustbehaftet in einer kompakten Form in der ANT gespeichert, welche im eigenen Beacon-Paket zusätzlich zu der Netzwerk-ID und der eigenen Position übertragen wird. Auf diese Weise wird nicht wie bei GRS-2-Hop die komplette Nachbarschaftstabelle übertragen, sondern nur eine grobe Übersicht in Form einer ANT. Der Empfänger des Beacon-Pakets kennt nun die Position des Senders und die grobe Position der Repräsentanten, welche in der Abbildung blau eingezeichnet sind. Die Position der Repräsentanten wird mit Hilfe eines Vektors, welcher von dem Sender ausgeht, kodiert. Sowohl die Länge als auch die Richtung des Vektors werden durch quantisierte Werte repräsentiert. In der linken Abbildung wurden sechzehn Intervalle zum Kodieren der Länge des Vektors verwendet und in der rechten zweiunddreißig. Dementsprechend sind die in 6(b)

in blau eingezeichneten kodierten Position der Repräsentanten exakter. In beiden Grafiken wurden zum Kodieren des Winkels acht Intervalle verwendet. Pro Gebiet einer ANT müssen je ein Winkel und eine Länge gespeichert werden. Somit benötigt die linke ANT $\frac{8}{8}(\log_2 8+\log_2 16) = 7$ Bytes und die rechte $\frac{16}{8}(\log_2 8+\log_2 32) = 16$ Bytes an Speicherplatz. In der Simulationsstudie hat sich die Unterteilung in acht Gebiete als guter Kompromiss zwischen Größe und Genauigkeit erwiesen. Eine ANT mit acht Gebieten und der beschriebenen Kodierung hat die konstante Größe von sieben Byte und kann ohne übermäßige Belastung den Beacon-Paketen hinzugefügt werden.



(a) ANT mit 8 Gebieten und 16 Intervallen zur Kodierung der Länge

(b) ANT mit 16 Gebieten und 32 Intervallen zur Kodierung der Länge

Abb. 6. Varianten der *Abstract Neighbor Table*

Alle GRS-Varianten berechnen für jeden möglichen Weiterleiter einen Wert, der seine Eignung repräsentiert. Im klassischen GRS ist dies der Abstand zwischen der Position des Knotens und der Position des Ziels. Nachdem für jeden möglichen Weiterleiter ein Wert berechnet wurde, wird der Nachbar mit dem kleinsten Wert als nächster Hop verwendet. Ist in den Beacon-Paketen eine ANT enthalten, kann mit Hilfe einer neuen Metrik für jeden potentiellen Weiterleiter ein Eignungswert berechnet werden, in dem die zusätzlichen Informationen berücksichtigt werden. Das hieraus entstandene Protokoll heißt *Greedy Routing with Abstract Neighbor Table* (GRANT). Wie sich auf dem Knoten V der Eignungswert für den potentiellen Weiterleiter E ausrechnen lässt, ist in Algorithmus 1 dargestellt. Ist der Knoten E mit dem Zielknoten D identisch, wird dieser priorisiert. Ansonsten wird die Variable $metricVal$ mit dem Abstand zwischen E und D initialisiert. Dann wird für jedes Gebiet $h \in [0, a-1]$ der gespeicherte Vektor dekodiert. Dieser Vektor kann dann

verwendet werden, um die Position des Repräsentanten zu bestimmen. Sollte der Abstand zwischen dem Repräsentant und D kleiner sein als $metricVal$, so wird der Abstand in $metricVal$ gespeichert. Auf diese Weise wird der kleinste räumliche Abstand zwischen E bzw. den Repräsentanten und D als Eignungswert für E verwendet. Um Fehler beim Routing zu vermeiden, muss die Länge des Vektors, der zur Berechnung der Position des Repräsentanten verwendet wird, auf den Abstand $\overline{ED}$ beschränkt werden. Sollte sich ein Repräsentant selbst in Radioreichweite zu V befinden und somit selbst ein potentieller Weiterleiter sein, so muss sichergestellt werden, dass für diesen ein kleinerer Eignungswert berechnet wird als für E. Hierfür wird die Länge des Vektors zusätzlich skontiert, wodurch sich der Fortschritt in Richtung D reduziert und der direkt benachbarte Repräsentant bevorzugt wird.

Algorithmus 1 : Berechnung des Eignungswerts eines potentiellen Weiterleiters

```
if E = D then
    return −∞
metricVal ← d(E, D)
maxLen ← metricVal
for h ∈ [0, a − 1] do
    if ANT[h].encLen > 0 then
        len ← decEstLen(ANT[h].encLen)
        ang ← decEstAngle(h, ANT[h].encAngle)
        if maxLen < len then
            len ← maxLen
        len ← len ∗ DISCOUNT
        NH_h ← getEstNGPos(E, ang, len)
        if d(NH_h, D) < metricVal then
            metricVal ← d(NH_h, D)
return metricVal
```

GRANT findet eine Route zu ca. 86% aller theoretisch erreichbaren Knoten mit einem Abstand von fünf Hops und ist damit nur geringfügig schlechter als die Variante mit der vollständigen Kenntnis der Zwei-Hop-Nachbarschaft, bei erheblicher Einsparung der zu übertragenen Daten. Durch diese Verbesserung werden die *teuren* Recovery-Strategien sehr viel seltener benötigt. Hierdurch wurde der durchschnittliche Streckungsfaktor des kürzesten Weges bei der FACE-2-Recovery von $2,4$ auf $1,4$ und bei der Recovery mit Distanzvektoren von $1,4$ auf $1,15$ reduziert. GRANT und die Ergebnisse der Simulationsstudie wurden in [17] veröffentlicht. Das entwickelte Verfahren und die zugehörige Metrik kann nun direkt in SPBM und anderen positionsbasierten Routing-Protokoll mit Beacon-Paketen eingesetzt werden, um die Zuverlässigkeit in Gebieten mit Radiohindernissen zu erhöhen.

7. Zusammenfassung

Im Projektverlauf sind mit SPBM und CBMF neue, robuste und skalierbare Multicast-Protokolle für selbstorganisierende vernetzte mobile Systeme entstanden, die per Simulation gründlich evaluiert wurden. Beide Protokolle verwenden bei der Paketvermittlung die Positionsinformationen ihrer Nachbarn. Während SPBM die Weiterleiter explizit auf dem aktuellen Knoten mit Hilfe einer Nachbarschaftstabelle ermittelt, verwendet CBMF einen verteilten Wettbewerb, um die geeigneten Weiterleiter auszuwählen. Bei diesem Wettbewerb werden die Positionen der direkten Nachbarn nicht auf dem Gerät des Senders benötigt, weshalb hier auf den regelmäßigen Austausch von Positionsdaten verzichtet werden kann. Allerdings kann in den Beacon-Paketen von SPBM zusätzlich eine kompakte Form der Nachbarschaftstabelle (ANT) übertragen werden, die es erlaubt mit Hilfe der GRANT-Weiterleiterauswahl auch in Gebieten mit vielen Radiohindernissen effizient zu kommunizieren. Beide Protokolle wurden als Linux-Kernel-Module implementiert und können somit auch im realen Einsatz verwendet werden. Des Weiteren wurde die Broadcast-Variante von SPBM in Kooperation mit der Universität Düsseldorf um eine Überlastkontrolle ergänzt. Sowohl die detaillierten Simulationsstudien als auch die ersten Experimente mit mobilen Geräten haben gezeigt, dass die beschriebenen Verfahren eine effiziente Gruppenkommunikation in mobilen Ad-Hoc-Netzen ermöglichen und sich hervorragend als universell einsetzbare Protokolle für zukünftige Anwendungen eignen.

Literaturverzeichnis

[1] *Simulation of Urban Mobility*. http://sumo.sourceforge.net/, 20.10.2008.

[2] BOSE, PROSENJIT, PAT MORIN, IVAN STOJMENOVIC und JORGE URRUTIA: *Routing with guaranteed delivery in ad hoc Wireless Networks*. In: *Proceedings of the 3rd international Workshop on Discrete Algorithms and Methods for Mobile Computing and communications (DIAL-M 1999)*, Seiten 48–55, Seattle, WS, USA, August 1999.

[3] FINN, GREGORY G.: *Routing and addressing problems in large metropolitan-scale internetworks*. Technischer Bericht ISI/RR-87-180, ISI, March 1987.

[4] FÜSSLER, HOLGER, JÖRG WIDMER, MICHAEL KÄSEMANN, MARTIN MAUVE und HANNES HARTENSTEIN: *Contention-Based Forwarding for Mobile Ad-Hoc Networks*. Elsevier's Ad Hoc Networks, 1(4):351–369, 2003.

[5] GABRIEL, K. und R. SOKAL: *A new statistical approach to geographic variation analysis*. Systematic Zoology, 18:259–278, 1969.

[6] KARP, BRAD N. und H. T. KUNG: *GPSR: Greedy Perimeter Stateless Routing for Wireless Networks*. In: *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000)*, Seiten 243–254, Boston, MA, USA, August 2000.

[7] KÄSTLE, JAN: *Contention-Based Forwarding for Uni- and Multicast in Sensor Networks*. Diplomarbeit, Department of Mathematics and Computer Science IV, University of Mannheim, September 2005.

[8] LEE, SUNG-JU, MARIO GERLA und CHING-CHUAN CHIANG: *On-Demand Multicast Routing Protocol*. In: *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC 1999)*, Seiten 1298–1302, New Orleans, LA, USA, September 1999.

[9] LOCHERT, CHRISTIAN, HOLGER FÜSSLER, MARTIN MAUVE und HANNES HARTENSTEIN: *Geographic Routing in City Scenarios*. ACM SIGMOBILE Mobile Computing and Communications Review (MC2R), 9(1):69–72, January 2005.

[10] MAUVE, MARTIN, HOLGER FÜSSLER, JÖRG WIDMER und THOMAS LANG: *Position-Based Multicast Routing for Mobile Ad-Hoc Networks*. Technischer Bericht TR-03-004, Department of Computer Science, University of Mannheim, 2003.

[11] MAUVE, MARTIN, HOLGER FÜSSLER, JÖRG WIDMER und THOMAS LANG: *Position-based multicast routing for mobile Ad-hoc networks*. ACM SIGMOBILE Mobile Computing and Communications Review (MC2R), 7(3):53–55, July 2003.

[12] MAUVE, MARTIN, HOLGER FÜSSLER, JÖRG WIDMER und THOMAS LANG: *Poster: Position-Based Multicast Routing for Mobile Ad-Hoc Networks*. In: *Proceedings of the Fourth ACM international Symposium on Mobile and Ad Hoc Networking and Computing (MobiHoc 2003, electronic edition)*, Annapolis, MD, USA, June 2003.

[13] OPENSTREETMAP. http://www.openstreetmap.org/, 20.10.2008.

[14] PERKINS, CHARLES E. und ELIZABETH M. ROYER: *Ad-Hoc On-Demand Distance Vector Routing*. In: *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 1999)*, Seiten 90–100, New Orleans, LA, USA, February 1999.

[15] SCHEUERMANN, BJÖRN, CHRISTIAN LOCHERT und MARTIN MAUVE: *Implicit Hop-by-Hop Congestion Control in Wireless Multihop Networks*. Elsevier's Ad Hoc Networks, 2007.

[16] SCHEUERMANN, BJÖRN, MATTHIAS TRANSIER, CHRISTIAN LOCHERT, MARTIN MAUVE und WOLFGANG EFFELSBERG: *Backpressure Multicast Congestion Control in Mobile Ad-hoc Networks*. In: *Proceedings of the 3rd ACM International Conference on Emerging Networking Experiments and Technologies (CoNEXT 2007)*, New York City, NY, USA, December 2007.

[17] SCHNAUFER, SASCHA und WOLFGANG EFFELSBERG: *Position-Based Unicast Routing for City Scenarios*. In: *Proc. of the 9th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2008)*, Newport Beach, CA, USA, 06 2008.

[18] TOUSSAINT, GODFRIED: *The relative neighborhood graph of a finite planar set*. Pattern Recognition, 12(4):261–268, 1980.

[19] TRANSIER, MATTHIAS: *Position-Based Multicast for Mobile Ad-hoc Networks*. Doktorarbeit, Universität Mannheim, 2008.

[20] TRANSIER, MATTHIAS, HOLGER FÜSSLER, THOMAS BUTTER und WOLFGANG EFFELSBERG: *Implementing Scalable Position-Based Multicast for the Linux Kernel*. In: *Proceedings of the 2nd German Workshop on Mobile Ad-Hoc Networking (WMAN 2004)*, Seiten 105–110, Ulm, Germany, September 2004.

[21] TRANSIER, MATTHIAS, HOLGER FÜSSLER, THOMAS BUTTER und WOLFGANG EFFELSBERG: *Poster: Contention-Based Multicast Forwarding*. In: *Proceedings of the 7th ACM inter-*

national Symposium on Mobile and Ad Hoc Networking & Computing, Poster Session (MobiHoc 2006), Florence, Italy, May 2006.

[22] TRANSIER, MATTHIAS, HOLGER FÜSSLER, MARTIN MAUVE, JÖRG WIDMER und WOLFGANG EFFELSBERG: *Poster: Dynamic Load Balancing for Position-Based Routing*. In: *Proceedings of the ACM International Conference on Emerging Networking Experiments and Technologies, Poster Session (CoNEXT 2005)*, Seiten 290–291, Toulouse, France, October 2005.

[23] TRANSIER, MATTHIAS, HOLGER FÜSSLER, JÖRG WIDMER, MARTIN MAUVE und WOLFGANG EFFELSBERG: *A Hierarchical Approach to Position-Based Multicast for Mobile Ad-hoc Networks*. Technischer Bericht TR-04-002, Department for Mathematics and Computer Science, University of Mannheim, 2004.

[24] TRANSIER, MATTHIAS, HOLGER FÜSSLER, JÖRG WIDMER, MARTIN MAUVE und WOLFGANG EFFELSBERG: *Scalable Position-Based Multicast for Mobile Ad-hoc Networks*. In: *Proceedings of the First International Workshop on Broadband Wireless Multimedia: Algorithms, Architectures and Applications (BroadWIM 2004)*, San José, CA, USA, October 2004.

[25] TRANSIER, MATTHIAS, HOLGER FÜSSLER, JÖRG WIDMER, MARTIN MAUVE und WOLFGANG EFFELSBERG: *A Hierarchical Approach to Position-Based Multicast for Mobile Ad-hoc Networks*. Wireless Networks – The Journal of Mobile Communication, Computation and Information, 13(4):447–460, August 2007.

Selbstkonfiguration adaptiver Anwendungen in ubiquitären Systemen

Stephan Schuhmann, Klaus Herrmann, Kurt Rothermel

Institut für Parallele und Verteilte Systeme
Universität Stuttgart

Zusammenfassung – Das Forschungsgebiet des Pervasive Computing hat in den letzten Jahren zunehmend an Bedeutung gewonnen. Hauptziel ist die Integration von Computertechnologie in Alltagsgegenstände und die Nutzung dieser hierdurch elektronisch angereicherten Benutzerumgebung, ihrer Geräte und Dienste für die Ausführung von Anwendungen. Dafür müssen diese Anwendungen in die Lage versetzt werden, sich dynamisch an wechselnde Umgebungen anzupassen, beispielsweise durch Verlagerung ihrer Funktionalität zwischen Geräten. Bedingt durch dynamische Umgebungen, Nutzermobilität sowie drahtlose Kommunikationstechnologien ist die Entwicklung von Anwendungen für Pervasive Computing Umgebungen hochkomplex. Daher wurden in diesem Projekt grundlegende Konzepte und Algorithmen entwickelt, um eine automatisierte Nutzerunterstützung in diesen Umgebungen zu ermöglichen. Hierbei lag der Schwerpunkt auf der Entwicklung von Algorithmen zur Selbstkonfiguration von Anwendungen mittels automatisierter Komposition und Adaption. Neben dynamischen, homogenen Ad Hoc Umgebungen sollten außerdem heterogene Umgebungen, in denen zusätzlich ressourcenstarke Infrastrukturgeräte vorhanden sind, berücksichtigt werden, um eine effiziente Ausführung von Konfigurationen und Adaptionen auch in solchen Umgebungen zu ermöglichen.

1. Einleitung

Der Forschungsbereich des Pervasive Computing wird charakterisiert durch die Interaktion vieler heterogener Geräte, welche von leistungsstarken Serverinfrastrukturen bis zu winzigen, in Alltagsgegenstände integrierten mobilen Sensoren reichen. Hierbei können Geräte durch standardisierte drahtlose Kommunikationstechnologien wie Bluetooth, Wireless LAN oder Infrarot miteinander kommunizieren. Die in einer bestimmten Umgebung verfügbaren Geräte stellen dabei ihre Funktionalität anderen in der Nähe befindlichen Geräten zur Verfügung.

Die Entwicklung von Anwendungen für dynamische Pervasive Computing Umgebungen stellt eine nichttriviale Aufgabe dar. Durch Gerätemobilität, schwankende Netzwerkverbindungen oder sich ändernde physikalische Kontexte sind die zur

Verfügung stehenden Hardware- und Softwareressourcen permanent fluktuierend. Als Konsequenz hieraus müssen Anwendungen vor ihrer tatsächlichen Ausführung zunächst konfiguriert werden, um sicher zu stellen, dass die von der Anwendung benötigten Ressourcen auch wirklich zur Verfügung stehen. Darüber hinaus müssen Anwendungen in der Lage sein, sich während ihrer Ausführungszeit permanent an wechselnde Ausführungsumgebungen zu adaptieren. Abbildung 1 zeigt eine beispielhafte Messenger-Anwendung. Diese muss sich im Laufe ihrer Ausführung durch Nutzermobilität permanent an die wechselnde Ressourcenverfügbarkeit anpassen.

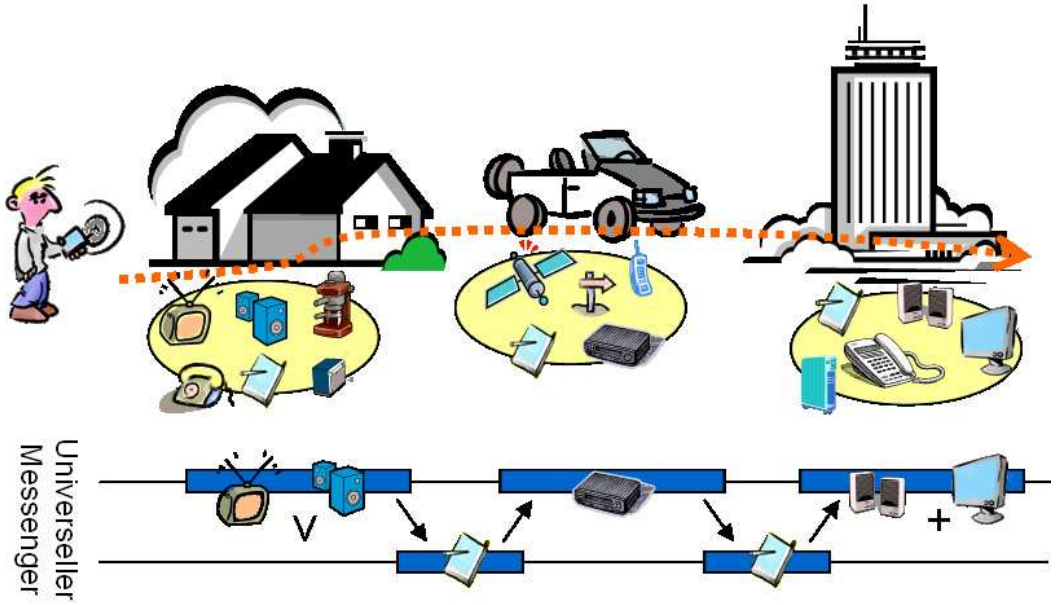


Abb. 1. Adaption einer beispielhaften Messaging-Anwendung

Das hier beschriebene Forschungsprojekt beschäftigte sich mit der Konfiguration und Adaption verteilter Anwendungen. Von besonderer Bedeutung hinsichtlich der Anwendbarkeit und Flexibilität des zu entwickelnden Systems war dabei die Möglichkeit, solche Konfigurationen und Adaptionen vollkommen automatisch und ohne Nutzerinteraktion in verschiedensten Umgebungen durchführen zu können. Um dieses Ziel zu erreichen, mussten neue Algorithmen entwickelt werden, welche sowohl die zur Verfügung stehenden Ressourcen als auch die Anforderungen der Anwendungen sowie Nutzerziele berücksichtigen.

Im Rahmen dieses Projekts sollte außerdem sowohl die Heterogenität der Geräte als auch die Heterogenität der Kommunikationstechnologien berücksichtigt werden. Das Gesamtziel dieses Projekts war es, unterstützende Systemsoftware zu kon-

zipieren, welche minimal bezüglich ihrer Ressourcenanforderungen ist, um ressourcenschwache Geräte zu unterstützen. Gleichzeitig sollte die zu entwickelnde Softwarearchitektur allerdings auch flexibel genug sein, um die effiziente Integration von Ressourcen auf leistungsstärkeren Geräten zu ermöglichen.

Das beschriebende Forschungsvorhaben ist eingebettet in das Projekt *3PC - Peer to Peer Pervasive Computing* an der Universität Stuttgart. Dieses beschäftigt sich seit seinem Beginn im Jahr 2003 mit der Aufgabe, eine automatische Unterstützung für adaptive Anwendungen durch Systemsoftware in Peer to Peer-basierenden spontanen Netzwerken zu realisieren. Hierbei werden in verschiedenen Teilprojekten unterschiedliche Aspekte dieser Systemunterstützung wie energieeffiziente Diensterkennungsmethoden, Sicherheitsprotokolle für ressourcenschwache Geräte oder die Unterstützung prozessbasierter Anwendungen erforscht.

2. Stand der Forschung bei Projektbeginn

Zu Beginn dieses Projekts existierten bereits zahlreiche verwandte Arbeiten, welche sich mit der Konfiguration von verteilten Anwendungen des Pervasive Computing beschäftigten.

Diese Projekte konnten im Wesentlichen in zwei Gruppen eingeteilt werden: Projekte zur Anwendungskonfiguration in mobilen Ad Hoc Netzen, und Projekte zur Anwendungskonfiguration in infrastrukturbasierten *Smart Environments*. Eine zusätzliche Diversifizierung konnte durch die Unterscheidung, ob die Konfiguration und Adaption automatisch vom System oder manuell durch den Anwendungsprogrammierer oder Benutzer vorgenommen werden muss, erreicht werden.

Infrastrukturen wie *Gaia* [12] oder *Aura* [15] konzentrierten sich auf Fragestellungen, die sich durch die Integration von Rechnersystemen in infrastrukturgestützten Umgebungen ergeben. Bei der Konzeption dieser Infrastrukturen wurde allerdings davon ausgegangen, dass bestimmte Rechnersysteme ständig für notwendige Koordinationsaufgaben zur Verfügung stehen, daher waren diese Systeme nicht in einfachen Ad Hoc Umgebungen anwendbar.

Projekte wie *Mobile Gaia* [4] oder *P2PComp* [6] benötigten zwar keine speziellen Infrastrukturen und konnten demnach in sämtlichen Umgebungen eingesetzt werden, allerdings nutzten diese Systeme zusätzlich vorhandene Rechenressourcen von leistungsstarken Infrastrukturgeräten nur ineffizient. Dadurch konnten in heterogenen Umgebungen nur suboptimale Ergebnisse hinsichtlich der Geschwindigkeit der Konfigurations- und Adaptionsprozesse erzielt werden.

Andere Systeme für Ad Hoc Umgebungen basierten hingegen nicht auf vollautomatisierter Anwendungskonfiguration, sondern legten die Verantwortung hierfür

entweder in die Hände des Anwenders (z.B. *Speakeasy*, [5]) oder des Programmierers (z.B. *one.world*, [7]).

Aus diesem Überblick wird deutlich, dass keines der bisher existierenden Projekte eine effiziente automatische Anwendungskonfiguration sowohl in Ad Hoc Umgebungen als auch in infrastrukturbasierten Szenarien bereitstellte. Darüber hinaus nutzte keines der genannten Systeme die Ergebnisse vergangener Konfigurationen für zukünftig anstehende Konfigurationsprozesse, um die Konfigurationslatenzen sowie die Konfigurationslast der involvierten Geräte weiter zu verringern.

Daher war ein wesentliches Ziel dieses Projekts, Konzepte, Mechanismen und Algorithmen zur effizienten automatischen Unterstützung von Anwendungskonfigurationen in verschiedensten Pervasive Computing Umgebungen zu entwickeln. Hierdurch sollte ein System entstehen, welches hinsichtlich seiner Flexibilität in dynamischen Umgebungen und der Berücksichtigung des Nutzerverhaltens allen bisherigen Projekten überlegen sein würde. Dies würde einen wesentlichen Fortschritt im Bereich der ubiquitären Anwendungskonfiguration bedeuten.

Als Basis für sämtliche Forschungen konnte auf die in eigener Vorarbeit entwickelte Softwareinfrastruktur für spontan vernetzte Systeme zurückgegriffen werden. Diese bestand aus dem Komponentensystem PCOM [2] sowie der Kommunikationsmiddleware BASE [3].

3. Wissenschaftliche Fortschritte

Das hier präsentierte Forschungsvorhaben wurde im August 2004 in die zweite Förderperiode des *Schwerpunktprogramms 1140 - Basissoftware für Selbstorganisierende Infrastrukturen für Vernetzte Mobile Systeme* aufgenommen und verblieb in der dritten Förderperiode des Schwerpunktprogramms. Somit betrug die Gesamtförderungsdauer des Projekts vier Jahre.

Im Rahmen dieses Projekts sollten zunächst Konzepte und Algorithmen zur Unterstützung adaptiver Anwendungen in dynamischen, stark ressourcenbeschränkten Systemen erforscht werden. Die automatisierte Konfiguration und Adaption von verteilten komponentenbasierten Anwendungen stand hierbei im Vordergrund, um sowohl den Entwicklungsaufwand für die Anwendungsprogrammierer zu reduzieren, als auch durch die Automatisierung den Benutzerkomfort zu erhöhen. Als Gesamtziel sollte ein integrierter Algorithmus konzipiert und umgesetzt werden, der Anwendungen unter Berücksichtigung der Ausführungsumgebung gemäß den Präferenzen des Anwenders automatisch optimal konfiguriert und anpasst.

Dafür wurde zunächst ein initialer Algorithmus zur automatischen, komplett verteilten Anwendungskonfiguration entwickelt. Anschließend wurde eine Erwei-

terung dieses Algorithmus konzipiert und umgesetzt, welche die Anpassung von zustandsbehafteten Komponenten und Anwendungen zur Laufzeit erlaubt. Danach wurden Qualitäts- und Kostenmaße untersucht, mit denen neben den Anforderungen von Anwendungen und den Ressourcenbeschränkungen der Geräte auch die Präferenzen von Anwendern berücksichtigt werden können. Schließlich wurde ein Softwareentwicklungsprozess vorgeschlagen, welcher sich an der Entwicklung moderner Desktopanwendungen orientiert und die anwenderspezifische Anpassung komponentenbasierter Anwendungen erlaubt.

In diesen ersten Arbeiten wurde vereinfachend davon ausgegangen, dass die in einer Umgebung verfügbaren Geräte hinsichtlich ihrer Netzanbindung relativ homogen sind. Des Weiteren wurde vorausgesetzt, dass alle Geräte stark ressourcenbeschränkt sind, ihre Netzkonnektivität unvorhersehbaren Schwankungen unterliegt und Anwender zu jedem Zeitpunkt beliebige Anwendungen nutzen werden. Die entwickelten Algorithmen stellen daher minimale Anforderungen an die beteiligten Geräte. Sie sind deshalb auch in heterogenen Umgebungen, die neben ressourcenschwachen auch ressourcenreiche Geräte mit stabiler Netzanbindung beinhalten, einsetzbar und in diesem Sinne generisch. Allerdings können sie das Optimierungspotential, das in heterogenen Umgebungen besteht, nicht nutzen, da eine Differenzierung von Geräten hinsichtlich ihrer Ressourcenlage nicht vorgenommen wird. Darüber hinaus wurde das in vielen Umgebungen spezifische Verhalten der Anwender nicht berücksichtigt, wodurch in Konfigurationsprozessen häufig ein Mehraufwand entstand. Um dieses Potential zu erschließen, wurden im weiteren Verlauf des Projekts zusätzlich heterogene Umgebungen betrachtet, d.h. bei der Verteilung des Konfigurationsprozesses musste die unterschiedliche Ressourcenlage und Netzanbindung der beteiligten Geräte berücksichtigt werden. Hierdurch sollte, abhängig von der aktuellen Anwendungsumgebung, ein optimierter Konfigurationsprozess mit minimierten Latenzen realisiert werden.

Dafür waren mehrere aufeinander aufbauende Arbeiten nötig. Zunächst musste eine Erweiterung der Systemarchitektur konzipiert werden, um den automatischen Austausch des Konfigurationsalgorithmus basierend auf den vorherrschenden Umgebungsbedingungen zur Laufzeit zu erlauben. Zur effizienten Nutzung ressourcenstarker Geräte in verschiedensten Umgebungen waren außerdem speziell angepasste Konfigurationsverfahren nötig, um die Konfigurationslatenzen zu minimieren und hierdurch den Anwendungsnutzern eine maximale Dienstgüte zu garantieren. Darüber hinaus sollte durch die Ausnutzung früherer Konfigurationsergebnisse eine Optimierung des Konfigurationsprozesses besonders in niedrigdynamischen Umgebungen erreicht werden.

Die entsprechend entwickelten Konzepte und Mechanismen werden in den folgenden Abschnitten detaillierter diskutiert.

3.1. Entwicklung eines Algorithmus zur initialen Konfiguration von Anwendungen

Die automatische Anwendungskonfiguration umfasst die Aufgabe, automatisch eine Komposition von Anwendungsteilen, sogenannten *Komponenten*, zu ermitteln, welche in ihrer Gesamtheit eine gültige Anwendung darstellen. Hierbei müssen zwei Klassen von Bedingungen erfüllt werden: *Strukturelle Bedingungen*, die festlegen, welche Kompositionen hinsichtlich der zu erbringenden Funktionalitäten gültig sind, und *Ressourcenbedingungen* aufgrund beschränkter Ressourcenverfügbarkeit. Daher musste zunächst ein Algorithmus entwickelt werden, welcher in der Lage ist, Anwendungen initial in den betrachteten Umgebungen zu konfigurieren, wobei beide Klassen von Bedingungen gleichzeitig erfüllt werden. Weitere Bedingungen an den Algorithmus waren, dass dieser vollständig, effizient und ausfallsicher sein sollte.

Um dieses Ziel zu erreichen, wurde zunächst gezeigt, dass das Problem der automatischen Bestimmung einer Konfiguration auf ein verteiltes Bedingungserfüllungsproblem (*Distributed Constraint Satisfaction Problem, DCSP*) zurückgeführt werden kann und somit NP-vollständig ist. Ein DCSP umfasst eine Menge an Variablen V mit zugehörigem Wertebereich W und zu erfüllenden Bedingungen B zwischen den Variablen. Zur Zurückführung der automatischen Konfiguration auf ein DCSP wurde gezeigt, dass Abhängigkeiten zwischen Komponenten auf V, vorhandene Komponenten auf W und Ressourceabhängigkeiten auf B abgebildet werden können [8]. Dadurch ist es möglich, Algorithmen zur Lösung eines DCSP auch für die automatische Konfiguration zu verwenden. Die Problemtransformation muss dabei zur Laufzeit vorgenommen werden und darf kein globales Wissen hinsichtlich der verfügbaren Komponenten und Ressourcen erfordern, um einen hohen Kommunikationsaufwand in ressourcenreichen Umgebungen zu vermeiden.

Um eine flexible Einsetzbarkeit der Algorithmen zu gewährleisten, wurde außerdem nicht von einer durchgängigen Verfügbarkeit eines ressourcenreichen Geräts ausgegangen, weshalb die Berechnung der Konfiguration über die verfügbaren Geräte verteilt werden musste und zentralisierte Berechnungsalgorithmen ausgeschlossen werden konnten. Da die Algorithmen in der Lage sein sollten, in jeder Umgebung eine Konfiguration zu finden, sofern diese existiert, konnten heuristische Ansätze ebenfalls ausgeschlossen werden. Allerdings musste die Laufzeit des Konfigurationsalgorithmus mit der von heuristischen Verfahren vergleichbar sein.

Auf dieser Grundlage wurden existierende Algorithmen aus dem Forschungsbereich der verteilten künstlichen Intelligenz hinsichtlich ihrer Eignung als Konfigurationsalgorithmus untersucht. Dabei stellte sich der *Asynchronous Backtracking (ABT)* Algorithmus von Yokoo [17] als geeigneter Algorithmus heraus, da es sich hierbei um einen komplett verteilten, vollständigen und effizienten abhängigkeits-

gerichteten Algorithmus handelt. Zur zusätzlichen Sicherung der Fehlertoleranz wurde ABT um Soft-State-Mechanismen zur Verwaltung des verteilten Algorithmenzustands sowie ein Terminierungsprotokoll, das die Beendigung des Konfigurationsalgorithmus trotz möglicher Verbindungsabbrüche feststellen kann, erweitert. Um die Nichtverfügbarkeit bereits konfigurierter Anwendungsteile in geeigneter Weise im Ablauf des Algorithmus zu berücksichtigen, wurde der Algorithmus außerdem erweitert, sodass zur Laufzeit zusätzliche Bedingungen in das Bedingungserfüllungsproblem eingefügt werden können. Der Vergleich mit einem einfachen heuristischen Verfahren zeigte, dass der vollständige Ansatz auf Basis von asynchronem Backtracking alle möglichen Konfigurationen findet und dem heuristischen Ansatz in den zu erwartenden Szenarien überlegen ist. Weitere Details zu diesem Algorithmus sind in [8], [9] zu finden.

3.2. Entwicklung eines Algorithmus zur automatischen Anpassung zustandsbehafteter Anwendungen

Durch den im vorigen Abschnitt beschriebenen Algorithmus konnten verteilte Anwendungen automatisch initial konfiguriert werden. Allerdings berücksichtigte dieser Algorithmus nicht, dass sich in dynamischen Szenarien häufig die Ressourcenbedingungen ändern, weshalb es erforderlich werden kann, Anwendungen zur Laufzeit zu adaptieren. Deshalb musste der bestehende Algorithmus so erweitert werden, dass er Anwendungen unter Berücksichtigung der entstehenden Kosten zur Laufzeit anpassen kann. Dazu mussten zunächst die Kostenfaktoren für Anpassungen bestimmt und in einem geeigneten Modell zusammengefasst werden.

Bei der Analyse der Kostenfaktoren stellten sich die Kosten für die Migration von Anwendungszuständen sowie die Kosten für die Berechnung einer neuen Konfiguration als wichtigste Einflussfaktoren heraus. Zur Untersuchung der Migrationskosten wurde ein auf Aufrufhistorien und Schnappschüssen basierendes Verfahren in die bestehende Systemsoftware integriert, das den Zustand von Komponenten halb-automatisch wiederherstellen und migrieren kann. Dadurch war es möglich, die Größe des zu transferierenden Zustands für jede Komponente zur Laufzeit abzuschätzen. Die genaue Funktionsweise dieses Verfahrens wird in [11] beschrieben.

Auf dieser Basis wurde im Anschluss ein allgemeines Kostenmodell entwickelt, mit dem für jede Komponente die Größe des Zustands und damit der Kommunikationsaufwand für den Wechsel bestimmt werden kann. Hinsichtlich der Berechnungskosten für eine neue Konfiguration ergab sich die Forderung, dass diese im Adaptionsfall die Kosten für die Berechnung einer initialen Konfiguration nicht übersteigen sollten. Daher konnten die Adaptionskosten nur heuristisch optimiert werden, weil die Berechnung einer optimalen Anwendungskonfiguration im Allgemeinen die Berechnung aller möglichen Anwendungskonfigurationen voraus-

setzt. Auf der Grundlage des Kostenmodells für die Migration von Anwendungszuständen wurde eine einfache Greedy-Heuristik in den bestehenden Konfigurationsalgorithmus integriert, welche Adaptionen gemäß dem im Bereich der Bedingserfüllungsprobleme häufig verwendeten *Value Ordering*-Verfahren durchführt. Bei diesem Verfahren wird derjenige Wert einer Variablen als nächstes gewählt, welcher die meisten Optionen für folgende Variablen offen läßt, wodurch die Effizienz von Adaptionsvorgängen durch eine geringere Anzahl nötiger Adaptionen deutlich verbessert wird. In Evaluationen wurden dementsprechend in den meisten Szenarien gute Ergebnisse mit geringen Adaptionslatenzen erzielt, ohne die Kosten für die Berechnung der angepassten Konfiguration zu erhöhen.

3.3. Bewertung der ermittelten Konfigurationen

Um die Bewertung von gefundenen Konfigurationen hinsichtlich vorgegebener Präferenzen zu ermöglichen, wurde eine Untersuchung anwenderspezifischer Qualitätskriterien bei der automatischen Konfiguration und Adaption von Anwendungen durchgeführt. Hierdurch wurde der zuvor entwickelte Algorithmus erweitert, um mögliche Konfigurationen hinsichtlich vorgegebener Benutzerpräferenzen zu bewerten.

Dafür wurden zunächst relevante Qualitätskriterien untersucht, und es wurden in einem vorbereitenden Schritt Entwicklungswerkzeuge konzipiert, mit denen Anwender komponentenbasierte Anwendungen an ihre Bedürfnisse anpassen können [16]. Um den Softwareentwicklungsprozess, der sich an der Entwicklung moderner Desktopanwendungen orientiert, im Pervasive Computing anwenden zu können, wurde eine exemplarische Architektur konzipiert, die aus dem Komponentensystem *PCOM*, der grafischen Programmiersprache *Nexel* sowie der in PCOM eingebetteten Konfigurationsalgorithmen besteht. Der Entwicklungsprozess besteht dabei aus drei Phasen: der Entwicklung kommerzieller Komponenten und Anwendungen als Software für PCOM (*Development*), der Anpassung der Software an benutzerspezifische Wünsche mittels Nexel (*Customization*), und schließlich der tatsächlichen Verwendung dieser Software durch die Benutzer innerhalb von PCOM-Anwendungen (*Utilization*). Dieser Entwicklungsprozess sowie die verwendeten Werkzeuge sind schematisch in Abbildung 2 dargestellt.

Ausgehend von diesen Werkzeugen sowie der bereits bestehenden Modellierung nicht-funktionaler Eigenschaften von Komponenten wurde dann ein Modellierungskonzept entwickelt, mit welchem Anwender ihre Präferenzen bezüglich unterschiedlicher Konfigurationen in Form von *Gruppierungen* mittels nicht-funktionalen Eigenschaften spezifizieren können. Dadurch müssen für die Optimierung von Anwendungen nicht sämtliche Konfigurationen berechnet werden, sondern der Berechnungsaufwand lässt auf ein mehrfaches Ausführen des Konfigurationsalgo-

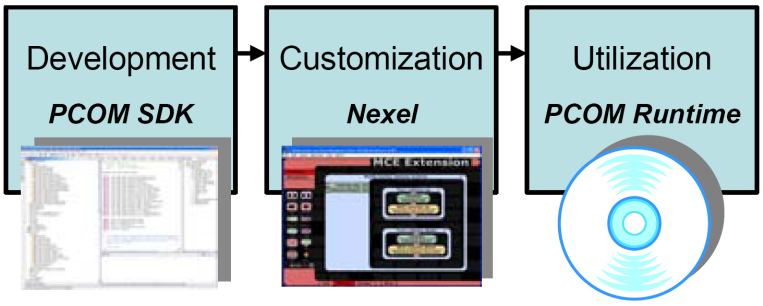


Abb. 2. Entwicklungsprozess und verwendete Werkzeuge

rithmus reduzieren. Gleichzeitig können hierbei noch lineare Abhängigkeiten, die zwischen verschiedenen Gruppierungen auftreten können, zur weiteren Effizienzsteigerung der Konfiguration genutzt werden.

3.4. Unterstützung von austauschbaren, spezialisierten Konfigurationsalgorithmen

In den bisher beschriebenen Arbeiten wurden lediglich komplett verteilte Konfigurationsalgorithmen berücksichtigt. Allerdings sind in heterogenen Umgebungen effektivere Konfigurationsmethoden denkbar. Ein flexibler Mechanismus zur Bereitstellung austauschbarer Konfigurationsalgorithmen erschien daher vorteilhaft. Abhängig von der aktuellen Ausführungsumgebung soll dann der jeweils passendste Konfigurationsalgorithmus gewählt werden. Dafür wurde eine erweiterte Architektur entwickelt, die eine automatische Anpassung des zu verwendenden Algorithmus erlaubt [10].

Diese Erweiterung ermöglicht eine Abstrahierung der Konfigurationsberechnung von der eigentlichen Ausführung der Anwendung, indem die Funktionalität des bisherigen PCOM-Komponentencontainers auf drei verschiedene Bausteine aufgesplittet wird, welche über Schnittstellen miteinander kommunizieren können. Der *Application Manager* ist hierbei für das Lifecycle Management von Anwendungen und die Auswahl eines passenden Konfigurationsalgorithmus verantwortlich. Durch ihn können Anwendungen gestartet, ausgeführt und beendet werden. Mittels der *Assembler* werden gültige Anwendungskonfigurationen berechnet, sie repräsentieren somit die Implementierung der Konfigurationsalgorithmen. Je nach Gerät und Ausführungsumgebung können verschiedene Assembler-Implementierungen verfügbar sein, beispielsweise ein komplett verteilter Assembler für Ad Hoc Umgebungen und ein komplett zentraler Assembler für ressourcenreiche Infrastrukturumgebungen. Der neue *Container* ist nun lediglich für die Verwaltung der Ressourcen und Komponenten auf den einzelnen Geräten ver-

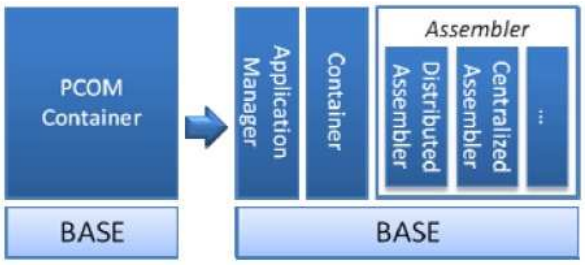


Abb. 3. Unterstützung austauschbarer Konfigurationsalgorithmen in PCOM

antwortlich. Die in Abbildung 3 dargestellte erweiterte Architektur stellt somit die Voraussetzungen für die unabhängige Entwicklung spezialisierter Konfigurationsalgorithmen. In Evaluationen zeigte sich, dass in verschiedenen Umgebungen unterschiedliche Grade der Verteiltheit bei der Konfigurationsberechnung zu geringsten Konfigurationslatenzen führen können.

3.5. Entwicklung eines effizienten Algorithmus für heterogene Umgebungen

Nachdem die architektonische Unterstützung verschiedener Konfigurationsalgorithmen umgesetzt wurde, musste folglich ein Konfigurationsalgorithmus entworfen werden, welcher in heterogenen Umgebungen die zusätzlichen Rechenressourcen effizient ausnutzt. Um mögliche Flaschenhälse bei der Konfiguration zu vermeiden, wurden die ressourcenschwachen mobilen Geräte wie PDAs oder Smartphones von der Berechnung ausgeschlossen. Sie mussten lediglich im Voraus Informationen über die von ihnen zur Verfügung gestellten Komponenten bereitstellen. Die einfachste Konfigurationsmethode bestand daher darin, die Konfiguration komplett zentral auf dem ressourcenstärksten Gerät in der Umgebung berechnen zu lassen.

Dafür wurden zunächst existierende zentrale Backtracking-Algorithmen aus dem Bereich des maschinellen Lernens untersucht und ein erweiterter Algorithmus namens *Direct Backtracking* entworfen [14]. Dieser Algorithmus baut auf dem auf Tiefensuche basierenden *Synchronous Backtracking (SBT)* [17] Algorithmus, einer zentralen und synchronen Variante von Asynchronous Backtracking, auf. Zur Beschleunigung des Konfigurationsprozesses wurde SBT um zwei zusätzliche Mechanismen erweitert: *proaktive Adaptionsvermeidung* und *intelligentes Backtracking*.

Die *proaktive Adaptionsvermeidung* wird in der Weise ausgeführt, dass während eines Konfigurationsvorgangs im Falle verschiedener zur Verfügung stehender Komponenten, die dieselbe geforderte Funktionalität bereitstellen, diejenige Komponente ausgewählt wird, welche eine minimale Anzahl an Ressourcen auf dem Gerät verbraucht, auf dem sie ansässig ist. Hierdurch wird die Wahrscheinlichkeit zukünftiger Ressourcenkonflikte auf dem entsprechenden Gerät während einer

Konfiguration minimiert, da die Ressourcen schonend vergeben werden. Dadurch sinkt die Anzahl nötiger Adaptionsvorgänge während einer Konfiguration.

Bei bestimmten Anwendungskonstellationen können Adaptionen jedoch dennoch unvermeidbar werden. In diesem Fall muss die Menge der für eine Anwendung ausgewählten Komponenten durch einen Adaptionsvorgang angepasst werden. Hierbei muss berücksichtigt werden, dass die Adaption einer Komponente zusätzlich die Adaption anderer Komponenten nach sich ziehen kann, um weiterhin sämtliche struktulle Bedingungen der Anwendung zu erfüllen. Um diese Adaptionen mit geringst möglichem Aufwand durchzuführen, wurde ein *intelligenter Backtrackingmechanismus* eingeführt. Dieser basiert auf dem Konzept, im Falle mehrerer möglicher Komponenten, welche adaptiert werden können, diejenige Auswahl anzupassen, welche mit geringst möglichem Aufwand durchgeführt werden kann und möglichst wenig zusätzliche Adaptionen nach sich zieht. Zur Bestimmung des Adaptionsaufwands wurde das zuvor entwickelte und in Abschnitt 3.2. beschriebene Kostenmodell verwendet.

In Evaluationen ergab sich durch Nutzung dieser erweiterten Mechanismen für Direct Backtracking ein im Vergleich zu SBT signifkant beschleunigter Konfigurationsprozess in ressourcenreichen Umgebungen, welcher die Konfigurationslatenz bis auf unter 10 % der entsprechenden Latenz von SBT senkte.

3.6. Entwicklung eines Rahmenwerks zur Anpassung des Grades der Verteiltheit

Um den Grad der Verteilung des Konfigurationsvorgangs optimal an die momentane Anwendungsumgebung anzupassen, mussten die in der Umgebung befindlichen Geräte entsprechend vorbereitet werden. Dafür mussten insbesondere drei Probleme gelöst werden:

- *Die automatische Anpassung des Verteilungsgrades der Konfigurationsberechnung:* Um verschiedene Umgebungen effizient zu unterstützen, war die automatische Auswahl eines passenden Konfigurationsalgorithmus nötig. Dafür wurde ein Verfahren entwickelt, welches zunächst basierend auf der Art der verfügbaren Geräte den Umgebungstyp – ressourcenschwache Ad Hoc Umgebung oder ressourcenstarke Infrastrukturumgebung – ermittelt. Im Falle einer Ad Hoc Umgebung wurde anschließend der verteilte, im Falle einer Infrastrukturumgebung der zentrale Konfigurationsalgorithmus ausgewählt. Dieser Selektionsmechanismus wurde hierbei so konzipiert, dass er einfach zur Unterstützung zusätzlicher Konfigurationsmethoden erweitert werden konnte.
- *Die automatische Ermittlung der Geräte, welche in ressourcenreichen Umgebungen die Berechnung der Konfiguration übernehmen:* Hierfür wurde ein auf Clustering basierendes Rahmenwerk entworfen und in PCOM integriert. Um die Clusterstruktur

zu ermitteln, wurde Basagni's *Distributed Clustering* Algorithmus [1] verwendet, als tatsächliches Clusteringkriterium wurden die vorhandenen Rechenressourcen auf den vorhandenen Geräten gewählt. Das ressourcenreichste Gerät in der Umgebung wurde dann zum *Clusterhead* bestimmt und somit verantwortlich für die Konfigurationsberechnung der benachbarten Geräte gemacht.

– *Das Erlangen konfigurationsspezifischer Informationen durch den Clusterhead:* Zur Realisierung eines effizienten Konfigurationsvorgangs auf dem Clusterhead, musste dieser zunächst die für die Konfiguration relevanten Informationen der vorhandenen Geräte – im Wesentlichen deren aktuelle Ressourcen- und Komponentenverfügbarkeiten – bestimmen. Hierfür wurde ein Verfahren integriert, mit welchem dem Clusterhead die relevanten Informationen über vorhandene Ressourcen automatisch durch Analyse von Veränderungen der Clusterstruktur im Voraus zur Verfügung gestellt werden. Geräte, deren Ressourcenverfügbarkeit sich änderten, teilten dem Clusterhead dabei automatisch ihre aktualisierte Ressourcenlage mit. Der Clusterhead baut beim Empfang dieser Informationen eine interne Repräsentation der entfernten Geräte in Form von sogenannten *virtuellen Containern* auf. Der konfiguriende Assembler greift bei einem Konfigurationsprozess dann lokal auf diese virtuellen Container zu. Dieses Konzept ist in Abbildung 4 illustriert und stellt einen der Konfiguration vorgelagerten Prozess dar. Hierdurch sinkt der tatsächliche Aufwand der Konfiguration, da während der Konfiguration keinerlei Kommunikation zwischen den Geräten notwendig ist.

Das gesamte Rahmenwerk wurde in PCOM integriert. In Evaluationen ergab sich durch Ausnutzung des Vorkonfigurationsprozesses eine effiziente zentrale Konfiguration, welche zu einer deutlichen Reduktion der Konfigurationslatenzen von bis zu 84 % in heterogenen Umgebungen führte [13].

3.7. Hybride Anwendungskonfiguration

Um das komplette Spektrum zwischen vollständig zentraler Konfiguration in stark heterogenen Umgebungen und vollständig verteilter Konfiguration in homogenen Umgebungen abzudecken, war ein erweitertes Verfahren nötig, welches die Vorteile der verteilten Konfiguration (generelle Anwendbarkeit in sämtlichen Umgebungen, keine Single-Point-of-Failure Problematik) mit denen der zentralen Konfiguration (effiziente Nutzung ressourcenstarker Geräte, geringer Kommunikationsoverhead) vereint.

Aufbauend auf den bisher konzipierten Konfigurationsmethoden wurde daher ein Verfahren entwickelt, welches eine optimierte hybride Anwendungskonfiguration ermöglicht. Die Idee hierbei war, eine Teilmenge aller Geräte zu bestimmen, welche dann für die restlichen Geräte die Konfiguration ihrer Komponenten und Res-

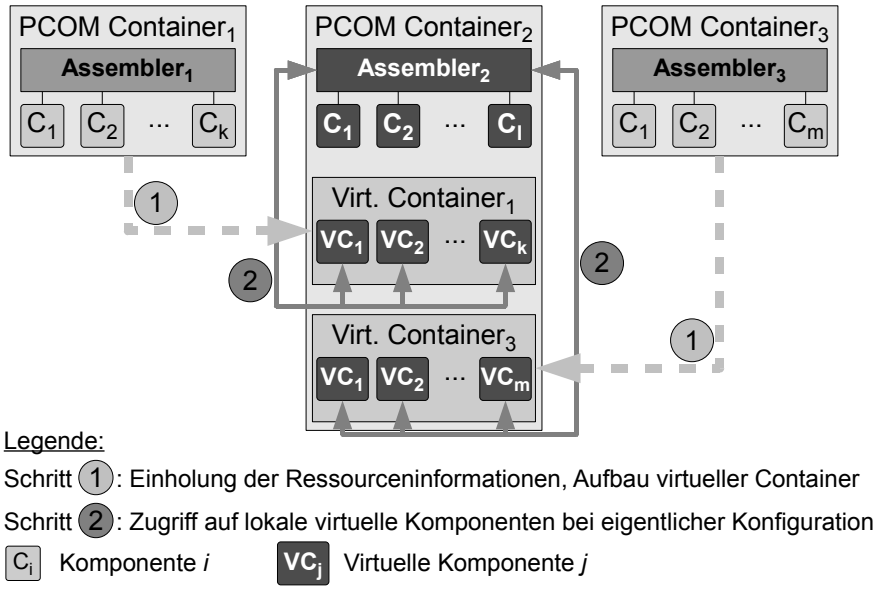


Abb. 4. Konzept der *virtuellen Container*

sourcen übernehmen. Um die Effizienz des Verfahrens zu garantieren, sollten unter Nutzung des Clustering-Rahmenwerks lediglich die ressourcenstarken Geräte wie Laptops, Desktop-PCs oder Server aktiv in die Konfiguration eingebunden werden. Hierfür wurde zunächst der im vorigen Abschnitt vorgestellte Selektionsmechanismus erweitert, um in Umgebungen mit mehreren ressourcenstarken Geräten eine Konfigurationsberechnung auf genau diesen Geräte zu ermöglichen.

Anschließend musste sichergestellt werden, dass eine Abbildung der ressourcenschwachen Geräte auf ressourcenstarke Geräte erfolgt, welche dann lokal für die ihnen zugeordneten schwachen Geräte virtuelle Container aufbauen und diese in den Konfigurationsprozess einbinden. Damit eine ausgeglichene Konfigurationslast zwischen den ressourcenstarken Geräten erreicht wird, wurde ein *Weight Balanced*-Verfahren integriert, das gemäß dem Round-Robin-Verfahren die Gerätezuordnung vornimmt. Hierdurch wird sichergestellt, dass jedem starken Gerät annähernd gleich viele schwache Geräte zugeordnet werden und somit die Konfigurationslast gleichverteilt wird. Um dynamische Umgebungen zu berücksichtigen, wurde zusätzlich eine automatische Aktualisierung der Geräteabbildungen integriert. Zur automatischen Erkennung von Änderungen der Geräteumgebung wird hierbei der in BASE vorhandene *Lease*-Mechanismus verwendet. Im Falle neu hinzugekommener oder nicht mehr verfügbarer ressourcenstarker und -schwacher Geräte wird durch eine Neugruppierung der Geräte die ausgeglichene Konfigurationslast auf den starken Geräten garantiert.

Abschließend wurde ein erweiterter Konfigurationsalgorithmus konzipiert, bei welchem die ermittelten ressourcenstarken Geräte jeweils zentral die Teilkonfigurationen für die ihnen zugeordneten Geräte berechnen, um anschließend untereinander die ermittelten Teilkonfigurationen auszutauschen und somit die Gesamtkonfiguration der Anwendung zu bestimmen.

In Evaluationen konnte gezeigt werden, dass in Umgebungen mit verschiedenen Graden an Heterogenität jeweils unterschiedliche Konfigurationsansätze zu den geringsten Konfigurationslatenzen führten. So war in homogenen ressourcenschwachen Umgebungen die verteilte Anwendungskonfiguration am schnellsten, während in schwach heterogenen Umgebungen die zentralisierte und in stark heterogenen Umgebungen die hybride Anwendungskonfiguration zu besten Ergebnissen hinsichtlich der Konfigurationslatenz führten. Auf diesen Ergebnissen aufbauend wurde schließlich eine einfache Verteilungsheuristik abgeleitet, welche abhängig vom Grad der Heterogenität der Umgebung die automatische Auswahl der in diesem Szenario passenden Konfigurationsmethode durchführt.

3.8. Einbindung der Ergebnisse voriger Konfigurationsvorgänge in den Konfigurationsprozess

In vielen typischen Szenarien wie Universitätshörsälen werden häufig identische Anwendungskomponenten auf denselben Geräten wiederverwendet. Diese Anwendungskomponenten beschreiben sogenannte *partielle Anwendungskonfigurationen*. Allerdings wurden diese bisher nicht genutzt, weshalb in solchen Fällen trotzdem jeweils der komplette Konfigurationsprozess für jede einzelne Komponente durchgeführt werden musste. Dies erzeugte jedoch hohen Rechenaufwand auf den Konfigurationsgeräten sowie entsprechende Konfigurationslatenzen und hohen Kommunikationsaufwand. Daher entstand die Idee, die Ergebnisse voriger Konfigurationsprozesse in aktuelle Konfigurationsvorgänge zu integrieren, um das Konfigurationsproblem effektiv zu reduzieren. Dafür mussten Verfahren entwickelt werden, um automatisch partielle Anwendungskonfigurationen zu bestimmen und zu speichern, effektiv an die in der Umgebung vorhandenen Geräte zu verteilen und automatisch in zukünftigen Konfigurationsvorgängen einzubinden.

Zur Speicherung und Wiedereinlesung dieser partiellen Anwendungskonfigurationen wurde zunächst ein auf XML basierendes Verfahren entwickelt, welches nach einem erfolgreichen Konfigurationsvorgang automatisch sämtliche entstehenden Teilkonfigurationen lokal speichert. Um diese Teilkonfigurationen anschließend zu verteilen und die Konsistenz unter den vorhandenen Geräten zu gewährleisten, wurden unter Nutzung der BASE-Kommunikationsmechanismen alle erzeugten Teilkonfigurationenen per Broadcast an die Geräte in der Umgebung gesendet.

Außerdem mussten die bisher verwendeten Konfigurationsalgorithmen erweitert werden, damit sie für gespeicherte partielle Konfigurationen automatisch die Verfügbarkeit der von ihnen verwendeten Komponenten überprüfen und diese dann in den Konfigurationsprozess einbeziehen konnten. Hierfür wurde zunächst das in in Abschnitt 3.2. beschriebene Kostenmodell soweit angepasst, dass auch partielle Anwendungskonfigurationen in ihm berücksichtigt werden können. Dies ermöglicht eine initiale Bewertung, ob bei anstehenden Konfigurations- und Adaptionsprozessen eine Verringerung der entstehenden Latenz durch die Nutzung einer partiellen Anwendungskonfiguration erwartet werden kann. Basierend auf vorhergegangenen Konfigurationen und Adaptationen werden außerdem Schlüsse für zukünftige Konfigurations- und Adaptionsprozesse gezogen und die erwarteten Kosten für zukünftige Anwendungskonfigurationen ausgehend von der mitgespeicherten Ausführungshistorie dynamisch angepasst. Die tatsächliche Erweiterung der Konfigurationsalgorithmen geschah, indem die Algorithmen insofern angepasst wurden, dass sie bei der Konfiguration von Komponenten zunächst mit Hilfe des erweiterten Kostenmodells überprüfen, ob vorhandene gültige Teilkonfigurationen genutzt werden sollten, da sie zu einer Verringerung der Konfigurationslatenz beitragen. Nur wenn keine nutzbaren Teilkonfigurationen integriert werden können, muss eine gültige Belegung der jeweiligen Komponenten berechnet werden.

Die Nutzung des Konzepts partieller Anwendungskonfigurationen führte zu einer weiteren signifikanten Reduktion der Konfigurationslatenz, vor allem bei zentraler Anwendungskonfiguration in heterogenen Umgebungen mit geringer Umgebungsdynamik. Hier konnte durch Verwendung von Teilkonfigurationen die Konfigurationslatenz auf unter 50 ms reduziert werden, was lediglich ein Viertel der Latenzen der bisher umgesetzten Algorithmen darstellt.

4. Zusammenfassung und Ausblick

Im Rahmen dieses Projekts wurden Algorithmen zur automatischen Konfiguration und Adaption komponentenbasierter Anwendungen mit Hilfe der in der Umgebung verfügbaren Geräte entwickelt. Hauptziel der Untersuchungen war die Erforschung von elementaren Konzepten zur Systemunterstützung in Pervasive Computing-Umgebungen. Hierfür wurden zunächst vollautomatisierte Algorithmen zur initialen Konfiguration sowie zur Anpassung von Anwendungen unter Berücksichtigung von Nutzerpräferenzen und vorgegebenen Qualitätsmaßen entworfen. Außerdem wurden erweiterte Strategien und Mechanismen entwickelt, um den Grad der Konfigurationsverteilung automatisch anzupassen und die zur Verfügung stehenden Rechenressourcen optimal auszunutzen. Darüber hinaus wurde ein Verfahren entworfen, um die Ergebnisse voriger Konfigurationen automatisch in den Konfigurationsprozess einbinden zu können. Durch die Gesamtheit dieser

Entwicklungen konnte das angestrebte Ziel erfolgreich erreicht werden. Dadurch konnte erstmals im Bereich des Pervasive Computing eine flexible und effiziente Anwendungskonfiguration in dynamischen Umgebungen realisiert werden, was einen wesentlichen Fortschritt im Hinblick auf die Erreichung einer entsprechenden Nutzerakzeptanz des Pervasive Computing darstellt.

In zukünftigen Arbeiten sollen erweiterte Verfahren zur intelligenten automatischen Vorberechnung und Verteilung von Konfigurationen und Teilkonfigurationen sowie zur Behandlung und Vermeidung von anwendungsübergreifenden Konflikten untersucht werden. Darüber hinaus soll das bestehende Komponentenmodell flexibilisiert werden, um alternative Konfigurationen zu ermöglichen und somit die Anzahl gültiger Konfigurationen zu erhöhen. Diese erweiterten Untersuchungen sollen zu einer weiteren Reduktion des Kommunikationsaufwandes und der Konfigurationslatenz sowie einer Erhöhung der Anwendungsverfügbarkeit führen, um der großen Vision einer transparenten Anwendungskonfiguration in ubiquitären Systemen näher zu kommen.

Literaturverzeichnis

[1] BASAGNI, STEFANO: *Distributed Clustering for Ad Hoc Networks*. In: *ISPAN '99: Proceedings of the 1999 International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN '99)*, Seiten 310–315, Washington, DC, USA, 1999. IEEE Computer Society.

[2] BECKER, CHRISTIAN, MARCUS HANDTE, GREGOR SCHIELE und KURT ROTHERMEL: *PCOM - A Component System for Pervasive Computing*. In: *2nd IEEE International Conference on Pervasive Computing and Communication (PerCom 04)*, Seiten 67–76, Orlando, Florida, USA, 2004. IEEE Computer Society.

[3] BECKER, CHRISTIAN, GREGOR SCHIELE, HOLGER GUBBELS und KURT ROTHERMEL: *BASE - A Micro-broker-based Middleware For Pervasive Computing*. In: *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications (PerCom 03)*, Seiten 443–451, Fort Worth, USA, 2003.

[4] CHETAN, SHIVA, JALAL AL-MUHTADI, ROY CAMPBELL und M. DENNIS MICKUNAS: *Mobile Gaia: a middleware for ad-hoc pervasive computing*. In: *Proceedings of the 2nd IEEE Consumer Communications and Networking Conference (CCNC 2005)*, Seiten 223–228, Las Vegas, Nevada, USA, 2005. IEEE Computer Society.

[5] EDWARDS, W. KEITH, MARK W. NEWMAN, JANA Z. SEDIVY, TREVOR F SMITH, DIRK BALFANZ, D. K. SMETTERS, H. CHI WONG und SHAHRAM IZADI: *Using Speakeasy for Ad Hoc Peer-to-Peer Collaboration*. In: *Proceedings of the ACM International Conference on Computer Supported Cooperative Work (CSCW 2002)*, Seiten 256–265, New Orleans, Louisiana, USA, 2002. ACM Press.

[6] FERSCHA, ALOIS, MANFRED HECHINGER, RENE MAYRHOFER und ROY OBERHAUSER: *A Light-Weight Component Model for Peer-to-Peer Applications*. In: *ICDCSW '04: Proceedings of the 24th International Conference on Distributed Computing Systems Workshops*

- *W7: EC (ICDCSW'04)*, Seiten 520–527, Washington, DC, USA, 2004. IEEE Computer Society.

[7] GRIMM, ROBERT: *One.world: Experiences with a Pervasive Computing Architecture*. IEEE Pervasive Computing, 3(3):22–30, 2004.

[8] HANDTE, MARCUS, CHRISTIAN BECKER und KURT ROTHERMEL: *Peer-based Automatic Configuration of Pervasive Applications*. In: *Proceedings of the 2nd IEEE International Conference on Pervasive Services (ICPS 2005)*, Seiten 249–260, Santorini, Griechenland, July 2005. IEEE Computer Society.

[9] HANDTE, MARCUS, CHRISTIAN BECKER und KURT ROTHERMEL: *Peer-based Automatic Configuration of Pervasive Applications*. Journal on Pervasive Computing and Communications, 1(4):251–264, 2005.

[10] HANDTE, MARCUS, KLAUS HERRMANN, GREGOR SCHIELE und CHRISTIAN BECKER: *Supporting Pluggable Configuration Algorithms in PCOM*. In: *PERCOMW '07: Proceedings of the Fifth IEEE International Conference on Pervasive Computing and Communications Workshops*, Seiten 472–476, Washington, DC, USA, 2007. IEEE Computer Society.

[11] HANDTE, MARCUS, GREGOR SCHIELE, STEPHAN URBANSKI und CHRISTIAN BECKER: *Adaptation Support for Stateful Components in PCOM*. In: *Workshop on Software Architectures for Self-Organization: Beyond Ad-Hoc Networking at Pervasive 2005*, München, Deutschland, 2005. Springer Verlag.

[12] ROMÁN, MANUEL, CHRISTOPHER HESS, RENATO CERQUEIRA, ANAND RANGANATHAN, ROY H. CAMPBELL und KLARA NAHRSTEDT: *Gaia: a middleware platform for active spaces*. In: *SIGMOBILE Mobile Computing and Communications Review*, Band 6, Seiten 65–67, New York, NY, USA, 2002. ACM Press.

[13] SCHUHMANN, STEPHAN, KLAUS HERRMANN und KURT ROTHERMEL: *A Framework for Adapting the Distribution of Automatic Application Configuration*. In: *Proceedings of ICPS'08: International Conference on Pervasive Services.*, Seiten 163–172, Sorrento, Italien, 2008. ACM Press.

[14] SCHUHMANN, STEPHAN, KLAUS HERRMANN und KURT ROTHERMEL: *Direct Backtracking: An Advanced Adaptation Algorithm for Pervasive Applications*. In: *Proceedings of the 21st International Conference on Architecture of Computing Systems (ARCS 2008)*, Band LNCS 4934 der Reihe *Lecture Notes in Computer Science*, Seiten 53–67, Dresden, Deutschland, 2008. Springer-Verlag.

[15] SOUSA, JOAO PEDRO und DAVID GARLAN: *Aura: an Architectural Framework for User Mobility in Ubiquitous Computing Environments*. In: *WICSA 3: Proceedings of the IFIP 17th World Computer Congress - TC2 Stream / 3rd IEEE/IFIP Conference on Software Architecture*, Seiten 29–43, Deventer, Niederlande, 2002. Kluwer, B.V.

[16] WEIS, TORBEN, MARCUS HANDTE, MIRKO KNOLL und CHRISTIAN BECKER: *Customizable Pervasive Applications*. In: *Proceedings of the 4th IEEE International Conference on Pervasive Computing and Communications (PerCom 2006)*, Seiten 239–244, Pisa, Italien, 2006. IEEE Computer Society.

[17] YOKOO, MAKOTO, EDMUND H. DURFEE, TORU ISHIDA und KAZUHIRO KUWABARA: *The Distributed Constraint Satisfaction Problem: Formalization and Algorithms*. IEEE Transactions on Knowledge and Data Engineering, 10(5):673–685, 1998.

COCOS – Coordinated Communicating Sensors

Reinhardt Karnapke und Jörg Nolte

Brandenburgische Technische Universität Cottbus

Zusammenfassung – Das COCOS-Projekt wurde 2004 in das SPP 1140 aufgenommen und hat in der ersten Phase bis 2006 Middleware-Plattformen für homogene mobile Sensornetze auf der Basis von Lego RCX Robotern mit Funkkommunikation untersucht, prototypisch implementiert und evaluiert. Hauptziel war es, derartige Netze auf einem hohen Abstraktionsniveau programmierbar zu machen und den Anwendern neben verteilten Objekttechnologien in drahtlosen Sensornetzen datenparallele Programmierparadigmen an die Hand zu geben. Diese Arbeiten wurden im zweiten Förderzeitraum 2006 – 2008 auf gemischte MANET-/Sensornetzumgebungen ausgedehnt, in denen das MANET u.a. die Rolle eines mobilen Backbones für das Sensornetz übernehmen kann. Dieser Bericht gibt einen Überblick über die Arbeiten und wesentlichen Ergebnisse aus den beiden Förderperioden.

1. Einleitung

Große Sensornetze liefern eine Vielzahl von Rohdaten, die in der Regel nicht in der Rate aus dem Netz abgeführt werden können, in der relevante Meßwerte entstehen. Das Netz muß daher zumindest eine Vorverarbeitung leisten und liefert im Idealfalle anstelle individueller Meßwerte nur wenige aggregierte Ergebnisse über Kopfstationen an die Außenwelt. Wenn man die verteilten Meßwerte in einem Sensornetz als die Punkte eines Sensorbildes betrachtet (vergl. Bild 1), so haben typische Sensornetzalgorithmen zwangsläufig eine große Ähnlichkeit mit verteilten und parallelen Bildverarbeitunsgsalgorithmen, die auf einem Parallelrechner mit schmalbandiger Kommunikation und knappen Betriebsmitteln ausgeführt werden müssen. Da in Sensornetzen einzelne Knoten nur eine geringe Rolle spielen, sind Programmierparadigmen, die es erlauben, ganze Teilnetze als Einheit zu betrachten, besonders attraktiv. Die Entwicklung solcher Paradigmen stand im Mittelpunkt der ersten Förderungsphase, die es sich zum Ziel gesetzt hatte, Objekttechnologien und datenparallele Programmierparadigmen in die Welt der Sensornetze einzuführen. Hier wurde ferner von einem mobilen Szenario ausgegangen, in dem mit Sensorik bestückte Lego-Kleinstroboter nach Bedarf eine dynamische Relokation des Netzes ermöglichen.

In Zukunft werden mobile Einsatzkräfte, die sich z.B. in einem Rettungseinsatz befinden, auch auf Sensornetztechnologien zurückgreifen, um Gefahrensituatio-

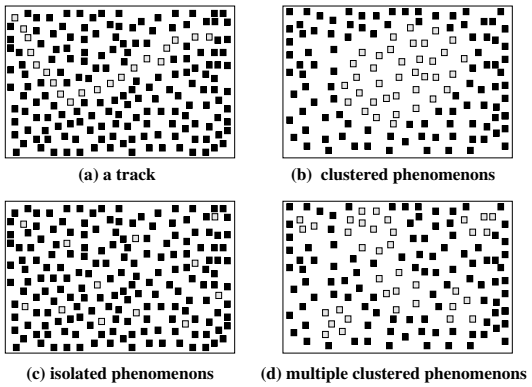


Abb. 1. Sensornetzbilder

nen besser begegnen zu können oder verletzte Personen aufzufinden. Dies kann sowohl über die Auswertung vorhandener Sensornetze geschehen, als auch mit Hilfe mobiler Sensorik. Hier kommen z.B. geeignete Kleinstroboter oder Rettungshunde, die mit Sensorknoten ausgestattet sind, in Betracht. Um solche Szenarien umzusetzen, ist es jedoch notwendig, Sensornetze und Mobile Ad-Hoc Netze (MANETs, z.B. auf WLAN-Basis) geeignet zu verbinden. Die Untersuchung solcher netzübergreifenden Middlewareschichten und Netzwerkprotokolle waren das Ziel der zweiten Projektphase, in der die in der ersten Phase untersuchten datenparallelen Programmierparadigmen auf heterogene Netze übertragen wurden und die MANET-Knoten die Rolle eines mobilen Backbone-Netzes für das Sensornetz übernommen haben.

2. Das COCOS Projekt

Das COCOS-Projekt wurde im Oktober 2004 als neues Projekt in das SPP 1140 aufgenommen. In der ersten Förderphase wurden datenparallele Programmierparadigmen für betriebsmittelarme *homogene* drahtlose Sensornetze untersucht und in einem objektorientierten Rahmenwerk umgesetzt. Das COCOS-Rahmenwerk wurde zunächst mit Hilfe eines generischen Simulators/Emulators (SERNet) [30] entwickelt und auf einer mobilen Sensornetzplattform bestehend aus Lego RCX-Robotern mit Radiokommunikation evaluiert.

Die erste Phase war stark durch ein Bottom-Up-Vorgehen bestimmt, da man im Bereich der eingebetteten Systeme und Sensornetze selten auf stabile Plattformen zurückgreifen kann und einfache Funknetze wie sie hier Verwendung finden bisweilen pathologische Eigenschaften aufweisen.

Zur Lösung dieser nicht unerheblichen technischen Probleme war es notwendig, sich mit Aspekten der Kommunikation von der MAC-Schicht bis hin zu kollektiven Operationen mit netzinterner Zwischenverarbeitung auf extrem betriebsmittelarmer Hardware auseinander zusetzen. Eine entsprechende Systemarchitektur war hier ein essentieller Schlüssel zur Problemlösung. Ferner war es eines der wesentlichen Ziele, nicht hinter den anerkannten Stand der Technik zurück zufallen und konsequent auf objektorientierte Programmiersprachen (C++) zurückzugreifen.

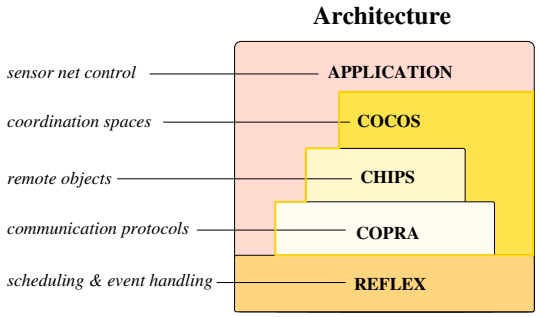


Abb. 2. Die COCOS Architektur

In Bild 2 sind die Komponenten des COCOS-Rahmenwerks mit ihren prinzipiellen Aufgaben dargestellt und werden nachfolgend erläutert.

2.1. Cocos

In COCOS werden die Sensoren und Aktuatoren eines Netzes als Instanzen von gewöhnlichen C++-Klassen repräsentiert. Diese Objekte können nach Belieben in verteilte Objekträume eingefügt und über kollektive Methodenaufrufe parallel manipuliert werden. Mit Hilfe solcher Operationen auf einem spezifischen Objektraum ist es z.B. möglich, viele verteilte Messdaten in einem einzigen parallelen Programmschritt im Netz zu aggregieren oder Aktuatoren wie die Motoren der RCX-Roboter kollektiv zu steuern und dadurch die Fahrtrichtung der Roboter zu ändern. Individuelle Objekte können ferner gleichzeitig Mitglieder mehrerer Objekträume sein. So kann ein spezifischer Temperatursensor z.B. Mitglied eines Objektraumes sein, in dem sich alle Temperatursensoren befinden. Der selbe Temperatursensor kann jedoch auch über einen anderen Objektraum erreichbar sein, in dem sich nur Sensoren befinden, deren Temperatur über einem gewissen Schwellwert liegt.

Es wurden diverse Objektraumkonzepte untersucht, implementiert und evaluiert [12] [9] [10]. Ein wesentliches Untersuchungskriterium war die topologische Stabilität der Objekträume in Abhängigkeit von den vorstellbaren Anwendungssze-

narien. Die topologische Stabilität eines Objektraumes ist im wesentlichen abhängig von der Stabilität der Mitgliedschaft (der Häufigkeit der Bei- und Austritte) und der Stabilität der Funknetztopologie. Wenn Objekträume z.B. verwendet werden, um alle Sensoren zusammenzufassen, die ein bestimmtes Phänomen detektiert haben (vergl. Bild 1), ist die Stabilität der Mitgliedschaft direkt abhängig von der Änderungsrate des jeweiligen physikalischen Phänomens.

Objekträume mit einer stabilen Mitgliederzahl *und* einer stabilen relativen geographischen Position der einzelnen Sensorknoten zueinander lassen demnach auch eine hohe topologische Stabilität der Objekträume erwarten, die wiederum sehr gut zur Optimierung kollektiver Operationen ausgenutzt werden kann.

Die Testplattform von COCOS ist jedoch ein *mobiles* Sensornetz, in dem alle Knoten potentiell nach Belieben ihre Position verändern können. Dennoch können auch hier Objekträume langlebigere Topologien besitzen, wenn das Sensornetz entweder in seiner *Gesamtheit* bewegt wird (z.B. die kontinuierliche Bewegung des Sensornetzes über ein größeres Areal), oder eine Applikation in dynamische Bewegungs- und längere stationäre Meßphasen unterteilt ist.

Eine wesentlich größere Bedeutung kommt in mobilen Sensornetzen jedoch Objekträumen ohne eine feste Topologie zu, da man in sehr dynamischen Anwendungen weder von einer stabilen Mitgliedschaft noch von einer stabilen Netzwerktopologie ausgehen kann. Hier wurden vorrangig adaptive Flutverfahren untersucht, bei denen trotz fehlender Topologieinformationen mit hoher Wahrscheinlichkeit im Zuge einer kollektiven Operationsausführung alle aktuellen Mitglieder erreicht werden können, ohne das Netz in seiner Gesamtheit zu fluten.

Die experimentellen Ergebnisse [10] legen dar, wie wichtig Experimente mit realen Sensornetzen sind, da die meisten Simulatoren wesentliche Aspekte der Kommunikation schlichtweg vernachlässigen. Während im Simulator alle Formen von Objekträumen zufriedenstellende Ergebnisse lieferten, waren in der Praxis nur die auf Fluten basierenden Objekträume sinnvoll nutzbar. Die Hauptursache für dieses Verhalten war ein eklatantes schichten übergreifendes Problem mit der verwendeten MAC-Schicht, die ineffektiv arbeitete, eine Nachrichtenübermittlung häufig verzögerte und de facto eine Abschätzung von Zeitschranken auf den höheren Schichten nicht ermöglichte. Daher war es essentiell sich auch intensiv mit MAC-Verfahren (vergl. Abschnitt 2.3.) auseinander zusetzen. Das aus dieser Forschung resultierende Medienzugriffsverfahren *MLMAC* [17] wurde mit einem Best Paper Award der Sensorcomm 2007 ausgezeichnet.

2.2. Chips

CHIPS (Convenient High-Level Invocation Protocol Suite) ist eine leichtgewichtige template-basierte RMI-Schicht, die für die höheren Schichten globale Objekt-

referenzen und entfernte Methodenaufrufe als Kommunikationsabstraktion zur Verfügung stellt [19]. Mit Hilfe der CHIPS-Ebene ist es möglich, komplexe Interaktionen innerhalb eines COCOS-Objektraumes auf relativ einfache Weise zu beschreiben, ohne manuell Nachrichtenformate für PDUs auf unterster Ebene festlegen zu müssen. So können verteilte Objekträume u.a. mit Hilfe verteilter dynamischer Datenstrukturen realisiert werden und es ist außerdem möglich, globale Objektreferenzen über Typparameter mit spezifischen Kommunikationsprotokollen zu assoziieren. Damit kann u.A. direkt Einfluss auf die Aufrufsemantik entfernter Methodenaufrufe genommen werden.

2.3. Copra

COPRA [7] (Communication Processing Architecture) ist ein Rahmenwerk zur Komposition von komplexeren Kommunikationsprotokollen aus einfachen Bausteinen. COPRA kommt insofern eine besondere Bedeutung im Rahmen des Gesamtprojektes zu, als Sensornetze aus Kostengründen häufig nur mit einfachsten Transceivern ausgerüstet sind. MAC- [17], Addressierungs- und Datensicherungsschichten [2] sind schlichtweg nicht vorhanden und die erzielbaren Bandbreiten sind sehr gering. Ohne geeignete Maßnahmen kann aufgrund von Kollisionen, stark benutzten öffentlichen Frequenzbändern und atmosphärischen Störungen der effektive Kanaldurchsatz leicht gegen Null gehen. COPRA stellt deshalb eine Reihe von elementaren Verarbeitungsstufen für die Kanalzugangskontrolle (CA, TDMA [15, 16, 18]), Datensicherung [1], Paketvermittlung für Multi-Hop-Kommunikation [20, 24, 31], und Nachrichtenaggregation und -kompression [23] zur Verfügung. Des weiteren wurden Methoden zur Aktualisierung über Funk [6, 22] und selbst stabilisierende Broadcastalgorithmen [29] realisiert. Diese einfacheren Stufen sind problemorientiert zu relativ komplexen Protokollmaschinen kombinierbar, die z.B. neben einer Multi-Hop Nachrichtenweiterleitung auch eine Datensicherung mit impliziten Hop-by-Hop-Bestätigungen erlauben. Ferner sind die MAC-Verfahren austauschbar und können durch zusätzliche Aggregations- und Kompressionsstufen ergänzt werden. So ist es in einem Multi-Hop-Netz mit TDMA-Zugangsverfahren möglich, durch Aggregation und Kompression die Sendefenster voll auszunutzen und in einem Kommunikationsschritt mehrere Pakete auf gemeinsamen Teilpfaden weiterzuleiten. Auf diese Weise wird die Bandbreite erheblich effektiver ausgenutzt und gleichzeitig der Energieverbrauch minimiert.

2.4. Reflex

Obwohl sich das COCOS-Projekt vordringliche mit Middleware befasst hat, war es aufgrund des Mangels geeigneter Betriebs- und Laufzeitplattformen nötig, sich

auch der unteren Schichten anzunehmen, bis hin zur Verwendung eines eigenen Betriebssystems.

REFLEX [32] [33] [34] ist ein ereignisgetriebenes Betriebssystem für eingebettete Systeme und Sensorknoten. Das Basissystem wurde unabhängig vom COCOS-Projekt am Lehrstuhl entwickelt und fungiert als eingebettete Betriebssystemplattform für das Projekt. Eine Portierung auf die im Projekt verwendeten RCX-Roboter wurde bereits frühzeitig vorgenommen. Im Rahmen dieser Portierung wurden auch die notwendigen Gerätetreiber für den RCX entwickelt. Da REFLEX fast ausschließlich in C++ implementiert ist, gestaltet sich die Portierung des Basissystems auf neue Plattformen problemlos und wurde im Laufe der zweiten Förderungsphase mehrfach, z.B. für TMoteSky Sensorknoten, durchgeführt.

2.5. Demonstratoren

Die Eigenschaften der COCOS-Plattform wurden anhand eines Roboterballets und eines Meßszenarios demonstriert. Lego RCX Roboter sind normalerweise nicht für die Radiokommunikation vorbereitet. Es wurden daher insgesamt 14 Lego Roboter durch die Herausführung einer seriellen Schnittstelle technisch leicht verändert und mit Funkmodulen (433MHz Easy-Radio) versehen. Sie wurden in den beiden Demonstratoren der ersten Förderphase (Roboterballett und Meßszenario) erfolgreich eingesetzt.

Im Roboterballett wurden die Bewegungsphasen von 8 Robotern vom anführenden neunten Roboter kollektiv kontrolliert. Im Meßszenario bewegten sich alle Roboter in einer zufälligen Richtung mit zufälligen Richtungsänderungen. Nach einer gewissen Zeit wurde die Bewegungsphase von einem Roboter beendet, welcher ab diesem Zeitpunkt alle anderen mittels kollektiver Operationen fernsteuerte. Alle Roboter ermittelten die Werte ihrer Lichtsensoren. Anschließend wurde der Roboter mit dem höchsten Wert ermittelt, welcher von diesem Zeitpunkt an die Steuerung übernahm und alle Roboter wieder in die Bewegungsphase überführte.

2.6. Kollektive Operationen für MANETs

Drahtlose Sensornetze (WSN) und allgemeine Mobile ad-hoc Netze (MANETs) wurden in der wissenschaftlichen Welt bisher weitgehend getrennt voneinander betrachtet. Dies ist nicht zuletzt auf unterschiedliche Zielsetzungen und die daraus resultierenden Hardwareplattformen zurückzuführen. Seit den Terroranschlägen auf das World Trade Center in den USA und den diversen Umweltkatastrophen der letzten Jahre hat sich diese Situation jedoch grundlegend geändert. So wird z.B. davon ausgegangen, dass Feuerwehrleute, die einen brennenden Tunnel betreten, Sensorik mitführen [8], welche zusätzliche Daten an die Einsatzleitung überträgt.

Andere Szenarien gehen davon aus, dass Helfer bei Einsätzen in Gebäuden die Orientierung verlieren können und stellen deshalb Algorithmen zur Positionsbestimmung zur Verfügung [14]. Ebenso ist es möglich, bereits vorhandene oder vor Ort ausgebrachte Sensorik zu nutzen, um den Einsatzkräften einen Überblick über die Gesamtsituation zu verschaffen. Dabei gewinnt eine roboterbasierte mobile Sensorik wie sie auch im COCOS-Projekt zum Einsatz kommt, zunehmend an Bedeutung [3] [4].

In allen diesen Szenarien können sich MANETs und WSN auf ideale Weise ergänzen. MANET-Knoten können die von den WSN vorverarbeiteten aggregierten Umweltdaten effektiv weiterverarbeiten und ein einfaches WSN durch komplexe Sensorik und Dienste erweitern. So entsteht ein heterogenes Netz mit unterschiedlich leistungsfähigen Knoten (vergl. Bild 3), dessen Verarbeitungs- und Kommunikationskapazität weit über die Möglichkeiten eines einfachen Sensornetzes hinausgehen [21]. Damit ist es grundsätzlich möglich, Messdaten aus lokal zur Unterstützung ausgebrachten WSN weiträumig zu analysieren und kooperierenden menschlichen Einsatzgruppen, die ein PDA-gestütztes WLAN-MANET bilden, eine globale Übersicht über das Einsatzgebiet mit seinen lokalen Gefahren zu liefern [14].

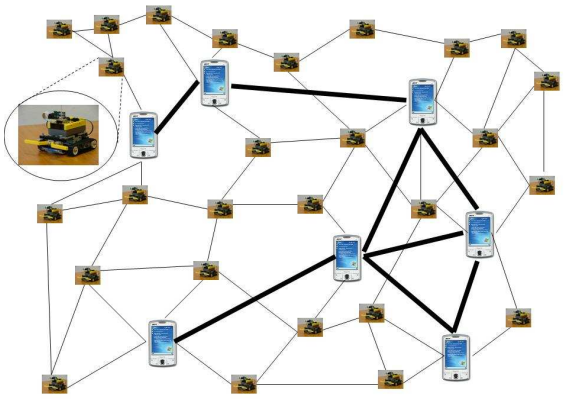


Abb. 3. Ein kombiniertes WSN/MANET

Eine adäquate Umsetzung dieses Szenarios erfordert jedoch Middleware-Plattformen, die MANETs und WSNs erheblich enger miteinander koppeln, als es auf einer klassischen Netzwerkebene möglich wäre.

In dieser Phase des Projektes wurde die COCOS-Sensornetzplattform mit ihren datenparallelen Programmierparadigmen auf heterogene MANETs ausgedehnt und in dort verbreitete Middleware-Plattformen integriert. Aus Anwendungssicht stellt

sich das kombinierte Netz nun als ein drahtloser Parallelrechner mit mobilen Knoten dar.

Bild 4 zeigt eine mögliche Anwendung der kollektiven Operationen von COCOS. In diesem Szenario sind Feuerwehrleute in das Innere eines brennenden Gebäudes vorgedrungen und ermitteln nun unter Verwendung der ausgebrachten Sensorknoten den sichersten Weg aus dem Gebäude.

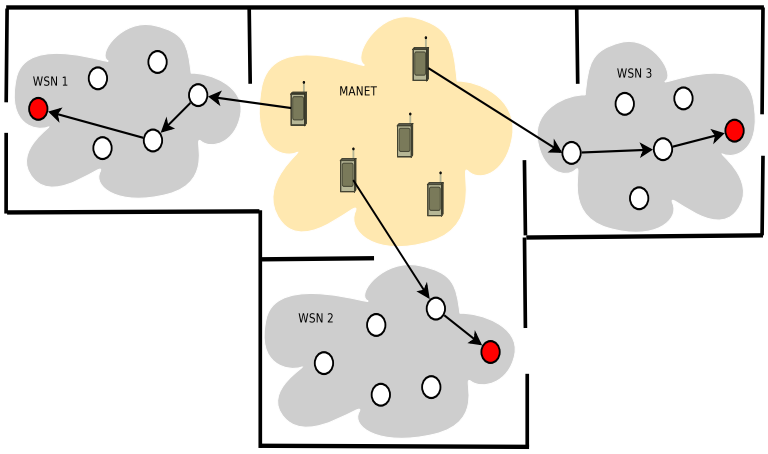


Abb. 4. Suche nach sicheren Ausgängen

Andere Umsetzungen, welche auf einem ähnlichen Szenario aufsetzen, gehen verschiedene Wege. So werden z.B. die Lebenszeichen der Feuerwehrleute an die Leitstelle übertragen [8], in anderen Fällen, z.B. im Siren-Projekt, werden hingegen Daten zwischen den einzelnen Feuerwehrleuten ausgetauscht [5]. Im COCOS-Projekt wurde der zweite Weg gewählt, jedoch mit einem wesentlichen Unterschied: Bei Siren werden nur Daten ausgetauscht, welche die Feuerwehrleute direkt gemessen haben. Die COCOS-Plattform erlaubt es jedoch, Informationen auch von weit entfernten Sensoren abzufragen sofern zu diesen überhaupt ein Kommunikationspfad besteht.

Um dieses Ziel zu erreichen, wurde zunächst eine COCOS-Plattform für WLAN vernetzte PDAs/Laptops realisiert und in das .Net Compact Framework integriert [28]. Die Umsetzung wurde sowohl unter Linux (Mono) als auch unter Windows (Microsoft Compact Framework) evaluiert. Dadurch wurde es möglich, aus C# heraus [11] Objekte in verteilten Objekträumen mit Hilfe kollektiver Operationen parallel zu manipulieren. Diese Projektphase wurde mit einer Demonstration der erreichten Funktionalität (kollektive Operationen im MANET) abgeschlossen.

2.7. Ein netzübergreifendes Routingprotokoll

Nachdem in der ersten Förderungsphase bereits kollektive Operationen für *homogene* Sensornetze realisiert wurden, war es nahe liegend, die Kommunikation zwischen MANET und WSN durch die Entwicklung eines netzübergreifenden Routingprotokolls zu realisieren. Dabei standen verschiedene Varianten zur Auswahl. Gateway-Lösungen, in denen einzelne, ausgezeichnete Knoten die Verbindung zwischen den unterschiedlichen Netzen darstellen kamen nicht in Frage, da es jedem MANET-Knoten möglich sein sollte, direkt mit benachbarten Sensorknoten zu kommunizieren. Aus diesem Grunde wurde ein einheitliches Adressierungsschema gewählt, welches es zusätzlich ermöglicht, auch mit *heterogenen* Sensornetzen zu kommunizieren. Das resultierende Routingprotokoll [25] [26] verwendet eine gewichtete Routingmetrik, welche die Hauptlast der Kommunikation durch das MANET leitet, um die Lebensdauer des Sensornetzes zu erhöhen (Bild 5).

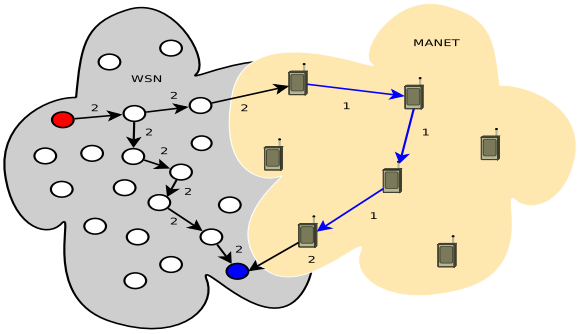


Abb. 5. Eine gewichtete Routingmetrik

2.8. Erweiterung des OMNET++ Simulators

Um die Entwicklung der Protokolle zu vereinfachen und Ergebnisse für große Netze zu erhalten wurden sie vor den Experimenten auf der realen Hardware in OMNET++ [27] simuliert. Dafür waren eine Reihe von Anpassungen notwendig. So wurde z.B. das CScharpSimpleModule [13] entwickelt und auf der OMNET++ Website verfügbar gemacht. Diese Erweiterung erlaubt es, C#-Programmcode in OMNET++ zu simulieren. Zusätzlich wurde Unterstützung für das REFLEX-Betriebssystem integriert, welche es ermöglicht, den Programmcode mit minimalen Änderungen sowohl im Simulator als auch auf der realen Hardware zu verwenden. Zur erfolgreichen Evaluierung des gewählten Einsatzszenarios war es schließlich noch

nötig, den Einsatz mehrerer Funkmodule (z.B. IEEE 802.11 für die Kommunikation im MANET und IEEE 802.15.4 für das WSN) in einem Knoten zu ermöglichen.

3. Zusammenfassung

Das COCOS-Projekt hat im Rahmen des SPP 1140 Middleware-Plattformen für homogene Sensornetze und gemischte MANET-/Sensornetzumgebungen untersucht, prototypisch implementiert und evaluiert. Hauptziel war es, derartige Netze auf einem hohen Abstraktionsniveau programmierbar zu machen und den Anwendern neben verteilten Objekttechnologien in drahtlosen Netzen datenparallele Programmierparadigmen an die Hand zu geben, die entfernte Methodenaufrufe auf verteilte Objektgruppen ausdehnen und so kollektive Auswertungen kompletter Teilnetze erlauben. In diesem Zusammenhang wurden sowohl Objektgruppen mit statischen Topologien als auch mit auf Fluttechniken beruhenden dynamischen Topologien untersucht. Während die Simulationsergebnisse für alle Topologien zufriedenstellend waren, haben sich in Experimenten mit realen Sensornetzen die dynamischen Topologien auch in Szenarien ohne Mobilität als überlegen herausgestellt. Dies ist auf die Tatsache zurückzuführen, dass die meisten vorhandenen Simulationsmodelle absolut unzureichend sind und zu viele Aspekte der realen Kommunikation vernachlässigen. COCOS hat sich daher von Anfang an zum Ziel gesetzt, nicht primär auf Simulationsergebnisse zu setzen, sondern frühzeitig auch reale Feldexperimente durchzuführen. Diese Entscheidung hatte jedoch die Überwindung einer Reihe von nicht unerheblichen technischen Hürden zur Folge, die von der prinzipiell problematischen Entwicklung von Software für betriebsmittelarme tief eingebettete Systeme, über eine geeignete eingebettete Betriebssystem-Plattform bis hin zur Entwicklung eigener Datensicherungs- und MAC-Protokolle reichte. Abschließend ist zu sagen, dass gerade die unteren Kommunikationsschichten bei einfachster Kommunikationshardware inhärent unzuverlässig sind. Ein wesentlicher Teil der Projektarbeit hat sich daher mit schichten übergreifenden Kommunikationsprotokollen und deren Integration in ein geeignetes softwaretechnisches Rahmenwerk auseinandergesetzt, das eine Komposition von problemspezifischen Protokollmaschinen ermöglicht.

Literaturverzeichnis

[1] BRZOZOWSKI, MARCIN: *Eine Datensicherungs- und Transportschicht für drahtlose Netze*. Masterarbeit. Brandenburgische Technische Universität, Cottbus, Mar 2006.

[2] BRZOZOWSKI, MARCIN, REINHARDT KARNAPKE und JÖRG NOLTE: *IMPACT - A Family of Cross-Layer Transmission Protocols for Wireless Sensor Networks*. In: *The First Inter-*

national Workshop on Research Challenges in Next Generation Networks for First Responders and Critical Infrastructures (NetCri 07), in conjunction with 26th IEEE IPCCC, 2007.

[3] CRANDALL, JACOB W., CURTIS W. NIELSEN und MICHAEL A. GOODRICH: *Towards predicting robot team performance*. SMC, 2003.

[4] FONG, TERRENCE, CHARLES THORPE und CHARLES BAUR: *Multi-robot remote driving with collaborative control*.

[5] JIANG, X., N. CHEN, K. WANG, L. TAKAYAMA und J. LANDAY: *Siren: Context-aware Computing for Firefighting*.

[6] JÜTTNER, STEVE: *Ein inkrementelles Aktualisierungsverfahren für tief eingebettete Systeme*. Bachelorarbeit. Brandenburgische Technische Universität, Cottbus, 2007.

[7] KARNAPKE, REINHARDT und JÖRG NOLTE: *COPRA - A COmmunication PRocessing Architecture for Wireless Sensor Networks*. In: *Euro-Par 2006 Parallel Processing*, Seiten 951–960. Springer, 2006.

[8] KOUMPIS, KONSTANTINOS, LESLEY HANNA und STEPHEN HAILES: *Tunnels of Terror*. IEE Computing and Control Engineering Magazine, Dec/Jan 2005/06.

[9] KRÜGER, MAIK, REINHARDT KARNAPKE und JÖRG NOLTE: *Controlling sensors and actuators collectively using the COCOS-framework*. In: *SANET '07: Proceedings of the First ACM workshop on Sensor and actor networks*, Seiten 53–54, New York, NY, USA, 2007. ACM Press.

[10] KRÜGER, MAIK, REINHARDT KARNAPKE und JÖRG NOLTE: *In-network Processing and Collective Operations using the COCOS-Framework*. In: *12th IEEE Conference on Emerging Technologies and Factory Automation*, 2007.

[11] KRÜGER, FELIX: *Spracheinbettung entfernter Methodenaufrufe und kollektiver Operationen für drahtlose heterogene Netze*. Studienarbeit. Brandenburgische Technische Universität, Cottbus, 2007.

[12] KRÜGER, MAIK: *Entwicklung und Implementierung von selbst organisierenden verteilten Objektgruppen für Sensornetze und Mobile Ad-hoc-Netzwerke*. Diplomarbeit. Brandenburgische Technische Universität, Cottbus, 2006.

[13] LAGEMANN, ANDREAS und JÖRG NOLTE: *CsharpSimpleModule – writing OmNet++ Modules with C# and Mono*. In: *OMNeT++ 2008*, Marseille, March 2008.

[14] LORINCZ, KONRAD, DAVID J. MALAN, THADDEUS R.F. FULFORD-JONES, ALAN NAWOJ, ANTONY CLAVEL, VICTOR SHNAYDER, GEOFFREY MAINLAND, MATT WELSH und STEVE MOULTON: *Sensor Networks for Emergency Response: Challenges and Opportunities*. IEEE Pervasive Computing, vol. 03, no. 4, pp. 16-23, Oct-Dec 2004.

[15] MANK, STEPHAN: *Auswahl und Implementierung geeigneter Mac-Verfahren für Sensornetze*. Studienarbeit. Brandenburgische Technische Universität, Cottbus, 2006.

[16] MANK, STEPHAN: *Medienzugriffsverfahren für mobile drahtlose Sensornetze mit unidirektionalen Verbindungen*. Diplomarbeit. Brandenburgische Technische Universität, Cottbus, 2008.

[17] MANK, STEPHAN, REINHARDT KARNAPKE und JÖRG NOLTE: *An adaptive TDMA based MAC Protocol for Mobile Wireless Sensor Networks, Best Paper Award*. In: *International Conference on Sensor Technologies and Applications*, 2007.

[18] MANK, STEPHAN, REINHARDT KARNAPKE und JÖRG NOLTE: *MLMAC - An adaptive*

TDMA MAC Protocol for Mobile Wireless Sensor Networks. In: *Ad-Hoc & Sensor Wireless Networks: An International Journal, Special Issue on 1st International Conference on Sensor Technologies and Applications*, 2008.

[19] NEUMANN, JENS: *Entwicklung eines RMI Layers für heterogene drahtlose Sensornetzwerke*. Masterarbeit. Brandenburgische Technische Universität, Cottbus, 2006.

[20] PETERS, DAVID: *Buckshot-Routing: Ein robustes Source Routing Protokoll für dichte Ad hoc Netze*. Diplomarbeit. Brandenburgische Technische Universität, Cottbus, 2008.

[21] RÖMER, KAY, OLIVER KASTEN und FRIEDEMANN MATTERN: *Middleware challenges for wireless sensor networks*. SIGMOBILE Mob. Comput. Commun. Rev., 6(4):59–61, 2002.

[22] SCHULZE, CARSTEN: *Drahtlose Systemaktualisierung für Sensornetzwerke*. Studienarbeit. Brandenburgische Technische Universität, Cottbus, 2007.

[23] SCHULZE, STEFFEN: *Kompressions- und Aggregationsverfahren für Nachrichten in Drahtlosen Netzwerken*. Bachelorarbeit. Brandenburgische Technische Universität, Cottbus, 2006.

[24] SENNER, TOBIAS: *Auswahl und Implementierung geeigneter Routingverfahren für Sensornetze*. Studienarbeit. Brandenburgische Technische Universität, Cottbus, 2006.

[25] SENNER, TOBIAS: *Routingverfahren für drahtlose heterogene Netze*. Diplomarbeit. Brandenburgische Technische Universität, Cottbus, 2007.

[26] SENNER, TOBIAS, REINHARDT KARNAPKE und JÖRG NOLTE: *A Combined Routing Layer for Wireless Sensor Networks and Mobile Ad-Hoc Networks*. In: *Proceedings of International Conference on Sensor Technologies and Applications*, 2008.

[27] VARGA, ANDRÁS: *The OMNeT++ Discrete Event Simulation System*. In: *Proceedings of the European Simulation Multiconference (ESM'2001)*, Prague, Czech Republic, Juni 2001.

[28] VÖLKEL, FRANK: *Kollektive Anwendungsmechanismen für heterogene drahtlose Netze*. Studienarbeit. Brandenburgische Technische Universität, Cottbus, 2007.

[29] VÖLKEL, FRANK: *Selbststabilisierende Broadcastalgorithmen für drahtlose Sensornetze*. Diplomarbeit. Brandenburgische Technische Universität, Cottbus, 2008.

[30] WALLAT, ANDRÉ: *SERNet - Eine generische Emulationsplatform für eingebettete Systeme mit drahtloser Kommunikation*. Diplomarbeit. Brandenburgische Technische Universität, Cottbus, Mar 2005.

[31] WALTER, JÜRGEN: *Virtual Geographic Tree Routing (VGTR) Ein Leichtgewichtiges Routing Protokoll für mobile Ad-Hoc Netze*. Studienarbeit. Brandenburgische Technische Universität, Cottbus, 2007.

[32] WALTHER, KARSTEN, REINHARD HEMMERLING und JÖRG NOLTE: *Generic Trigger Variables and Event Flow Wrappers in Reflex*. In: *ECOOP - Workshop on Programming Languages and Operating Systems*, Jun 2004.

[33] WALTHER, KARSTEN, REINHARDT KARNAPKE und JÖRG NOLTE: *An existing complete House Control System based on the REFLEX Operating System: Implementation and Experiences over a Period of 4 Years*. In: *Proceedings of 13th IEEE Conference on Emerging Technologies and Factory Automation*, 2008.

[34] WALTHER, KARSTEN und JÖRG NOLTE: *Event-Flow and Synchronization in Single Threaded Systems*. In: *Proceedings of First GI/ITG Workshop on Non-Functional Properties of Embedded Systems (NFPES)*, Mar 2006.

Middleware für mobile spontan vernetzte Sensornetzwerke

Jan Blumenthal und Dirk Timmermann

Universität Rostock
Fakultät für Informatik und Elektrotechnik
Institut für Angewandte Mikroelektronik und Datentechnik

Zusammenfassung – Das Projekt „Middleware für mobile spontan vernetzte Sensornetzwerke“ der Universität Rostock ist ein Teilprojekt des DFG Schwerpunktprogrammes „Basissoftware für Selbstorganisierende Infrastrukturen für Vernetzte Mobile Systeme“ (SPP 1140) und demonstriert, dass für Sensornetzwerke vor allem Forschungsbedarf hinsichtlich der Ressourcenknappheit, insbesondere bei geringen Energiereserven, bestehen. Es erforschte und entwickelte energieeffiziente Verfahren zur Lokalisierung, Gruppenbildung und Administrierung von Sensornetzwerken und bildet somit eine solide Basis für statische und schwach dynamische Netzwerke.

1. Ausgangslage und Motivation

Zu Beginn des Projektes waren in der Wissenschaftsgemeinschaft schon zahlreiche einfache Verfahren (z.B. Lokalisierung, Datenaggregation, Betriebssysteme) für Sensornetzwerke bekannt. Jedoch wurde selten der Ressourcenverbrauch der Verfahren beachtet, hervorgerufen z.B. durch Senden & Empfangen von Nachrichten, Speichern von Daten, Polling, Prozessorbelastung durch Berechnungen, Verzicht auf Low-Power-Modi. Eine Vielzahl der publizierten Verfahren setzten zentrale Basiseinheiten zur Messdatenanalyse voraus, wodurch i.d.R. ein immenser Datenverkehr im direkten Umfeld sowohl zu dieser Station hin als auch zurück ins Netzwerk produziert wird. Dies führt im realen Netzwerk zwangsläufig zum Ausfall von autarken Sensorknoten in diesem Bereich, da die Energiereserven relativ schnell erschöpft sind.

Höherwertige Middleware- und Dienstarchitekturen waren vor allem auf ad hoc Netzwerke und ressourcenstarke, verteilte Netzwerke fokussiert. Der Trend bestand darin, die existierenden Technologien aus dem Desktopbereich weitgehend auf ad hoc- und Sensornetzwerke zu übertragen.

Das Projekt hatte zum Ziel, die besonderen Einschränkungen und Randbedingungen bei der Entwicklung von Basissoftware für ressourcenarme Sensornetzwerke zu analysieren und geeignete Verfahren zu entwickeln, welche die oben genannten Randbedingungen berücksichtigen. Insbesondere war zu untersuchen,

wie höherwertige Aufgaben und Dienste trotz spontaner Vernetzung über drahtlose Verbindungen zwischen einer Vielzahl von ressourcenarmen Sensorknoten umgesetzt werden können [4, 23].

Die Meilensteine im Projektzeitraum waren:

- Konzeption von Middleware- und Dienstarchitekturen für ressourcenlimitierte Sensornetzwerke,
- Entwicklung von lokationsabhängigen Diensten, die durch die Kombination von ermittelten Positionen und aufgenommenen Messdaten ermöglicht werden, und dem Netzwerk eine höherwertige Information bereitstellen,
- Dezentrale Analyse von Phänomenen in Netzwerken auf Basis lokationsabhängiger Dienste auf den Sensorknoten,
- Prototypische Realisierung eines Sensornetzwerkes.

Die anvisierten Zielvorgaben der Sensorknoten (Volumen von ca. 1^3 mm, ideale Übertragungstechniken) konnten durch die zur Verfügung stehenden Sensorknoten (MicaMote, CC1010 u.a.) nicht ansatzweise erreicht werden, sodass reale Feldtests für diese Art der Sensorknoten bislang nicht durchgeführt werden konnten [10].

2. Das Projekt im Überblick

Als Basis für weitere Arbeiten wurden zuerst grundlegende Untersuchungen zu den benötigten Kommunikations- und Middlewarefunktionalitäten für Sensornetzwerke (SNW) unter dem Fokus durchgeführt, dass wirtschaftlich sinnvolle Sensornetzwerke, die aus tausenden winzigen Sensorknoten bestehen, geringste Kosten pro Sensorknoten erfordern. Sensorknoten zeichnen sich daher durch knappste Ressourcen bezüglich Speicher, Rechenfähigkeit und Größe aus. Zusätzlich wurde untersucht, inwieweit sich durch die Vernetzung von Knoten neben den anwendungsspezifischen Anforderungen eines Knotens auch dynamische Systemanforderungen an das Netzwerk ergeben. Das Primärziel bei der Programmierung der Software verschiebt sich somit vom herkömmlichen Einzelergebnis einer Sensorknotenapplikation zur Gesamtapplikation eines Netzwerkes, d.h. die Kooperation der Knoten ist erfolgsentscheidend [4].

Die knappen Ressourcen der Sensorknoten bedingen zusätzlich optimierte, kleine und schnelle sowie auf die Bedürfnisse des jeweiligen Sensornetzwerkes angepasste Softwarelösungen. Es war folglich zu untersuchen, inwieweit sich die gewünschte Heterogenität der bekannten Service-Architekturen in der Ausführungsebene (Executable) auf die Quellcode- bzw. Beschreibungsebene auswirken wird. Aufgrund der geringen Speichergröße und der minimalistischen Betriebssysteme für Sensornetzwerke sind Architekturen mit dynamischen Datenstrukturen zur Laufzeit unangebracht. Jedoch ist eine Aktualisierung der Software zur

Laufzeit im Ausbringungsgebiet, also eine Dynamik in der Programmausführung, zwingend erforderlich, um die Kosten zu vermeiden, die durch Einsammeln und erneutes Ausbringen der Sensorknoten verursacht werden. Dieser scheinbare Widerspruch aus statischen Datenstrukturen und dynamischen Programmstrukturen wurde eingehend untersucht [2].

Eine besondere Herausforderung in Sensornetzwerken stellte die Wahl eines geeigneten Übertragungsverfahrens dar, das Broadcasts und Multihop-Übertragungen, eine kleine Bauform, Energieeffizienz und einen geringen Preis vereinen muss [10].

Um die entwickelten Verfahren und Algorithmen mangels geeigneter Hardware zu verifizieren, wurden umfangreiche Simulationen mit den Softwarewerkzeugen OmNET++, ns-2 und SeNeTs durchgeführt. Dabei wurde vor allem deutlich, dass die sequentielle Simulation eines verteilten Netzwerkes aus autarken Knoten zahlreiche Zeit- und Synchronisationsprobleme birgt, die im realen System so nicht existieren [11].

Das Anbieten von lokationsabhängigen Diensten in Sensornetzwerken erfordert die Bestimmung der Position der zufällig verteilten Sensorknoten. Die Positionsbestimmung erforderte zunächst eine umfangreiche Analyse der bereits publizierten Verfahren. Die meisten Algorithmen gehen wie bei der statischen Positionierung von einer Aufgabenteilung zwischen Beacons (Position bekannt) und Sensorknoten (Position unbekannt) im Netzwerk aus. Für die lokale Positionsbestimmung ist i.d.R. eine regional begrenzte Kommunikation mit Nachbarknoten bzw -beacons nötig, wodurch ein signifikanter Energieverbrauch zu verzeichnen ist. Die Ursachen und Zusammenhänge des Energieverbrauches, des Kommunikationsaufwandes und der Positionsgenauigkeit wurden eingehend untersucht und führten schließlich zu deutlichen Energiereduzierung und Genauigkeitsverbesserung bei geeigneter Kalibrierung des Systems [6, 19]. Die Forschungsarbeiten im Bereich der mobilen Positionierung von Sensorknoten haben in den letzten Jahren stark zugenommen. Gerade die mobile Positionierung erfordert periodisch Ressourcen, die nicht zu vernachlässigen sind. Die untersuchten Algorithmen wurden dahingehend untersucht und optimiert, inwieweit die algorithmische Komplexität der Verfahren einen Einsatz in Sensornetzwerken zulässt bzw. reduziert werden kann [18].

Lokationsabhängige Dienste bieten entweder ein lokationsabhängiges Einzelergebnis oder ein durch Kooperation mehrerer Knoten ermitteltes Gesamtergebnis an. Die an der Kooperation beteiligten Sensorknoten bilden ein Gebiet (GeoCast-Zone), das von dem Dienst ermittelt werden muss. Ausgehend von den hohen Anforderungen an Software in Sensornetzwerken entwickelten wir ein Verfahren zur ressourcenarmen und dezentralen Detektion und Beschreibung von GeoCast-Zonen. Diese Verfahren bildeten die Voraussetzung zu Untersuchungen der dezentralen Phänomendetektion in mobilen Sensornetzwerken [12, 16].

Für die Administrierung des Netzwerkes, die Auswertung und Visualisierung der Mess- und Kooperationsergebnisse war die Entwicklung einer geeigneten Software erforderlich, da die bekannten Lösungen zumeist proprietär waren und den limitierten Ressourcenanforderungen nicht genügten [5].

Zum Abschluss des Projektes wurden die entwickelten Algorithmen und Verfahren in einem Demonstrator in Kooperation mit dem Teilprojekt SWARMS dieses Schwerpunktprogrammes präsentiert [13].

3. Erzielte Ergebnisse

Die vorgeschlagenen Softwarearchitekturen und entwickelten Algorithmen stellen eine solide Basis für die Entwicklung von Anwendungen für ressourcenarme Sensornetzwerke dar. Sie ermöglichen vor allem den Einsatz in Systemen mit eingeschränkten Energiereserven und garantieren gleichzeitig ihre korrekte Funktion. Die wesentlichen Beiträge dieses Forschungsprojektes sind:

- Definition der Randbedingungen in ressourcenarmen Sensornetzwerken,
- Design von Middleware- und Dienstarchitekturen,
- Entwurf von Strategien zur Administrierung und Konfiguration eines Sensornetzwerkes,
- Entwicklung von Algorithmen zur dezentralen Lokalisierung von ressourcenarmen Sensorknoten,
- Autarke Gruppenbildung in Sensornetzwerken.

Die Ergebnisse der theoretischen Forschungen wurden abschließend in einem Demonstrator, der in Kooperation mit dem Teilprojekt SWARMS des Schwerpunktprogramms SPP 1140 erstellt wurde, in der Praxis erfolgreich überprüft [13]. Parallel dazu erfolgten weitere praktischen Umsetzungen, z.B. die Entwicklung eines prototypischen Sensorknotens zur Überwachung der Feuchtigkeit in Deichen während eines Hochwassers [21].

Im Folgenden werden die einzelnen Beiträge im Detail dargestellt.

3.1. Definition der Randbedingungen in ressourcenarmen Netzwerken

Aus der umfangreichen Analyse ergaben sich für ressourcenarme Sensornetzwerke folgende Anforderungen an den Entwurfs- und Implementierungsprozess von Software für Sensornetzwerke [4]:

1. Sensornetzwerke erfordern eine **Selbstorganisation** der Sensorknoten.
2. Alle Sensorknoten übernehmen Funktionen zur **Aufrechterhaltung des Netzwerkes**.

3. Durch eine **kooperative Bearbeitung** von Aufgaben wird eine höhere Ergebnisqualität erreicht.
4. Sensornetzwerke benötigen geeignete **Evolutionsstrategien**, um Ausfälle zu kompensieren und Softwareaktualisierungen zu ermöglichen.
5. Die **Energieeffizienz** von Mechanismen, Algorithmen und Protokollen ist entscheidend für die Lebensdauer des Netzwerkes.
6. **Adaptive Dienst- und Proxy-Architekturen** ermöglichen die Kommunikation mit externen Netzwerken.

Die knappste Ressource innerhalb eines Netzwerkes ist die zur Verfügung stehende Energie. Es ist demnach unerlässlich, neben stromsparender Hardware auch energieeffiziente Algorithmen und Software einzusetzen. Es sind daher Software-Architekturen erforderlich, die den Energiebedarf der Hardware senken, z.B. durch geringen RAM-Verbrauch, kurze Laufzeiten von Programmen oder minimale Kommunikation. Folglich müssen die zu entwickelnden Dienst- und Proxy-Architekturen neben der Funktion auch den Energieverbrauch berücksichtigen.

3.2. Design von Middleware- und Dienstarchitekturen

Für Sensornetzwerke angepasste Middlewaretechnologien lassen sich in zwei grundlegende Aspekte teilen, den Programmieraspekt (Programming Aspect) und den Verhaltensaspekt (Behavior Aspect) [4, 15]. Wir untersuchten die aktuell in der Diskussion stehenden Middleware-Technologien und stellten ein eigenes Konzept [14] mit Ansätzen zur automatischen Generierung von Sensornetzwerkanwendungen vor [3]. In unseren Untersuchungen stellten wir fest, dass die meisten Middleware-Technologien meist nur einen Aspekt behandeln.

Wir untersuchten die aus dem Desktopbereich bekannten Architekturen wie *Universal Plug-n-Play* (UPnP), Web Services basierend auf *Simple Object Access Protocol* (SOAP) oder *Device Profile for Web Services* (DPWS). Unsere Analyse ergab, dass sich diese Architekturen nicht für den direkten Einsatz in Sensornetzwerken eignen, da sie unter zu Sensornetzwerken konträren Designzielen entworfen wurden und wichtigen Kriterien wie der notwendigen Minimierung des Kommunikations- und Ressourcenaufwandes nicht entsprechen. Stattdessen konzentrieren sich die Systeme auf die Kapselung der Heterogenität der Systeme. Ein Lokationsbezug zu angebotenen Diensten erfolgt nur über Kontextinformationen in den Anwendungen. Zudem setzen diese Architekturen voll funktionsfähige Netzwerkstacks voraus, d.h. Segmentierung, Routing und Retransmission von Nachrichten, die für Sensornetzwerke nicht zwangsläufig gegeben oder völlig überdimensioniert sind.

Die Entwicklung einer Service-Architektur für die von uns verwendeten Chipcon-Sensorknoten machte es erforderlich, zunächst die dynamische Rekonfigurierbarkeit von Anwendungen per Funkübertragung (OTA: Over-the-Air) sicherzu-

stellen. Dazu entwickelten wir ein OTA-BIOS, das auf jedem Sensorknoten fest installiert und nicht verändert werden kann [22]. Dieses BIOS übernimmt rudimentäre Betriebssystem-Funktionen wie Initialisierung der Hardware (Timer, Interrupts), Übertragung von Nachrichten und das Neuprogrammieren (Flashen) von Anwendungen. Alle anderen Softwarekomponenten (Middleware, Dienste, Anwendungen) können zur Laufzeit ausgetauscht werden. Dazu wurde ein Übertragungsprotokoll entwickelt, das die korrekte Übertragung und Initialisierung der Software auf den Sensorknoten sicherstellt [2]. Eine besondere Herausforderung bestand in der Symbiose aus der Entwicklung einer geeigneten Speicherorganisation für die Harvard-Architektur des CC1010, die in Übereinstimmung mit dem allgemeinen Softwaremodell eines Sensorknotens gebracht werden musste. Dies beinhaltet die Definition zusätzlicher Interrupts in der Interruptservicetabelle, die Entkopplung von Anwendungen und des OTA-BIOS durch Sprung-Tabellen, die Beherrschung verschiedener Speichertypen (SRAM, XRAM), die Vermeidung von Stackoperationen sowie von direkter Adressierung, die Nachrichtenvalidierung durch CRC-Überprüfung u.v.m. unter den Gesichtspunkten der Ressourcenarmut und der Robustheit.

Das vorgestellte OTA-BIOS ist die Voraussetzung für die Entwicklung unserer mobilen ressourcen-optimierten Service-Architektur (RASA) [7]. Die Service-Architektur vereinfacht die Datenaggregation und die lokale Zusammenarbeit der Sensorknoten, erlaubt die Extraktion von impliziten Daten dynamischer Prozesse und unterstützt die Anpassung der Software zur Laufzeit.

3.3. Entwurf von Strategien zur Administrierung und Konfiguration eines Sensornetzwerkes

In Sensornetzwerken ist die Installation, Aktualisierung und Steuerung von unterschiedlichen Sensorknotenanwendungen sehr aufwändig. Dazu sind mindestens ein erfolgreicher Bootvorgang und ein initialisiertes Kommunikationssystem auf jedem Sensorknoten notwendig. Gerade in der Entwicklungsphase sind diese Voraussetzungen nicht gegeben. Zudem ist die manuelle Administrierung eines Sensornetzwerkes mit hunderten von Knoten aufgrund des zeitlichen Aufwandes nahezu unmöglich.

Da die sequentielle Simulation eines verteilten Systems aus autarken Knoten zahlreiche Zeit- und Synchronisationsprobleme birgt, entwickelten wir die Performance- und Monitoringumgebung SeNeTs [3], um einfach und schnell Sensornetzwerk-Anwendungen entwickeln und unter realen Bedingungen ausführen zu können.

SeNeTs ist eine verteiltes Softwaresystem, das es ermöglicht, große heterogene drahtlose Sensornetzwerke realitätsnah auszuführen. Es nutzt dafür neben dem

primären i.d.R. funkbasierten Übertragungskanal einen zweiten, unabhängigen Kommunikationskanal – den Administrierungskanal – über den jederzeit Sensorknotenanwendungen installiert, zurückgesetzt und neu gestartet werden können. Auf diese Weise können Knotenanwendungen überwacht und schrittweise ausgeführt werden ohne den Funkkanal des Sensornetzwerkes und damit die Ausführung der Sensornetzwerkanwendung zu stören. SeNeTs eignet sich somit hervorragend für Performancemessungen in Sensornetzwerken, das Abfragen von Informationen und das Ausführen von Diensten auf Sensorknoten. Ein weiterer Vorteil von SeNeTs ist die zeitliche Ersparnis durch die automatisierte und parallele Installation von Anwendungen auf hunderten von Sensorknoten.

Zur einfachen und benutzerfreundlichen Konfiguration und Visualisierung von Sensornetzwerken wurde die grafische Benutzeroberfläche EnviSense entwickelt [5, 8]. EnviSense ist vollständig in JAVA entwickelt worden und dadurch plattformunabhängig. Das Programm ist in der Lage, verschiedene Sensornetzwerke, z.B. Chipcon, Scatterweb und SeNeTs, gleichzeitig zu konfigurieren und zu administrieren. Zugleich ist eine Funktionskontrolle des Sensornetzwerkes möglich. EnviSense unterstützt vordefinierbare Sensorknotenprofile und kann Anwendungen auf Sensorknoten aktualisieren. Es enthält zudem umfangreiche Analysekomponenten wie eine Historyfunktion für alle Attribute eines Sensorknotens inklusive grafischem Verlauf, Visualisierung der wahren und berechneten Positionen von Sensorknoten im Raum, Darstellung von Topologie- und Routingbäumen u.v.m.

3.4. Entwicklung von Algorithmen zur dezentralen Lokalisierung von ressourcenarmen Sensorknoten

Die Analyse existierender anerkannter Positionierungsverfahren auf ihre Einsatzfähigkeit in ressourcenarmen Sensornetzwerken (APIT, Schwerpunktbestimmung, Multilateration, Robust Positioning System usw.) zeigte, dass diese bezüglich des Ressourcenbedarfes (algorithmische Komplexität und damit Rechenbedarf, Speichergröße und Kommunikationsaufwand) hohe Anforderungen stellen und zumeist einen erheblichen Energieverbrauch aufweisen.

Für einige dieser Verfahren entwickelten wir Optimierungsstrategien, die die Positionsgenauigkeit der Sensorknoten deutlich verbessern und gleichzeitig den Energiebedarf des Netzwerkes reduzieren, wodurch die Lebensdauer des Netzwerkes deutlich erhöht werden kann [6, 19]. Eine weitere Verbesserung der Positionsgenauigkeit kann durch eine günstige Anordnung der Sensorknoten erreicht werden [20]. Zur Bewertung des Lokalisierungsfehlers und des zur Berechnung nötigen Energieaufwandes wurde das Power-Error-Product (PEP) vorgestellt. Das PEP zeigte, dass der beste Kompromiss aus Positionsgenauigkeit und Energiebedarf nur bei einer sehr kleinen optimalen Übertragungsreichweite erreicht wird.

Eine deutliche Schwäche des CL-Verfahrens besteht in dem höheren Positionierungsfehler in den Randgebieten eines begrenzten Sensornetzwerkes. Das neu präsentierte Verfahren Centroid Localization with Edge Correction (CLwEC) reduziert diesen randbedingten Fehler erheblich [1, 9]. Die dazu notwendigen Modifikationen am originalen Schwerpunktverfahren führen nur zu geringfügig höheren Ressourcenanforderungen, die sich in minimalem Speichermehrbedarf und längeren Nachrichten äußern. Es ist daher für ressourcenlimitierte Sensornetzwerke prädestiniert.

Eine weitere Verbesserung des CL-Verfahrens wurde durch die Überführung der einfachen in eine gewichtete Schwerpunktbestimmung (Weighted Centroid Localization, WCL) erreicht. Es wurde nachgewiesen, dass unterschiedliche Gewichtsfunktionen den Positionierungsfehler maßgeblich reduzieren und bei für Sensornetzwerke typischen fehlerbehafteten Distanzmessungen mit hoher Varianz die Genauigkeit einer Trilateration erreichen können [1].

Die Betrachtungen bestätigten außerdem, dass auch für den WCL-Algorithmus optimale Übertragungsreichweiten der Referenzknoten existieren, die eine Fehlerminimierung in Abhängigkeit der Gewichtsfunktion erlauben. Es wurde gezeigt, dass diese optimalen Übertragungsreichweiten abhängig von der Anordnung der Referenzknoten sind. Bei gleicher Referenzknotendichte ergibt die quadratische Anordnung der Referenzknoten die geringsten Fehler.

3.5. Gruppenbildung in Sensornetzwerken

Ausgehend von den hohen Anforderungen an Software in Sensornetzwerken entwickelten wir ein dreistufiges dezentrales Gruppierungsverfahren zur ressourcenarmen Detektion und Beschreibung von Sensorknoten [12, 16, 17]. In der ersten Stufe erfolgt die abstrakte Beschreibung der Eigenschaft, die ein Knoten besitzen muss, um Gruppenmitglied zu werden. Jedes Gruppenmitglied ermittelt autark durch Kooperation mit seinen Nachbarn die Gruppengrenzen. Der dazu notwendige Kommunikationsaufwand ist extrem gering, da nur Nachrichten aus fest definierten Richtungen eine Reaktion in einem Sensorknoten hervorrufen, wodurch unkontrolliertes Fluten des Netzwerkes verhindert wird. Das Verfahren ist im Speicherverbrauch ebenfalls extrem ressourcensparend. Die Speicherung der Grenzinformationen benötigt auf einem Sensorknoten lediglich konstant 64 Byte pro Gruppe in einem zweidimensionalen Raum.

Nach der Ermittlung der Gruppengrenzen wird durch die Gruppenverwaltung definiert, wie die Datenweiterleitung erfolgt bzw. wie die Daten aggregiert werden sollen. Diese Verfahren bilden die Voraussetzung zu Untersuchungen der dezentralen Phänomendetektion in mobilen Sensornetzwerken, da sie an beliebigen Orten im Netzwerk bereits aggregierte Daten über Gruppen bzw. Phänomene

ermöglichen. Ihr prädestinierter Einsatzzweck sind daher vor allem Sensornetzwerke, in denen ein Nachrichtenverkehr von und zur Basisstation unerwünscht bzw. unsinnig ist, weil eine Auswertung vor-Ort möglich und zugleich viel effizienter ist.

4. Zusammenfassung

In diesem Projekt untersuchten wir aktuell in der Diskussion stehende Middleware-Technologien und stellten ein eigenes Konzept mit Ansätzen zur automatischen Generierung von Sensornetzwerkanwendungen vor. Unsere Analysen ergaben, dass sich die meisten herkömmlichen Architekturen nicht für den direkten Einsatz in Sensornetzwerken eignen, da sie unter zu Sensornetzwerken konträren Designzielen entworfen wurden und wichtigen Kriterien wie der notwendigen Minimierung des Kommunikations- und Ressourcenaufwandes nicht entsprechen. Stattdessen konzentrieren sich die Systeme auf die Kapselung der Heterogenität der Systeme.

Die Entwicklung einer Service-Architektur für die von uns verwendeten Chipcon-Sensorknoten machte es erforderlich, zunächst die dynamische Rekonfigurierbarkeit von Anwendungen per Funkübertragung (OTA: Over-the-Air) sicherzustellen. Dazu entwickelten wir ein OTA-BIOS, das auf jedem Sensorknoten fest installiert und nicht verändert werden kann. Dieses BIOS übernimmt rudimentäre Betriebssystem-Funktionen wie Initialisierung der Hardware (Timer, Interrupts), Übertragung von Nachrichten und das Neuprogrammieren (Flashen) von Anwendungen. Alle anderen Softwarekomponenten (Middleware, Dienste, Anwendungen) können zur Laufzeit ausgetauscht werden. Dazu wurde ein Übertragungsprotokoll entwickelt, das die korrekte Übertragung und Initialisierung der Software auf den Sensorknoten sicherstellt.

Das vorgestellte OTA-BIOS ist die Voraussetzung für die Entwicklung unserer mobilen ressourcen-optimierten Service-Architektur (RASA). Die Service-Architektur vereinfacht die Datenaggregation und die lokale Zusammenarbeit der Sensorknoten, erlaubt die Extraktion von impliziten Daten dynamischer Prozesse und unterstützt die Anpassung der Software zur Laufzeit.

Da die sequentielle Simulation eines verteilten Systems aus autarken Knoten zahlreiche Zeit- und Synchronisationsprobleme birgt, entwickelten wir die Performance- und Monitoringumgebung SeNeTs. SeNeTs ist ein verteiltes Softwaresystem, das es ermöglicht, große heterogene drahtlose Sensornetzwerke einfach und schnell zu entwickeln und realitätsnah auszuführen.

Zur einfachen und benutzerfreundlichen Konfiguration sowie Visualisierung von Sensornetzwerken wurde die grafische Benutzeroberfläche EnviSense entwickelt. Das Programm ist in der Lage, verschiedene Sensornetzwerke gleichzei-

tig zu konfigurieren und zu administrieren. EnviSense unterstützt vordefinierbare Sensorknotenprofile und kann Anwendungen auf Sensorknoten aktualisieren.

Die Analyse existierender anerkannter Positionierungsverfahren zeigte, dass diese bezüglich des Ressourcenbedarfs oft hohe Anforderungen stellen und zumeist einen erheblichen Energieverbrauch aufweisen. Für einige dieser Verfahren entwickelten wir Optimierungsstrategien, die die Positionsgenauigkeit der Sensorknoten deutlich verbessern und gleichzeitig den Energiebedarf des Netzwerkes reduzieren, wodurch die Lebensdauer des Netzwerkes deutlich erhöht werden kann. Eine weitere Verbesserung der Positionsgenauigkeit kann durch eine günstige Anordnung der Sensorknoten erreicht werden. Zur Bewertung des Lokalisierungsfehlers und des zur Berechnung nötigen Energieaufwandes wurde das Power-Error-Product (PEP) vorgestellt.

Eine deutliche Schwäche vieler Verfahren besteht in dem höheren Positionierungsfehler in den Randgebieten eines begrenzten Sensornetzwerkes. Das neu präsentierte Verfahren Centroid Localization with Edge Correction (CLwEC) reduziert diesen randbedingten Fehler erheblich.

Ausgehend von den hohen Anforderungen an Software in Sensornetzwerken entwickelten wir ein dreistufiges dezentrales Gruppierungsverfahren zur ressourcenarmen Detektion und Beschreibung von Sensorknoten. Der dazu notwendige Kommunikationsaufwand ist extrem gering, da nur Nachrichten aus fest definierten Richtungen eine Reaktion in einem Sensorknoten hervorrufen, wodurch unkontrolliertes Fluten des Netzwerkes verhindert wird.

Die vorgeschlagenen Softwarearchitekturen und entwickelten Algorithmen stellen eine solide Basis für die Entwicklung von Anwendungen für ressourcenarme Sensornetzwerke dar. Sie ermöglichen vor allem den Einsatz in Systemen mit eingeschränkten Energiereserven.

5. Ausblick

Zusätzlich zu den oben genannten Ergebnissen wurden während des Projektes Bereiche aufgedeckt, die einer weiteren Forschung bedürfen. Sie werden unten im Einzelnen kurz aufgeführt.

In hoch-dynamischen Netzwerken sollten noch Modifikationen näher betrachtet werden, die den möglicherweise erhöhten Kommunikationsaufwand begrenzen. Dies betrifft vor allem das kommunikationsintensive Tracking von sehr mobilen Objekten, wenn die Synchronisation zwischen den beteiligten Knoten fehl schlägt und dadurch die mobilen Objekte scheinbar verloren gehen.

Im Rahmen der Weiterentwicklung der Gruppenbildung als Basis für die Bereitstellung kooperativer Dienste ist zu überprüfen, ob die vorgeschlagene Aufteilung

der Umgebung eines Knotens in statische bzw. adaptive Kreissektoren noch präziser beschrieben werden kann. Es sollte untersucht werden, ob zusätzliche Informationen der Umgebung in der Detektion der Gruppengrenze miteinbezogen werden können. Beispielsweise ist es vorstellbar, dass die Gruppenzugehörigkeit nicht binär auf Basis einer Eigenschaftsdefinition sondern auf Basis einer Entwicklungstendenz gefällt wird.

Die notwendigen theoretischen Forschungen werden derzeit erschwert, da für große Netzwerke aus hunderten, winzigen Knoten keine wirtschaftlich erschwingliche und vor allem stabile Hardware existiert, wodurch aussagefähige Feldtests mit den entwickelten Algorithmen zumeist nur simulativ durchgeführt werden können. Aktuell verfügbare Sensorknoten nutzen meist nur mäßig geeignete Übertragungstechnologien, die anfällig auf Störquellen sind, eine Distanzbestimmung erschweren und relativ große Antennen besitzen.

Literaturverzeichnis

[1] BLUMENTHAL, JAN: *Ressourcenarme und dezentrale Lokalisierung autonomer Sensorknoten in Sensornetzwerken*. Doktorarbeit, Universität Rostock, Deutschland, 2008.

[2] BLUMENTHAL, JAN, STEVE DÜBEL und DIRK TIMMERMANN: *Software Updates for Sensor Networks in Changing Process Environments*. In: *5th International Forum "Life Science Automation"*, Seite 60, Landsdowne, Virginia, USA, Oktober 2007.

[3] BLUMENTHAL, JAN, FRANK GOLATOWSKI, MARC HAASE und MATTHIAS HANDY: *The Industrial Information Technology Handbook*, Band 1 der Reihe *Industrial Electronics*, Kapitel Software Development for Large-Scale Wireless Sensor Networks. CRC Press, San Francisco, USA, November 2004.

[4] BLUMENTHAL, JAN, MATTHIAS HANDY, FRANK GOLATOWSKI, MARC HAASE und DIRK TIMMERMANN: *Wireless Sensor Networks - New Challenges in Software Engineering*. In: *9th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Seiten 551–555, Lissabon, Portugal, September 2003.

[5] BLUMENTHAL, JAN, FRANK REICHENBACH, FRANK GOLATOWSKI und DIRK TIMMERMANN: *Controlling Wireless Sensor Networks using SeNeTs and EnviSense*. In: *3rd IEEE International Conference on Industrial Informatics (INDIN 05)*, Perth, Australien, August 2005.

[6] BLUMENTHAL, JAN, FRANK REICHENBACH, MATTHIAS HANDY und DIRK TIMMERMANN: *Low Power Optimization of the Coarse Grained Localization Algorithm in Wireless Sensor Networks*. In: *1st Workshop on Positioning, Navigation and Communication 2004 (WPNC 04)*, Seiten 137–146, Hannover, Deutschland, März 2004.

[7] BLUMENTHAL, JAN, FRANK REICHENBACH und DIRK TIMMERMANN: *Drahtlose Sensornetzwerke in der Laborautomation*. GIT Labor-Fachzeitschrift, Landau, Deutschland, Seiten 128–129, Februar 2005.

[8] BLUMENTHAL, JAN, FRANK REICHENBACH und DIRK TIMMERMANN: *EnviSense – Steuerung drahtloser Sensornetzwerke*. CeBit 2005, März 2005.

[9] BLUMENTHAL, JAN, FRANK REICHENBACH und DIRK TIMMERMANN: *Decreasing the Localization Error in Border Areas of Sensor Networks*. In: *4th IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS 2008)*, Santorini, Griechenland, Juni 2008.

[10] BLUMENTHAL, JAN und DIRK TIMMERMANN: *Middleware für mobile spontan vernetzte Sensornetzwerke*. Startkolloquium DFG-Schwerpunktprogramm Basissoftware, Karlsruhe, Deutschland, Oktober 2002.

[11] BLUMENTHAL, JAN und DIRK TIMMERMANN: *Simulation von Sensornetzwerken*. Universität Bonn, Februar 2003.

[12] BLUMENTHAL, JAN und DIRK TIMMERMANN: *Kooperative Dienste in Sensornetzwerken*. DFG-Jahreskolloquium, Lübeck, Deutschland, März 2007.

[13] BLUMENTHAL, JAN, DIRK TIMMERMANN, CARSTEN BUSCHMANN, STEFAN FISCHER, JOCHEN KOBERSTEIN und NORBERT LUTTENBERGER: *Minimal Transmission Power as Distance Estimation for Precise Localization in Sensor Networks*. In: *International Wireless Communications and Mobile Computing Conference (IWCMC 2006)*, Seiten 1331–1336, 2006.

[14] BOBEK, ANDREAS, HENDRIK BOHN und FRANK GOLATOWSKI: *UPnP AV Architecture – Generic Interface Design and Java Implementation*. In: *23rd IASTED International Multi-Conference on Applied Informatics*, Seiten 699–704, Innsbruck, Österreich, February 2005.

[15] GOLATOWSKI, FRANK, JAN BLUMENTHAL, MATTHIAS HANDY, MARC HAASE, HAGEN BURCHARDT und DIRK TIMMERMANN: *Service-Oriented Software Architektur for Sensor Networks*. In: *International Workshop on Mobile Computing (IMC 2003)*, Seiten 93–98, Rostock, Deutschland, Juni 2003.

[16] GORSKI, PHILIPP: *Gruppenbildung in Sensornetzwerken*. Studienarbeit, Universität Rostock, Deutschland, 2006.

[17] GORSKI, PHILIPP: *Gruppierungsverfahren in Sensornetzwerken*. Studienarbeit, Universität Rostock, Deutschland, 2007.

[18] REICHENBACH, FRANK: *Ressourcensparende Algorithmen zur exakten Lokalisierung in drahtlosen Sensornetzwerken*. Doktorarbeit, Universität Rostock, Deutschland, 2007.

[19] REICHENBACH, FRANK, JAN BLUMENTHAL und DIRK TIMMERMANN: *Improved Precision of Coarse Grained Localization in Wireless Sensor Networks*. In: *9th Euromicro Conference on Digital System Design (DSD 2006)*, Seiten 630–637, Dubrovnik, Croatia, September 2006.

[20] SALOMON, RALF und JAN BLUMENTHAL: *Coarse-Grained Localization: Extended Analyses and Optimal Beacon Distribution*. In: *10th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2005)*, Catania, Italien, September 2005.

[21] SCHULZ, JENS: *Entwicklung eines funkgestützten Sensorknotens zur Detektion der Feuchtigkeit in künstlichen Deichen*. Studienarbeit, Universität Rostock, Deutschland, 2006.

[22] SEIFERT, MATTHIAS: *OTA-Programmierung von Sensorknoten*. Studienarbeit, Universität Rostock, Deutschland, 2005.

[23] TIMMERMANN, DIRK, FRANK REICHENBACH, JAN BLUMENTHAL und JAKOB SALZMANN: *Lokalisierung und Robustheit in drahtlosen Sensornetzwerken*. VDE/ITG Diskussionssitzung „Internet der Dinge - Neue Entwicklungen der drahtlosen Sensornetzwerke“, Kamp-Lintfort, Deutschland, März 2007.

SWARMS – Software Architecture for Radio-Based Mobile Self-Organizing Systems

Carsten Buschmann[1], Stefan Fischer[1], Norbert Luttenberger[2] und Jochen Koberstein[2]

[1] Universität zu Lübeck, Institut für Telematik, Ratzeburger Allee 160, 23538 Lübeck
[2] Christian-Albrechts-Universität zu Kiel, AG Kommunikationssysteme

Zusammenfassung – Im SWARMS-Projekt wurde untersucht, wie die Programmierung von Anwendungen für gemeinsam operierende „Schwärme" von mobilen funkvernetzten Systemen auf der Basis eines Read/Write- Kooperationsparadigmas unterstützt werden kann. Dieses Paradigma sieht vor, dass dem Anwendungssystem ein *shared information space* zur Verfügung gestellt wird, der sowohl die einbettende Umgebung als auch den Zustand des Schwarms beschreibt. Die Einzelsysteme kooperieren durch lesende und schreibende Operationen auf den shared information space. Auf dieser Infrastrukturbasis wurden in SWARMS vor allem Fragen zur Erfassung von Kontextinformation (insbesondere Ort und Zeit) und zum Engineering von Sensornetz-Anwendungen bearbeitet. Besondere Aufmerksamkeit wurde dabei der Einbettung der Simulation in den Prozeß der Softwareentwicklung für drahtlose Sensornetze, der Durchführung von Experimenten und der prototypischen Entwicklung ausgewählter Komponenten gewidmet.

1. Einleitung

Das SWARMS-Projekt ist im Fachgebiet der Sensor- und Aktornetze angesiedelt. Entwicklung und Betrieb solcher Netze können aufgrund der Eigenschaften ihrer Knoten und der speziellen Anwendungen nicht mehr anhand der bekannten Schemata durchgeführt werden; man kann hier tatsächlich von einem Paradigmenwechsel in der Datenverarbeitung sprechen. Als besondere Herausforderungen sind vor allem zu nennen:

- die oftmals große Zahl der Knoten,
- die Ressourcenarmut der einzelnen Knoten insbesondere in Bezug auf die Verfügbarkeit von Energie,
- die Notwendigkeit einer Selbstorganisation des Netzes aufgrund der fehlenden Möglichkeit für ein menschliches Eingreifen während des Betriebs,
- die Notwendigkeit, robuste und fehlertolerante Algorithmen und Protokolle zu verwenden.

Die Vision der beiden an diesem Antrag beteiligten beiden Arbeitsgruppen bestand nun in einer weitestgehenden Automatisierung der Entwicklung und des Betriebs von Sensor- und Aktornetzen, um den genannten Herausforderungen begegnen zu können. Dazu musste zunächst ein Gesamtkonzept entwickelt werden, was dann auf eine ganze Reihe von Teillösungen heruntergebrochen wurde. Dieser Artikel stellt sowohl das Gesamtkonzept wie auch die wichtigsten dieser Teillösungen dar.

SWARMS wurde während seiner sechsjährigen Laufzeit gemeinsam von zwei Arbeitsgruppen aus Schleswig-Holstein, nämlich dem Institut für Telematik an der Universität zu Lübeck und der AG Kommunikationssysteme der Christian-Albrechts-Universität zu Kiel bearbeitet. Neben zwei Promotionen, die in Kürze fertig gestellt werden, entstanden in SWARMS insgesamt 29 Studien-, Diplom- und Masterarbeiten. Die Ergebnisse des Projekts wurden in 33 Publikationen der wissenschaftlichen Öffentlichkeit präsentiert.

Der Rest des Artikels ist wie folgt gegliedert: nach dieser kurzen Einführung wird in Abschnitt 2. die Problemstellung des SWARMS-Projektes ausführlich dargestellt. Abschnitt 3. stellt den generellen Lösungsansatz für die geschilderte Problematik, also das SWARMS-Konzept dar. Der sich anschließende Abschnitt 4. präsentiert die wichtigsten Lösungen, die im Rahmen des Projektes erarbeitet wurden. Der Artikel schließt in Abschnitt 5. mit einer Zusammenfassung, die auch eine kurze Beschreibung der Einbettung von SWARMS in das gesamte Schwerpunktprogramm enthält.

2. Das Problem

Sensoren und Aktoren erleichtern bereits heute das Alltagsleben der Menschen. Beispiele sind Bewegungssensoren und damit verbundene Lichtsteuerungen, Temperatursensoren zur Heizungssteuerung oder Abstandssensoren in Kraftfahrzeugen. Jedoch werden solche Sensoren in der Praxis bisher überwiegend isoliert für lokale Aufgaben eingesetzt. Der Zusammenschluss der Sensoren zu *Sensornetzen*, die für die Durchführung umfassenderer Aufgaben genutzt werden können, ist heute erst in prototypischen Ansätzen realisiert. Das Potential zu deren Realisierung wird durch den technischen Fortschritt jedoch rasch größer: Einerseits können immer bessere Sensoren immer billiger produziert werden, wobei diese Sensoren gleichzeitig immer weniger Energie verbrauchen. Andererseits setzen sich drahtlose Kommunikationstechniken wie WLAN, Bluetooth oder Ultra Wide Band auf breiter Front durch und können ebenfalls kostengünstig eingesetzt werden. Dies wird dazu führen, dass Anwendungen z.B. zur Umweltüberwachung, zur Verkehrssteuerung oder zur Realisierung von „Assisted Living“ in der Pflege mit großer Wahrscheinlichkeit in naher Zukunft auch in der Praxis auf breiter Front realisiert werden.

Es ist jedoch noch keineswegs klar, wie System- und Anwendungsarchitekturen für solche Netze und deren einzelne Knoten aussehen werden. Vielmehr bestehen aus wissenschaftlicher Sicht zu den heute üblichen Netzverbünden substantielle Unterschiede; Sensornetze stellen einen Paradigmenwechsel dar [1, 28]:

- Sensornetze können aus vielen tausend sehr einfachen Einzelknoten bestehen, die alle in dieselbe Anwendung eingebunden sind. Zwar wird die Netzgröße eines Sensornetzes von Verbünden wie dem Internet um viele Größenordnungen übertroffen; dort ist aber die gemeinsame Erledigung von Aufgaben nur in einem erheblich kleineren Umfang üblich, so dass außer der reinen Kommunikation keine wirklich dezentral organisierte, übergreifende verteilte Anwendung stattfindet. Somit ergeben sich für Sensornetze ganz andere Anforderungen an die Skalierbarkeit der Kooperations- und Kommunikationsmechanismen. Auch müssen aus diesem Grund die üblichen Verfahren für Design und Implementierung von System- und Anwendungssoftware für verteilte Systeme überdacht werden.
- Ebenfalls deutlicher Innovationsbedarf ergibt sich aus der Leistungsfähigkeit der Einzelknoten: In einem Sensornetz verfügen diese zumeist nur über minimale Ressourcen (Speicher, Energie, Prozessorleistung). Dies macht es unumgänglich, durch umfassende Kooperation die global vorhandenen Ressourcen auch übergreifend nutzbar zu machen. Es müssen entsprechend Algorithmen und Kommunikationsprotokolle gefunden werden, die speziell auf große verteilte Systeme aus schwachen Einzelknoten ausgelegt sind.
- Anwendungen in Sensornetzen sind sehr oft datenzentrisch (evtl. ergänzt um eine dienstbasierte Sicht); ein einzelnes Netzwerk bedient oft nur eine einzige Anwendung, die um eine Aggregation der gesammelten Daten herum aufgebaut ist. Folglich ist es notwendig, ganz neuartige Überlegungen bzgl. einer stark anwendungsbezogenen, skalierbaren Netzwerkarchitektur anzustellen.
- Robustheit wird zu einer wichtigen Anforderung, die sich durch Redundanz und deren geschickten Nutzung umsetzen lässt.
- Die einzelnen Knoten haben meist keinerlei Benutzersteuerung, und das System als Ganzes muss über größere Zeiträume ohne Benutzereingriff und unüberwacht funktionieren.
- Die Datenerfassung in Sensornetzen erfolgt automatisch, online und in Realzeit, und die Speicherung und Verarbeitung der Daten wird größtenteils im Sensornetz selbst ausgeführt.

Diese besonderen Bedingungen führen nun dazu, dass man sich sorgfältige Gedanken über die Funktionalität der System- und Anwendungssoftware der Sensorknoten, ihre Anordnung in der Systemarchitektur und schließlich über den Softwareerstellungsprozess machen muss. Die Ziele des SWARMS-Projekt bestanden im

Wesentlichen darin, ein innovatives Lösungskonzept für die Problematik zu entwerfen und in einigen Teilbereichen umzusetzen.

3. Das Grundkonzept von SWARMS

Die Mehrheit der in der Literatur betrachteten Sensornetze basiert auf ortsfesten Knoten (siehe z.B. [27, 33, 35]), d.h. Topologieänderungen des Netze kommen nur durch äußere Einflüsse zustande und sind wesentlich seltener als in mobilen Netzen. Dementsprechend sind auf niedrige Bewegungsgeschwindigkeiten angepasste Routingverfahren wie z.B. AODV, DSR, OLSR usw. und zugehörige Kommunikationsparadigmen anwendbar. Im SWARMS-Projekt gehen wir hingegen explizit davon aus, dass die Sensorknoten hochmobil sind, was die Nutzung von Routingverfahren aufgrund der ständigen Topologieänderungen ineffizient macht [14]. Ferner gehen wir davon aus, dass der einzelne Sensorknoten selbst an den im Netz gesammelten Informationen interessiert ist, z.B. um seine Kommunikation an die aktuelle Situation anzupassen oder gar Aktionen wie angepasste Bewegungen durchzuführen.

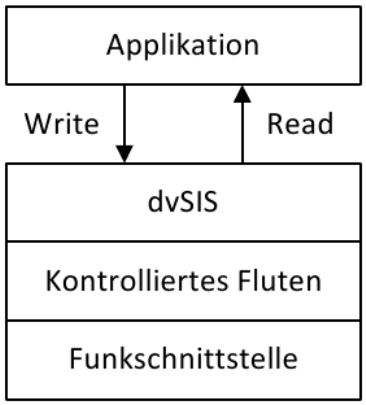
Im SWARMS-Projekt wurde deshalb untersucht, wie das datenzentrierte Paradigma des verteilten virtuellen gemeinsamen Informationsraums (*distributed virtual shared information space*, dvSIS [5, 21]) genutzt werden kann, um "Schwärme" von mobilen funkvernetzten *small footprint*-Rechensystemen zu programmieren und zu betreiben. Das dvSIS-Paradigma sieht vor, daß dem Anwendungssystem ein *shared information space* zur Verfügung gestellt wird, der einerseits Sensordaten aus der einbettenden Umgebung und andererseits Informationen zum Zustand und zum Operationskontext des Schwarms enthält.

Der gemeinsame Informationraum kann von allen beteiligten Knoten des Netzes gemeinsam gelesen und beschrieben werden, d.h. die Einzelsysteme kooperieren durch lesende und schreibende Zugriffe auf diesen *shared information space*. Dieser Ansatz ist auf den ersten Blick sehr ähnlich den bekannten Tupel Räumen wie z.B. Linda [11], TSpaces [37] oder JavaSpaces [32]. Im Gegensatz zu diesen Ansätze gehen wir von der Verteilung des Informationsraumes auf ein drahtloses Netz von Knoten voraus und vermeiden das Tupel-Paradigma. Da es ferner keine zentrale Instanz gibt, hält jeder beteiligte Knoten eine lokale Sicht auf den Informationraum. Diese Sichten müssen nicht zwangsläufig konsistent zu anderen lokalen Sichten sein, da nicht immer alle Mitglieder des Schwarms vollständig funktionsfähig und funktechnisch erreichbar sind. Aus diesem Grunde ist dieser Informationraum nur virtuell vorhanden, d.h. typischerweise besitzt kein Knoten die gesamte im Netz vorhandene Information.

Um die Information im Netz zu verteilen und mit anderen Knoten zu teilen, wird das Netz kontrolliert geflutet [18, 25, 31]. Die Kontrolle dieses Flutens ist dabei

inhaltsbasiert, was das Versenden von Verwaltungsinformationen wie z.B. Nachbarschaftslisten oder Routinginformationen prinzipiell überflüssig macht. Die Auswahl, welche Informationen versendet werden, ist applikationsspezifisch. Ein sehr rudimentärer Ansatz wäre z.B., neuere Messwerte gegenüber älteren Messwerten beim Fluten zu bevorzugen oder bereits mehrfach empfangene Datensätze nicht erneut zu versenden.

Die untersuchte Architektur umfaßt–wie untenstehendes Bild zeigt–vier Schichten, wobei im folgenden nur die beiden inneren Schichten kurz diskutiert werden.



Der **dvSIS** dient der Applikation als Abstraktionsschnittstelle, die es ihr ermöglicht, von den unterlagerten Details des Datenaustauschs zu abstrahieren. Die Applikation kann mit *read*-Operationen auf den *local view* der im gesamten Netz vorhandenen Informationen zugreifen. Mit *write*-Operationen wird die Weiterleitung der jeweils geschriebenen Information an die anderen Knoten des Netzes initiiert. Unter diesen anderen Knoten kann auch ein Gateway zum Festnetz (z.B. dem Internet) sein.

Der dvSIS wird als Dokument modelliert, für das eine Dokumentengrammatik vorliegt. Diesbezüglich wurden sowohl Modellierungen mit W3C XML Schema als auch mit ASN.1 untersucht. Die Abstützung auf dem dokumentenbasierten Ansatz und den zugehörigen formalen Grammatiken wurde dazu genutzt, die Implementierung der lesenden und schreibenden Operationen durch einen Generierungsprozeß automatisch zu erzeugen, und darüberhinaus ein Internet-Gateway zu konstruieren, das die Information, die z.B. durch ein Sensornetz erfaßt wird, weltweit zur Verfügung zustellen.

Die Entscheidung, sich mit dem **kontrollierten Fluten** zu befassen, wurde aus zwei Gründen getroffen: Zum einen wird das Fluten durch die Broadcast-Charakteristik der Funkkommunikation ideal unterstützt; zum anderen läßt sich auf der Basis des dvSIS das Fluten effizient kontrollieren: Jeder Knoten vergleicht die Information, die er über seine Funkschnittstelle empfängt, mit der Information, die er in seinem *local view* gespeichert hat. Findet er sie dort vor, leitet er sie nicht mehr weiter. Wir nennen diese Art der Flutungskontrolle *content-based flooding control* (CBFC). CBFC kommt also ohne ein eigenes *Flooding Control Protocol* mit ggf. komplexen Mechanismen für die Nachrichtenidentifikation aus.

Auf dieser Infrastrukturbasis wurden in SWARMS vor allem Fragen zur Erfassung von Kontextinformation (insbesondere Ort und Zeit) und zum Engineering von Sensornetz-Anwendungen bearbeitet. Die entsprechenden Untersuchungen erfolgten u.a. mit einer Anzahl von prototypischen Entwicklungen und praktischen Experimenten. Diese werden im folgenden Kapitel ausführlich dargestellt.

4. Lösungen und Umsetzungen

4.1. dvSIS

Der dvSIS-Ansatz ist datenzentriert und erfordert deshalb eine aussagekräftige Strukturierung der Information, welche in Bezug auf *Wireless Sensor Networks* (WSN) aus Messdaten und zugehörigen Metadaten (Kontext) besteht. Das Modell des dvSIS fußt auf einem dokumentenorientierten Ansatz. Nutzt man XML zur Dokumentenkodierung, dann lässt sich die Menge der gültigen dvSIS-Instanzen mit Hilfe einer Grammatik in Form eines W3C XML-Schema [2, 34] definieren. Da die zwischen Knoten ausgetauschten Nachrichten Ausschnitte des dvSIS enthalten, müssen auch diese der Grammatik genügen. Entsprechend lassen sich empfangene Nachrichten von jedem Knoten gegen die Grammatik validieren, ungültige Nachrichten können unmittelbar verworfen werden.

Typische Sensorknoten sind stark ressourcenbeschränkte Geräte, auf denen übliche Werkzeuge zur Verarbeitung von XML-Daten schlicht nicht lauffähig sind. Dementsprechend wurde im Rahmen des SWARMS Projekts ein besonders effizientes XML Language Binding entwickelt [17, 24], was schlussendlich nicht nur das Parsen, sondern auch das Validieren von semi-strukturierten Daten auf Sensorknoten ermöglicht. Zwei Kernaspekte sind dabei die ereignisbasierte Verarbeitung der Daten und die zur Validierung eingesetzten *Cardinality Constraint Automatas* (CCA).

Ein auf dem dvSIS-Paradigma aufsetzender Entwicklungsprozess beginnt so bei der Definition einer Grammatik [6], d.h. dem Schreiben eines W3C XML-Schema Dokuments, welches die Struktur des dvSIS beschreibt. Davon ausgehend wird mit Hilfe des Language Bindings entsprechender Quellcode generiert, der zum einen die (De-) Serialisierung und die Validierung von Daten ermöglicht. Zum anderen werden Datenstrukturen und Methodenrümpfe zur Handhabung des dvSIS generiert, die als Basis für die zu entwickelnde Applikation dienen.

4.2. Lokation und Kontext

Die Kenntnis des Umfeldes, in dem ein Selbstorganisationsprozess stattfinden soll, ist maßgeblich für dessen Gestaltung. Das bedeutet, dass ohne eine detaillierte Kenntnis des Kontextes ein solcher Prozess nicht zielführend gestaltet werden kann. Daher ist die Ermittlung der begleitenden Umstände praktisch nicht vom Organisationsprozess als solchem zu trennen. Somit sollen die entsprechenden Forschungsarbeiten an dieser Stelle auch gemeinsam vorgestellt werden.

Ein wesentliches Kontextelement wird von Informationen über die direkt benachbarten Knoten gestellt. Dabei ist insbesondere der Abstand zu anderen Knoten von Interesse. Die Kenntnis der Distanzen zu Nachbarn ist in vielen Situationen hilfreich, so beispielhaft um eigene Messergebnisse von nahen Knoten bestätigen zu

lassen, oder um die Distanzen entlang eines Routingpfades zu messen. Aus diesen Gründen haben wir das „Neighborhood Intersection Distance Estimation Scheme“ (NIDES) zur Distanzschätzung [9] entwickelt.

NIDES basiert auf der Beobachtung, dass zwei benachbarte Knoten viele (weitere) Nachbarn gemein haben, wenn sie dicht beieinander liegen, wohingegen sie weniger Nachbarn teilen werden, wenn sie weiter voneinander entfernt sind. Nehmen wir an, dass der Kommunikationsradius eines Knoten durch einen Kreis repräsentiert wird, in dessen Mitte der Knoten liegt. Dieser Kreis bildet mit dem Kommunikationsradius benachbarter Knoten einen Schnitt. Je größer die Schnittfläche, desto dichter liegen die beiden Knoten beieinander (siehe Abbildung 1).

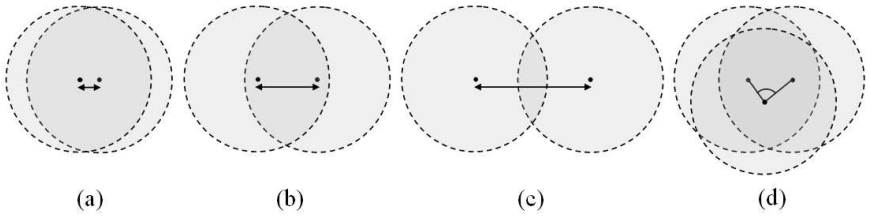


Abb. 1. Nachbarschaftüberschneidung bei verschiedenen Distanzen und Winkeln.

Es lässt sich nun ein funktionaler Zusammenhang zwischen der Schnittfläche und der Distanz der beiden Knoten herleiten. Fasst man den Anteil der gemeinsamen Nachbarn zweier Knoten als ein Maß für die Größe des Schnittes ihrer Kommunikationsradien auf, kann man von dem Anteil der gemeinsamen Nachbarn auf die Distanz der Knoten schließen.

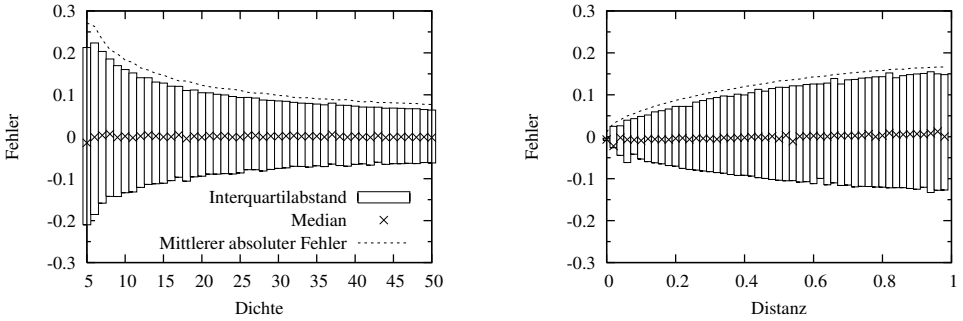
$$d(s/n) = -4\sqrt{2}r \sin\left(\frac{1}{3}\arcsin\left(\frac{3}{16}\sqrt{2}(\frac{s}{n}-1)\pi\right)\right). \quad (1)$$

Die geschätzte Distanz d ist abhängig von der Gesamtanzahl der Nachbarn n, der Zahl der gemeinsamen Nachbarn s sowie dem Kommunikationsradius. Details können [7] entnommen werden.

Auf Basis dieser Überlegungen haben wir ein Protokoll zu Distanzschätzung entwickelt. Zunächst senden alle Knoten eine „hello“ Nachricht aus, die es allen Knoten ermöglicht, ihre Nachbarn kennen zu lernen. Die Knoten tragen ihre Nachbarn in eine Liste ein, die später noch um die Distanzen zu den Nachbarn ergänzt wird. Sie wird im Speicher gehalten und ist nach den IDs der Nachbarn sortiert, um einen schnellen Zugriff zu gewährleisten.

In einem zweiten Paket broadcasten alle Knoten ihre Nachbarschaftsliste. Die Nachbarn können bei Empfang der Liste mit einem einzigen Durchlauf die Anzahl s der gemeinsamen Nachbarn bestimmen, daraus die Distanz bestimmen und in ihre eigene Nachbarschaftsliste eintragen. Danach kann die empfangene Liste verworfen werden, nur die eigene Liste muss dauerhaft im Speicher gehalten werden. Die

komplexesten Kalkulationen zur Berechnung der Distanz sind sin und arcsin, die als einfache Tabellenfunktionen implementiert werden können. Somit kann NIDES auch auf ressourcenbeschränkten, mobilen Geräten aus dem Bereich des Ubiquitous Computing oder auf Sensorknoten eingesetzt werden.



(a) Fehlerverteilung als Bruchteil des Kommunikationsradius über der Netzwerkdichte.

(b) Fehlerverteilung als Bruchteil des Kommunikationsradius über der tatsächlichen Distanz.

Abb. 2. Fehlercharakteristik mit dem Kreismodell.

Wir haben die Genauigkeit des Verfahrens in umfangreichen Simulationen untersucht. Abbildung 2(a) zeigt die Verteilung des Schätzfehlers in Abhängigkeit von der Netzwerkdichte. Negative Werte stehen dabei für eine Unterschätzung der Distanz, positive für eine Überschätzung. Wie zu erwarten ist die Genauigkeit der Schätzungen abhängig von der Netzwerkdichte. Geht man von einer Dichte von 15 aus, was für Sensornetze eine sinnvolle Annahme ist, liegt der mittlere Fehler bei 0,15. Es weisen 50% der Distanzschätzungen einen Fehler von weniger als 0,12 auf, und nur 10% der Schätzungen weisen einen Fehler von etwa 0,3 oder mehr auf. Abbildung 2(b) zeigt die Fehlerentwicklung in Abhängigkeit von der tatsächlichen Distanz der Knoten auf. Dabei wird wiederum von einer Dichte von 15 ausgegangen, Distanz und Fehler sind als Bruchteile des Kommunikationsradius angegeben. Es ist zu erkennen, dass der Fehler mit der Distanz zunimmt. Das liegt daran, dass mit zunehmender Distanz die Überlappung der Kreise abnimmt, und somit Informationen aus einer kleineren Fläche ausgewertet werden. Schwankungen in der Knotenverteilung beispielsweise können sich so stärker auswirken. Insgesamt konnte gezeigt werden, dass NIDES im Vergleich mit anderen Verfahren eine herausragende Genauigkeit bei der Schätzung von Distanzen aufweist.

In [7] wurde aufgezeigt, dass das hier vorgestellte Verfahren auch auf komplexere Radiomodelle angewendet werden kann, ohne dass die Genauigkeit sich verschlechtert. Darüber hinaus wurde in [10] dargestellt, wie NIDES zu einem Multihop-Distanzschätzverfahren erweitert werden kann.

4.3. Sensor Web Enablement

Das dvSIS-Paradigma ermöglicht, wie in Abschnitt 4.1. dargestellt, die Kommunikation innerhalb des Sensornetzes, d.h. zwischen den beteiligten Sensorknoten selbst. Soll die vom Sensornetz erfasste Information nicht nur innerhalb des Sensornetzes, sondern auch darüber hinaus genutzt werden, ist ein entsprechender Netzübergang zwischen Sensornetz und dem Netz notwendig, in dem die Daten "konsumiert" werden. Konsumierenden Netzen können z.B. lokale kabelgebundene Netze oder das World Wide Web sein.

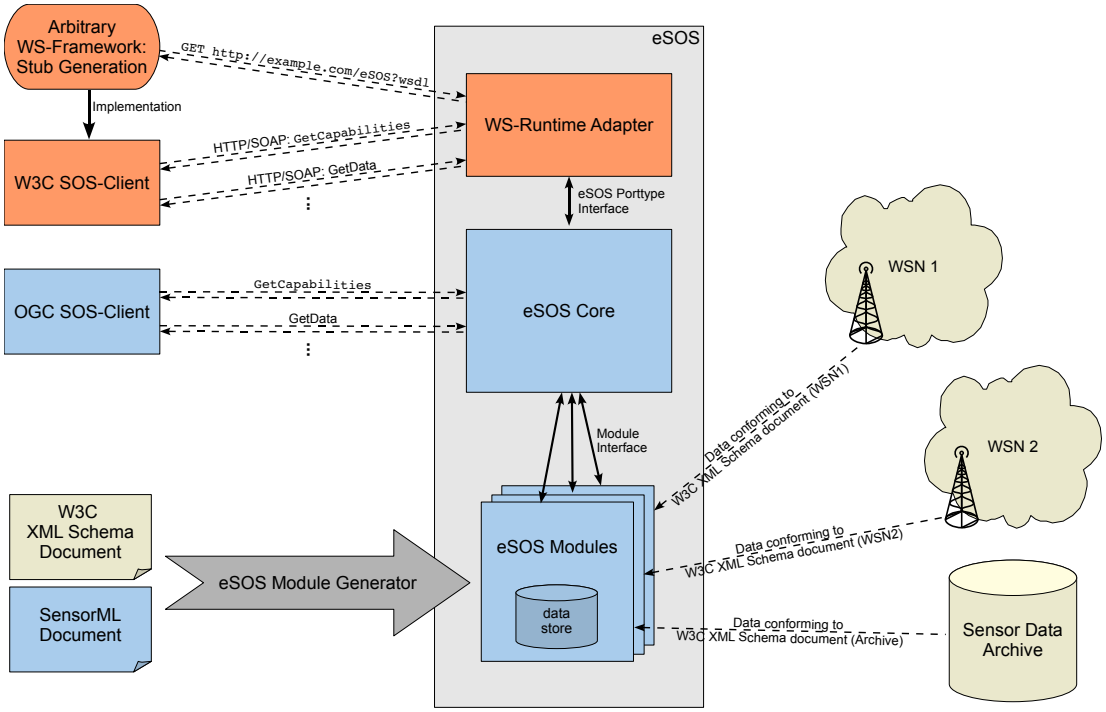


Abb. 3. Architektur des eSOS

Solche Netzübergänge werden häufig auf das zugrundeliegende Sensornetz angepasst, um aus diesem die Information abfragen zu können. Einer der populärsten Ansätze für TinyOS basierte Netze ist TinyDB [26], welcher selektive Anfragen von Sensordaten mit Hilfe einer SQL ähnlichen Syntax erlaubt. Die hier avisierten WSNs sind von kleinerer bis mittlerer Größe und die Bildung größerer Hierarchien solcher Netzübergänge ist nicht vorgesehen. Im Gegensatz dazu wurden durch das *Open Geospatial Consortium* (OGC) diverse Standards entworfen, welche die möglicherweise weltweite Nutzung, Strukturierung und Verknüpfung von verschiedensten Sensordatenquellen zum Ziel haben. Einige dieser Standards beschäftigen sich mit der Definition von XML-basierten Sprachen zur Beschreibung und Abfrage von Sensordaten und zugehörigen Metainformationen. Zu den Entwicklungen der OGC

gehört auch der sog. *Sensor Observation Service* (SOS, [29]), welcher einen Netzübergang bereitstellt.

Im Rahmen des SWARMS-Projektes wurde ein Netzübergang entwickelt, der insbesondere folgenden Anforderungen genügt:

- Leichte Einbindung in existierende Strukturen seitens der konsumierenden Netze
- Anbindung von dvSIS-basierten WSNs
- Selektive Abfrage von Sensordaten

Dieser sogenannte *extended Sensor Observation Service* (eSOS, siehe Abb. 3) bietet auf Seiten des konsumierenden Netzes sowohl eine SOS konforme Schnittstelle entsprechend der OGC Standards an, als auch eine W3C Web Service konforme Schnittstelle. Dadurch lässt sich dieser Netzübergang sowohl in OGC konforme Netze und Hierarchien als auch in W3C konforme Service orientierte Architekturen nahtlos einbinden.

Diese Schnittstellen ermöglichen die Abfrage von Metainformationen bzgl. der erfassenden WSNs und die Abfrage von Sensordaten nach bestimmten Kriterien, wie z.B. Ort und Zeit aber auch der Messwerte selber. Die Definition dieser Kriterien selbst erfolgt dabei bei beiden Schnittstellen mit Hilfe OGC konformer Dokumente.

Um Sensordaten aus dvSIS-basierten WSNs zu erfassen, zu speichern und im eSOS abfragbar zu machen, kommen sog. eSOS Module zum Einsatz. eSOS Module greifen Sensordaten über eine entsprechende Funkschnittstelle im WSN ab und legen sie in einer internen Datenstruktur ab. Diese wird über kd-Bäume und/oder R-Bäume indiziert, wodurch Anfragen sehr effizient durchgeführt werden können. eSOS Module werden *vollständig* aus dem W3C XML-Schema des jeweiligen dvSIS und einem SensorML Dokument [3] generiert, welches Metainformationen zum entsprechenden WSN enthält. Ferner werden eSOS Module dynamisch in den eSOS geladen, was das An- und Abmelden von WSNs am eSOS zur Laufzeit ermöglicht.

4.4. Entwicklung, Test und Evaluation von Software

Die Entwicklung von Anwendungen für Sensor-/Aktor-Netze ist aus verschiedenen Gründen schwierig. Einerseits verfügen die meisten Geräte über keinerlei Benutzerschnittstelle, andererseits ist insbesondere die Kommunikation zwischen den Geräten sehr fehleranfällig, weiterhin ist die Softwareentwicklung aufgrund der inhärenten Verteiltheit von Sensor-/Aktor-Netzen zusätzlich erschwert. Daher ist die Simulation von Sensornetzanwendungen ein wichtiges Werkzeug, um bereits zum Entwicklungszeitpunkt die Tragfähigkeit der entworfenen Konzepte, Algorithmen, Protokolle und Anwendungen überprüfen zu können. Von besonderem Wert sind Simulationen, wenn der Simulator den gleichen Code ausführt wie

nach dem Deployment die tatsächlichen Geräte, da dann auch beispielsweise durch Programmierfehler entstehende Fehlfunktionen aufgedeckt werden können, nicht zuletzt durch Einsatz von Debuggern, wie sie auf eingebetteter Hardware häufig nicht zur Verfügung stehen. Hier wurde im Rahmen des SWARMS-Projekts das im nächsten Abschnitt vorgestellte System Spyglass entwickelt. In Zusammenarbeit mit der coalesenses GmbH wurde eine Adaptionsschicht [30] für den Simulator Shawn [22] entwickelt, die es ermöglicht, Programmcode gleichermaßen im Simulator wie auch auf verschiedenen Hardwareplattformen (wie iSense [8], den pacemates [23] oder TelosB) laufen zu lassen. In ähnlicher Weise wurde der Simulator OMNeT++, der in 4.4.2. vorgestellt wird, erweitert.

4.4.1. SpyGlass

Um Sensornetze auch bei ihrer Arbeit beoachten zu können, wurde im Rahmen des Projekts ein Werkzeug zur Visualisierung des Zustandes von Sensor-/Aktor-Netzen entwickelt, das den Namen SpyGlass trägt [4]. Seine Aufgabe besteht darin, Einblicke in die (Fehl-)Funktionsweisen von Sensor-/Aktor-Netzen zu geben.

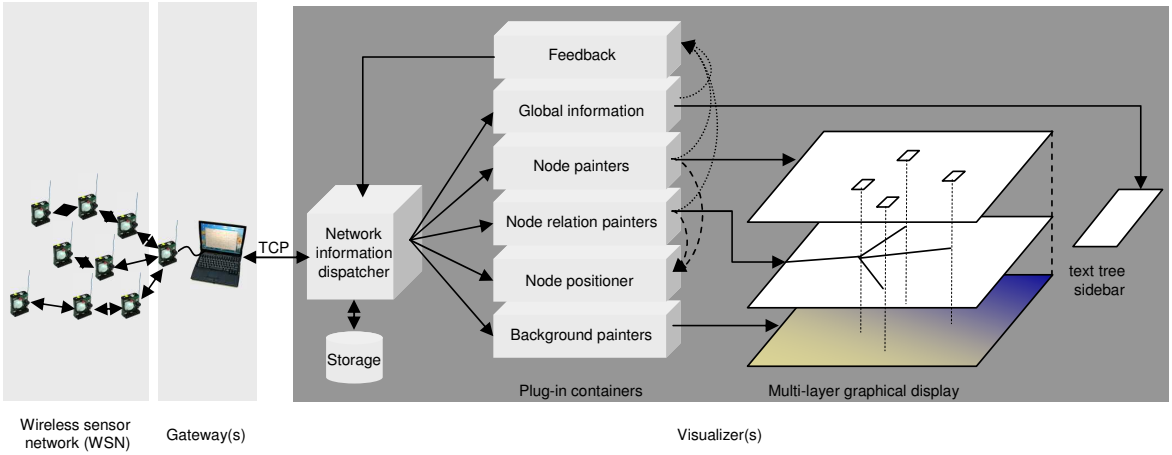


Abb. 4. Architektur von SpyGlass.

Die Architektur von SpyGlass ist in Abbildung 4 dargestellt. SpyGlass besteht aus drei wesentlichen Funktionskomponenten: dem Sensornetz, Gateways und Visualisierungskomponenten. Die Sensorknoten sammeln Informationen und leiten diese zum Gateway weiter, das die Daten sammelt. Die Visualisierungskomponente kann nun TCP-Verbindungen zu einem oder mehreren Gateways herstellen, die dann die Daten an sie weiterleiten. Dort werden die Daten an verschiedene Plugins weitergeleitet, die sie interpretieren und auf einem mehrlagigen Panel darstellen. Auf diese Weise ist es möglich, vielfältige Sachverhalte in Sensornetzen schnell und einfach zu visualisieren.

4.4.2. Simulation mit OMNeT++

Neben dem o.g. Simulator SHAWN, der von Anfang an zur Simulation von WSNs entwickelt wurde, wurde mit OMNeT++ [36] eine zweite, bereits etablierte Simulationplattform gewählt. OMNeT++ bietet neben einem modularen Aufbau eine kommandozeilenbasierte und eine grafische Benutzerschnittstelle. Während erstere auf geringen Ressourcenverbrauch ausgerichtet ist, bietet die grafische Oberfläche einen umfassenden und intuitiven Einblick in die Funktion des simulierten Netzes und den Zustand des Simulators. So lassen sich die Bewegungen der Knoten oder das Versenden und Empfangen von Nachrichten bis hinein in die Softwaremodule eines simulierten Knotens verfolgen.

Da OMNeT++ von Hause aus keine spezielle Unterstützung für WSNs mitbringt, nutzen wir die Mobility Framework [13] Erweiterung, die einige grundlegende Module wie z.B. verschiedene Funk- oder Bewegungsmodule mitbringt. Darauf aufbauend wurde im Rahmen des SWARMS-Projektes die *Simulator Extension for Operations Environment Models* (SEE, [16, 17]) entwickelt, welche wichtige Grundlagen für die Simulation von WSN schafft. Diese lassen sich in drei Bereiche unterteilen:

1. Bereitstellung von realistischen Sensordaten in der Simulation
2. Einführung einer Abstraktionsebene zur einfachen Portierung des Codes zwischen Simulator und realen Knoten
3. Spezialisierte Simulationmodelle

Insbesondere für inhaltsbasierte Ansätze im Bereich von WSNs ist es essentiell, realistische Eingabedaten bereitzustellen, da diese Grundlage für eine Reihe von Entscheidungen der Knoten sind. Zu diesem Zweck wurde ein Modul entwickelt, welches zweidimensionale Daten, wie z.B. Temperaturverteilungen, verwaltet und den Knoten entsprechend ihrer Position abfragbar macht. So ist es z.B. möglich, Bewegungsalgorithmen zu untersuchen, bei denen Knoten Regionen mit relevanten Messwerten bevorzugt anfahren. Abbildung 5 zeigt den Vergleich zwischen einer Sensordatenerfassung mit Hilfe mobiler Sensorknoten, die zum einen auf dem Random Waypoint Modell und zum anderen auf dem Smart Random Waypoint Modell basiert. Bei letzterem wählen die Knoten den nächsten Wegpunkt bevorzugt in Bereichen mit relevanten Daten (dunklere Bereiche), was zu einer deutlich schnelleren Erfassung des interessanten Gebiets führt.

Ähnlich wie auch in SHAWN, bietet die SEE eine sog. *Hardware Abstraction Layer* (HAL, [15]), welche einer Applikation alle notwendigen Schnittstellen zur Verfügung stellt. Da diese HAL sowohl im Simulator als auch auf einer späteren Zielplattform bereitgestellt werden kann, ist es möglich, eine Applikation ohne Veränderungen am Quelltext zu portieren. Die SEE-HAL wird zudem in Form eines XML-Dokumentes spezifiziert, welches neben der eigentlichen Schnittstellenbeschreibung zusätzliche Informationen für die einzelnen Plattformen enthalten

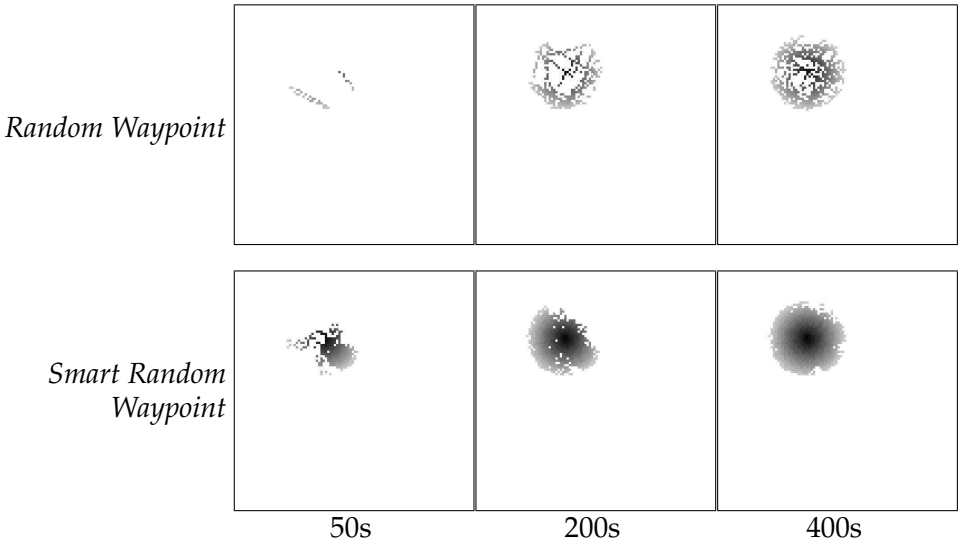


Abb. 5. Random Waypoint vs. Smart Random Waypoint

kann. So ist es z.B. möglich, Sensordatenquellen für die Simulation festzulegen. Ein Generator erstellt aus dieser XML-basierten Beschreibung eine API und z.B. entsprechende Sensorkonfigurationen für die Simulation.

Da jede Simulation auf Modellen basiert, die die eigentlichen Phänomene der realen Welt lediglich vereinfacht widergeben, erreicht eine Simulation nur eine begrenzte Genauigkeit. Ein wichtiger Faktor für diese Abbildungstreue ist daher die Wahl der Modelle. Um bestimmten Gegebenheiten Rechnung zu tragen, wie sie z.B. im WiSeBEES Experiment anzutreffen waren (siehe Abschnitt 4.6.), wurden außerdem ein Bewegungs- und ein Funkmodell für urbane Räume entwickelt und in OMNeT++ integriert [19]. Es hat sich in Simulationen gezeigt, dass insbesondere das urbane Bewegungsmodell, bei dem Knoten Straßenzügen folgen, erheblichen Einfluss auf die Konnektivität und damit auf das Verhalten des Netzes hat [20].

4.5. Quality of Service

Das Thema Dienstgüte (QoS) ist für Sensornetze bisher kaum erforscht worden. Das liegt zum einen daran, dass sich die Arbeitsgruppen bisher darauf konzentrieren mussten, in Anbetracht der ressourcenarmen Knoten überhaupt erst einmal Kommunikation und Datenhaltung zu realisieren und zum anderen daran, dass in solchen Umgebungen Dienstgüte als eine zu große Herausforderung erscheint.

Die wesentliche Beobachtung besteht darin, dass sich die traditionellen QoS-Anforderungen nicht auf Sensornetze übertragen lassen. Begriffe wie Bandbreite oder Jitter spielen in Sensornetzanwendungen keine Rolle.

Vielmehr geht es beispielsweise darum, die Lebenszeit eines Sensornetzes zu verlängern bzw. zu maximieren oder einen Mindestgrad für die Zuverlässigkeit eines Datenwerts einzuhalten oder auch die Übertragung von Nachrichten zuverlässig zu machen. In diesem Sinne stehen folglich auf der einen Seite eines Dienstgüte-Modells für Sensornetze *Anwendundungsanforderungen*, die eingehalten werden sollen. Es sei angemerkt, dass es nicht Ziel der hier vorgestellten Arbeiten war, harte Garantien einhalten zu können, da dieses ohnehin aufgrund der drahtlosen Datenübertragung nicht möglich ist. Vielmehr war es das Ziel, die Anwendungsanforderungen einerseits, bestimmte *Rahmenbedingungen* wie Ressourcenbeschränkungen etc. andererseits sowie die Abhängigkeiten zwischen diesen Größen zu modellieren. Dabei stehen Anforderungen und Rahmenbedingungen häufig nicht direkt, sondern über mehrere zwischengeschaltete Eigenschaften in Verbindung.

Da es aufgrund seiner Komplexität nicht möglich ist, das gesamte Modell hier vorzustellen, soll hier in Ausschnitten die Funktionsweise am Beispiel der Datenweiterleitung beschrieben werden.

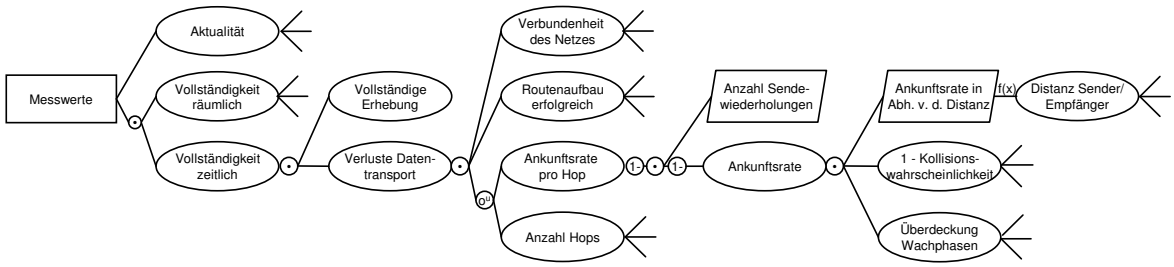


Abb. 6. Ausschnitt des Dienstgütemodells.

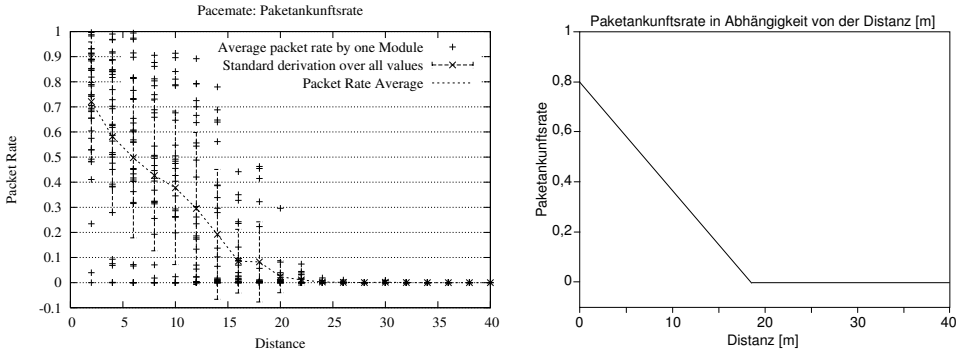
Im Modell (Abbildung 6) werden die Anwendungsanforderungen als Rechtecke dargestellt, die Rahmenbedingungen als Parallelogramme. Letztere stellen dabei nicht zwingend Unveränderliche dar, sondern beispielsweise auch Folgen der Wahl der Funktechnik oder der Hardwareplattform. Sie sind im Allgemeinen verbunden durch unterschiedliche Zwischengrößen (Ovale), die wiederum untereinander wie auch mit Anforderungen und Rahmenbedingungen durch mathematische Vorschriften in Zusammenhang stehen. Diese Vorschriften sind in den Kreisen angegeben, und können z.B. multiplikativer (•), exponenzieller (o^u bedeutet „obere Eingangsgröße hoch untere“) oder funktionaler ($f(x)$) Natur sein.

Unterschiedliche Einflussgrößen dieses Modell wurden sowohl mittels Simulationen als auch experimentell untersucht und überprüft. Dieses soll hier am Beispiel eines Routingverfahrens demonstriert werden, das speziell solche Routen wählt, die über besonders verlustarme Kommunikationsverbindungen verlaufen.

Abweichend von den meisten bekannten Routingverfahren wird nicht die Anzahl der Hops zum Ziel als Routingmetrik verwendet, vielmehr werden die einzel-

nen Hops mit einem Gewicht multipliziert, das die Verlustrate widerspiegelt. Es hat einen hohen Wert, wenn der Link viele Paketverluste aufweist, bzw. einen niedrigen bei hoher Ankunftsrate. Es wird vom Verfahren nun diejenige Route ausgewählt, bei der das Produkt der Gewichte über alle Hops minimal ist. Die Grundidee des Verfahrens sowie Details zur Wahl der Gewichte können [12] entnommen werden. Abweichend vom ursprünglichen Verfahren werden hier solche Routen gewählt, die in beide Richtungen besonders Erfolg verschprechend erscheinen, da bei einer Unicast-Route im Allgemeinen mit bidirektionaler Kommunikation zu rechnen ist.

Die Beurteilung der Paketverlustraten erfolgt mittels einer speziellen Softwarekomponente, dem *Neighborhood Monitor*. Dieser belauscht den gesamten Funkverkehr (d.h. auch solche Pakete, die dediziert an andere Sensorknoten adressiert sind) und verfolgt anhand der auf MAC-Ebene vergebenen Sequenznummern die Paketverlustraten.



(a) Paketankunftsrate in Abhängigkeit von der Distanz.

(b) Shawn-Modell.

Abb. 7. Experimentelle Ergebnisse und Simulatormodellierung der Paketankunftsrate.

Um realistische Ergebnisse erzielen zu können, wurden zunächst bestimmte Rahmenbedingungen des Dienstgütemodells experimentell untersucht, und in entsprechende Modelle für den Simulator Shawn [22] umgesetzt. Abbildung 7 zeigt beispielhaft die Paketankunftsrate in Abhängigkeit von der Distanz, sowie das entsprechend abgeleitete Simulatormodell.

Mit Hilfe dieser Arbeiten konnte gezeigt werden, dass ausgehend von der Untersuchung der Zusammenhänge im Dienstgütemodell die geeignete Berücksichtigung der Rahmenbedingungen bei der Entwicklung von Protokollen deutliche Vorteile bezüglich der Qualität der Diensterbringung von Sensornetzen bietet. So stieg die Auslieferungsrate von Messwerten an das Gateway eines Sensornetzes durch die Verwendung der dienstgütebezogenen Routingmetrik von 42% auf 86%.

4.6. WiSeBEES

Das WiSeBEES Experiment (**Wi**reless **Se**nsornode **B**ased **E**nvironment **E**xploring **S**warm) ist ein Feldexperiment zur Untersuchung des dvSIS-Ansatzes in einem realen Umfeld und zum Erfassen von Referenzdaten, welche zur Evaluation und Unterstützung der Simulation selbst genutzt werden. Das WiSeBEES Experiment besteht aus bis zu 15 mobilen Sensorknoten, die im urbanen Umfeld eine Temperaturkarte aufzeichnen. Dazu werden die gesammelten Daten kontrolliert geflutet. Jeder Knoten bildet so eine lokale Sicht des dvSIS aus, die auf lokal gemessenen und von anderen Knoten erfassten Sensordaten basiert.

Als Sensorknoten wurden modifizierte Linksys WRT54 WiFi Router verwendet, welche über 16MB RAM, 8MB Flash, eine MiPSel CPU mit 200 MHz, ein IEEE 802.11g WLAN und zwei Ethernet Interfaces, einen I^2C-Bus, ein GPS-Modul und einen Temperatursensor verfügen. Als Betriebssystem wird OpenWRT, ein angepasstes Linux, eingesetzt. Diese Plattform hat als Experimentierplattform große Vorteile, da sie selbst im Feldeinsatz z.B. via WLAN und SSH-Login leicht zu überwachen und zu steuern ist. Ferner ermöglicht sie die Aufzeichnung umfangreicher Log-Dateien für die spätere Auswertung. Abgesehen davon ist sie in der Anschaffung deutlich günstiger als vergleichbar ausgestattete dedizierte Sensorknoten. Der Nachteil liegt auf Seiten der Größe und des Stromverbrauchs, was aber für experimentelle Ausbringungen kein Problem darstellt.

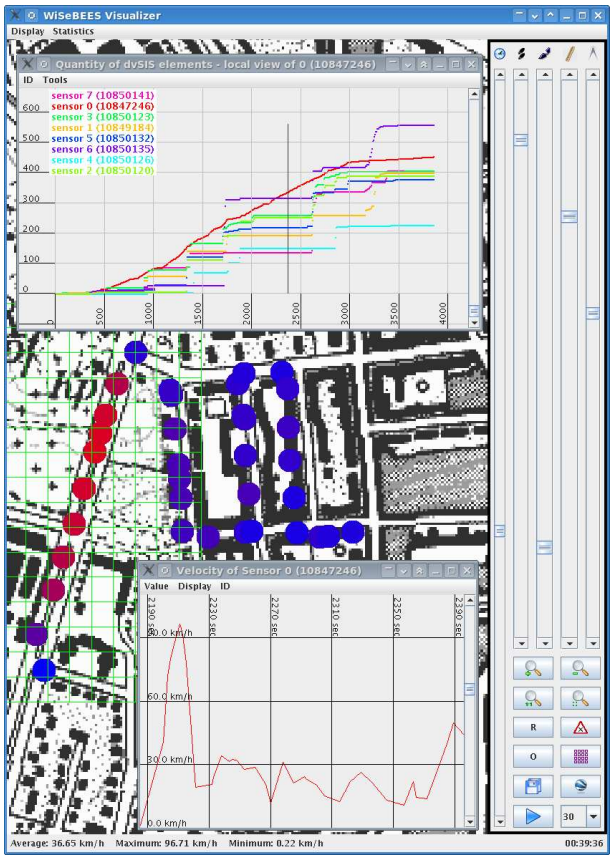
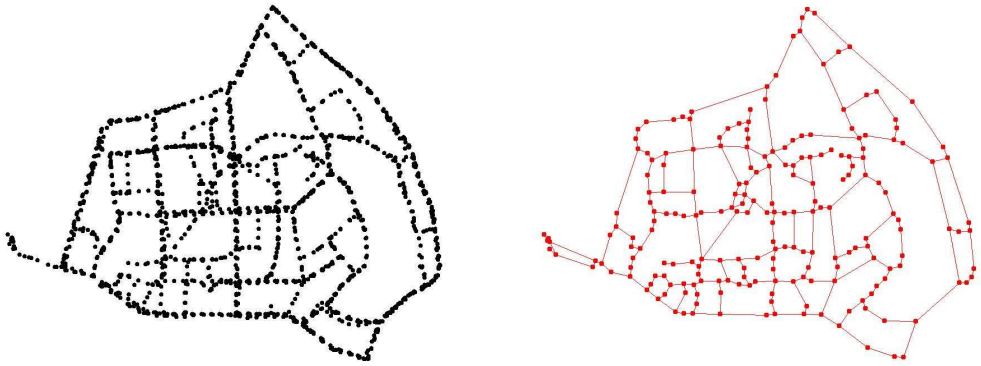


Abb. 8. WiSeBEES Visualisierungswerkzeug

Um die im Feldversuch gesammelten Daten auswerten zu können, wurde ein Visualisierungswerkzeug erstellt, welches die Daten räumlich und zeitlich darstellen kann. Ferner gibt es eine Reihe von statistischen Auswertungsmöglichkeiten, um z.B. Latenzen oder Wertverläufe widerzugeben. Im oberen Teil der Abbildung 8 ist die Größe der lokalen dvSIS-Instanzen dargestellt, welche bei Begegnungen der Sensorknoten sprunghaft

wächst. Im Hauptfenster des Visualisierers können die gesammelten Messwerte nach verschiedenen Kriterien selektiert auf eine Karte projiziert werden. Der untere Teil der Abbildung zeigt den Werteverlauf bzgl. der Geschwindigkeit eines Knotens über die Zeit.

Unabhängig vom dvSIS-Ansatz wurden die gesammelten Daten als Referenz für die Simulation genutzt. Dies betrifft zum einen die korrekte Einstellung der simulierten Funkschnittstelle und zum anderen das genutzte Bewegungsmodell. Für letzteres wurden aus den gesammelten Daten die Positionen der Knoten über die Zeit extrahiert, um aus diesen Wegpunktmengen ein Abbild des Straßennetzes zur erzeugen (siehe Abb. 4.6.). Das Straßennetz wird dabei als Graph repräsentiert und dient der realistischen Bewegung der Knoten in der Simulation, wie in 4.4.2. beschrieben. Mit Hilfe dieses Bewegungsmodells ist es ferner möglich, die Bewegungen der Knoten während eines Feldtests im Simulator nachzubilden. Damit wird es möglich, auch die simulierte Funkschnittstelle so einzustellen, dass sie die in der Realität beobachtete Kommunikation der Knoten bestmöglich widergibt. So ermöglichen die gesammelten Referenzdaten eine Anpassung der Simulationsumgebung an die zu simulierende reale Umgebung.



(a) Von den Knoten im Feldtest aufgezeichnete Wegpunktmenge

(b) Aus Wegpunkten generierter Graph des Straßenmodells

5. Weitere Ergebnisse und Ausblick

Neben den oben beschriebenen Aktivitäten war das Projekt auch in die gemeinsamen Arbeiten des Schwerpunktprogramms eingebunden. Insbesondere mit der Projektgruppe in Rostock konnten im Rahmen eines gemeinsamen Prototypen die verschiedenen in SWARMS erarbeiteten Verfahren und Protokolle praktisch umgesetzt und in einer fremden Umgebung ausgiebig getestet werden. Der Prototyp bestand aus einer größeren Zahl von Sensorknoten, die auf einer großen Tischfläche ausgebreitet wurden und dann mit gegenseitiger Hilfe eine Lokalisierung der Kno-

ten vornahm. Der Prototyp wurde während eines der Kolloquien des Schwerpunktprogramms vorgestellt.

Weiterhin wurde von Mitgliedern des Projekts die coalesenses GmbH gegründet, die heute ein aktives Unternehmen auf dem Gebiet der Sensornetze ist. In die dort entwickelte Hard- und Softwareplattform iSense gingen zahlreiche Ideen und Erkenntnisse aus SWARMS ein.

SWARMS hat ohne Zweifel schon zu einem frühen Zeitpunkt dazu beigetragen, das Wissen über Sensornetze und deren Entwicklung und Betrieb deutlich zu erweitern. Heute gibt es eine Vielazhl von Arbeiten in Deutschland und Europa, die auf diesem Wissen aufbauen und weitere interessante Fragestellungen rund um das Thema Sensornetze bearbeiten.

Literaturverzeichnis

[1] AKYILDIZ, IAN F., W. SU, Y. SANKARASUBRAMANIAM und E. CAYIRCI: *Wireless Sensor Networks: A Survey*. Computer Networks, 38(4):393–422, 2002.

[2] BIRON, P. V. und A. MALHOTRA: *XML Schema Part 2: Datatypes*. http://www.w3.org/TR/xmlschema-2/, May 2001.

[3] BOTTS, MIKE und ALEXANDRE ROBIN: *OpenGIS Sensor Model Language (SensorML)*. OpenGIS Implementation Specification 07-000, Open Geospatial Consortium Inc., 2007.

[4] BUSCHMANN, C., D. PFISTERER, S. FISCHER, S. P. FEKETE und A. KRÖLLER: *SpyGlass: A Wireless Sensor Network Visualizer*. ACM SIGBED Review, 2(1), 2005.

[5] BUSCHMANN, CARSTEN, STEFAN FISCHER, JOCHEN KOBERSTEIN und NORBERT LUTTENBERGER: *Ein Ansatz zur effizienten Softwareentwicklung für drahtlose Sensornetze auf der Basis von Shared Information Spaces*. In: *Systemsoftware für Pervasive Computing, Fachgespräch der GI/ITG-Fachgruppe Kommunikation und Verteilte Systeme*, Stuttgart, 14.-15. Oktober 2004.

[6] BUSCHMANN, CARSTEN, STEFAN FISCHER, JOCHEN KOBERSTEIN und NORBERT LUTTENBERGER: *Towards Information Centric Application Development for Wireless Sensor Networks System Support for Ubiquitous Computing*. In: *Workshop at the Sixth Annual Conference on Ubiquitous Computing (UbiComp 2004)*, Nottingham, England, September 7 2004.

[7] BUSCHMANN, CARSTEN, HORST HELLBRÜCK, STEFAN FISCHER, ALEXANDER KRÖLLER und SANDOR FEKETE: *Radio Propagation-Aware Distance Estimation Based on Neighborhood Comparison*. In: *Proceedings of the 14th European conference on Wireless Sensor Networks (EWSN 2007), Delft, The Netherlands*, Januar 2007.

[8] BUSCHMANN, CARSTEN und DENNIS PFISTERER: *iSense: A Modular Hardware and Software Platform for Wireless Sensor Networks*. Technischer Bericht, 6. Fachgespräch Drahtlose Sensornetze der GI/ITG-Fachgruppe Kommunikation und Verteilte Systeme, 2007.

[9] BUSCHMANN, CARSTEN, DENNIS PFISTERER und STEFAN FISCHER: *Estimating Distances Using Neighborhood Intersection*. In: *11th IEEE International Conference on Emerging*

Technologies and Factory Automation, Prague, Czech Republic, September 2006.

[10] BUSCHMANN, CARSTEN, CHRISTIAN WERNER, HORST HELLBRÜCK und STEFAN FISCHER: *NIDES: Ein Verfahren zur Multihop-Distanzschätzung mittels Nachbarschaftsanalyse*. In: *Proceedings of the 15. ITG/GI - Fachtagung Kommunikation in Verteilten Systemen (KiVS)*, 2007.

[11] CARRIERO, NICHOLAS und DAVID GELERNTER: *Linda in Context*. Communications of the ACM, 32(4):444–458, 1989.

[12] CHHABRA, JASMEET und BRENT ELLIOTT: *Real-World Experiences with an Interactive Ad Hoc Sensor Network*. In: *ICPPW '02: Proceedings of the 2002 International Conference on Parallel Processing Workshops*, Seite 143, Washington, DC, USA, 2002. IEEE Computer Society.

[13] DRYTKIEWICZ, W., S. SROKA, V. HANDZISKI, A. KOEPKE und H. KARL.: *A Mobility Framework for OMNeT++*. In: *3rd International OMNeT++ Workshop 2003*, 2003.

[14] FILIPOVIC, A. und A. DATTA: *Building Blocks of Energy and Cost Effcient Wireless Sensor Networks*. In: *1st European Workshop on Wireless Sensor Networks (EWSN)*, Berlin, 2004.

[15] KOBERSTEIN, JOCHEN und NORBERT LUTTENBERGER: *System-Level WSN Application Software Test using Multi-Platform Hardware Abstraction Layer*. In: *Mobile Ad-hoc and Sensor Networks 2006, Lecture Notes in Computer Science*, Seiten 835–846.

[16] KOBERSTEIN, JOCHEN und NORBERT LUTTENBERGER: *Wireless Sensor Networks in virtueller Umwelt—der SEE-Ansatz für die WSN-Simulation*. In: *4. Fachgespräche Drahtlose Sensornetze, GI/ITG KuVS*, 2005.

[17] KOBERSTEIN, JOCHEN und NORBERT LUTTENBERGER: *Data-centric Wireless Sensor Network Programming Methodology*. In: *3rd International Conference on Networked Sensing Systems (INSS)*, 2006.

[18] KOBERSTEIN, JOCHEN, NORBERT LUTTENBERGER, CARSTEN BUSCHMANN und STEFAN FISCHER: *Shared Information Spaces for Small Devices: The SWARMS Software Concept*. In: *Informatik 2004*, September 2004.

[19] KOBERSTEIN, JOCHEN, HAGEN PETERS und NORBERT LUTTENBERGER: *Graph-Based Mobility Model for Urban Areas Fueled with Real World Datasets*. In: *1st International Workshop on OMNeT++*, Marseille, France, March 2008. ACM.

[20] KOBERSTEIN, JOCHEN, HAGEN PETERS und NORBERT LUTTENBERGER: *Making WSN Simulation More Realistic: a Case Study*. In: *International Workshop on Sensor Network Engineering (IWSNE'08)*, Santorini, Greece, June 2008.

[21] KOBERSTEIN, JOCHEN, FLORIAN REUTER und NORBERT LUTTENBERGER: *The XCast Approach for Content-based Flooding Control in Distributed Virtual Shared Information Spaces–Design and Evaluation*. In: *1st European Workshop on Wireless Sensor Networks (EWSN 2004)*. Springer, January 2004.

[22] KRÖLLER, A., D. PFISTERER, C. BUSCHMANN, S. P. FEKETE und S. FISCHER: *Shawn: A new approach to simulating wireless sensor networks*. In: *Design, Analysis, and Simulation of Distributed Systems 2005, Part of the SpringSim 2005*, April 2005.

[23] LIPPHARDT, MARTIN, HORST HELLBRÜCK, DENNIS PFISTERER, STEFAN RANSOM und STEFAN FISCHER: *Practical Experiences on Mobile Inter-Body-Area-Networking*. In: *Proceedings of the Second International Conference on Body Area Networks (BodyNets'07)*, 2007.

[24] LUTTENBERGER, NORBERT und JOCHEN KOBERSTEIN: *Middleware and Simulation Support for WSN Motion Control Agorithm Design and Analysis*. In: *IEEE International Conference on Distributed Computing in Sensor Systems*, San Francisco, California, USA, June 18 - 20 2006.

[25] LUTTENBERGER, NORBERT, FLORIAN REUTER und JOCHEN KOBERSTEIN: *XML Language Binding Support for Pervasive Communication in Distributed Virtual Shared Information Spaces*. In: *PerWare (Middleware Support for Pervasive Computing Workshop at the 2nd Conference on Pervasive Computing*, Orlando, Florida, March 2004.

[26] MADDEN, SAMUEL R., MICHAEL J. FRANKLIN, JOSEPH M. HELLERSTEIN und WEI HONG: *TinyDB: An Acquisitional Query Processing System for Sensor Networks*. ACM Trans. Database Syst., 30(1):122–173, 2005.

[27] MARTINEZ, KIRK, ROYAN ONG und JANE HART: *Glacsweb: a sensor network for hostile environments*. In: *Proceedings of The 1st IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, 2004.

[28] MATTERN, FRIEDEMANN und KAY RÖMER: *Drahtlose Sensornetze*. Informatik Spektrum, 26(3):191–194, 2003.

[29] NA, ARTHUR und MARK PRIEST: *OpenGIS Sensor Observation Service*. OpenGIS Implementation Specification 06-009r5, Open Geospatial Consortium Inc., 2007.

[30] PFISTERER, DENNIS und CARSTEN BUSCHMANN: *Coalescing simulation and embedded WSN application development*. In: *International Workshop on Sensor Network Engineering (IWSNE'08)*, 2008.

[31] REUTER, FLORIAN, JOCHEN KOBERSTEIN und NORBERT LUTTENBERGER: *XCast-Kommunikation im Virtual Shared Information Space*. PIK special issue on Mobile Ad-Hoc-Networks, December 2003.

[32] SUSANNE HUPFER, ERIC FREEMAN UND KEN ARNOLD UND: *JavaSpaces Principles, Patterns, and Practice*. Addison-Wesley Longman Ltd., Essex, UK.

[33] SZEWCZYK, ROBERT, ERIC OSTERWEIL, JOSEPH POLASTRE, MICHAEL HAMILTON, ALAN MAINWARING und DEBORAH ESTRIN: *Application driven systems research: Habitat monitoring with sensor networks*. Communications of the ACM Special Issue on Sensor Networks, 2004.

[34] THOMPSON, H. S., D. BEECH, M. MALONEY und N. MENDELSOHN: *XML Schema Part 1: Structures*. W3C, http://www.w3.org/TR/xmlschema-1/, May 2001.

[35] TURAU, V., M. WITT und C. WEYER: *Analysis of a Real Multi-Hop Sensor Network Deployment: The Heatland Experiment*. In: *3rd International Conference on Networked Sensing Systems (INSS)*, 2006.

[36] VARGA, A.: *The OMNeT++ Discrete Event Simulation System*. In: *Proceedings of the European Simulation Multiconference (ESM)*, 2001.

[37] WYCKO, P.: *T Spaces*. IBM Systems Journal, 37(3), 1998.

DFG Schwerpunktprogramm 1140

Mit dem Titel "Basissoftware für selbstorganisierende Infrastrukturen für vernetzte mobile Systeme" vereint das Schwerpunktprogramm 1140 der DFG Forschungsvorhaben hoher Aktualität. Es widmet sich der Thematik drahtloser Ad-hoc- und Sensornetze, zwei aufstrebende Techniken zur Ausbringung drahtloser Netzwerke, welche in naher Zukunft durch neuartige Anwendungen einen maßgeblichen Einzug in das alltägliche Leben halten werden. Durch die Konzeption höherwertiger Dienste für diese Netze leistet das Schwerpunkt-programm 1140 einen essentiellen Beitrag zur und Fortschritt in der aktuellen Forschung und erschafft gleichzeitig ein solides Fundament zur Förderung und Entwicklung zahlreicher Anwendungen.

ISBN: 978-3-86644-309-9

www.uvka.de