Basisssoftware für drahtlose Ad-hoc- und Senornetze
Martina Zitterbart, Peter Baumung (Hrsg.)

Basissoftware für drahtlose Ad-hoc- und Sensornetze
Basissoftware für drahtlose Ad-hoc- und Sensornetze

Martina Zitterbart
Peter Baumung
(Hrsg.)
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!


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 Buschmann¹, Stefan Fischer¹, Norbert Luttenberger² und Jochen Koberstein²  
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


1. Ziele des Gesamtprojekts


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

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

Die Projektdauer gliederte sich in drei Bewilligungsphasen, die im Folgenden näher beschrieben werden. Die erste Projektphas e 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 Netzwerkstelefonie (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.


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-


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


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
3. Messung von Dienstgütemerkmalen

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


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


3. Messung von Dienstgütemerkmalen

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 \( N_i \) und \( N_{i+1} \) abgebildet. Für die Stabilitätsprognose sind hierbei insbesondere die Knoten relevant, die sich in der Schnittmenge \( B_{i,i+1} = N_i \cap N_{i+1} \) 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 \textit{Brückenknoten} genannt. Simulationsergebnisse haben gezeigt, dass die Stabilität einer Verbindung mit der Anzahl der Brückenknoten pro Link zunimmt.

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

\[
S(n_0, n_l) = \frac{1}{l^2} \sum_{i=0}^{l-1} |B_{i,i+1}|
\]

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-
Ad-hoc Service Management für selbstorganisierende vernetzte mobile Systeme

Abb. 4: Prinzip der Brückenknoten

Abb. 5: Stabilitätsgewinn in verschiedenen Szenarien

### 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.

![Konzept der knotenbasierten Bandbreitenmessung](image)

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 zeigt Abbildung 6. Dabei wird vorausgesetzt, dass alle Knoten den Übertragungskanal permanent überwachen. So können sie die lokale Auslastung, also den Zeitausschnitt, 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 errechnen, die der Datenrate entspricht, die er einem zusätzlichen Datenfluss zur Verfügung stellen kann.


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 geeignet berechnungsverfahrens konnte die Genauigkeit der Bandbreitenmessung deutlich verbessert werden, wie die Streudiagramme in Abbildung 7 belegen. In diesem Diagramm stellt jeder Punkt die gemessene Bandbreite der tatsächlichen Bandbreite in einem einzelnen Simulationslauf gegenüber. Die Punkte gruppieren sich offensichtlich nah an der eingezeichneten Ideallinie. Auch die angeführten Korrelationskoeffizienten von über 0,98 zeigen, dass dieses Messverfahren brauchbare Ergebnisse liefert.

Abb. 7: Genauigkeit der Bandbreitenmessung
Die Dynamik eines mobilen Ad-hoc-Netzwerks beeinflusst auch maßgeblich die verfügbaren Bandbreiten, so dass diese erheblichen Schwankungen unterliegen 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 vorangegangenen 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.


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 mitgeführten 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 $t_f^i$, zu dem $n_i$ zum ersten mal genannt wurde,
- den Zeitpunkt $t_l^i$, 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 $t_l^i$ 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 $t_l^i - t_f^i$ und der Aktualisierungsrate $u_i = c_i / (t_l^i - t_f^i)$ bewertet. Um stabile von instablen Knoten zu unterscheiden, werden anschließend Knoten mit besonders langer Lebensdauer und hoher Aktualisierungsrate statistisch identifiziert.
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 Erkennungsraten in einem Simulationsexperiment mit fünf stationären und 45 mobilen Knoten auf einer 1000 m x 1000 m großen Fläche. Die obere Linie markiert dabei den durchschnittlichen Prozentsatz der richtig erkannten fixen Knoten. Offensichtlich steigt 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 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


6. Zusammenfassung

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 Netzwerk knoten 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 Adressierungskonzept entwickelt werden, bei dem der Simulator die Multicast-Pakete per Unicast an die endgültigen Empfänger weiterleitet [10].


6. Zusammenfassung

Literaturverzeichnis


Kooperative schichtenübergreifende Überlastkontrolle für mobile Ad-Hoc-Netzwerke

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


1. Einführung

 Kooperative schichtenübergreifende Überlastkontrolle für mobile Ad-Hoc-Netzwerke


2. Implizite Überlastkontrolle: CXCC


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


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-


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.


Die Idee, die BarRel zugrunde liegt, nutzt CXCCs Limitierung der Zahl von wischengespeicherten 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 $i$-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!


Der erste und offensichtliche Ansatz ist der Versand eines einzelnen Bestätigungs paketes 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 Kommunikationsfluss in Gegenrichtung möglich.


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-
5. Koordiniertes Network Coding: noCoCo


Kooperative schichtenübergreifende Überlastkontrolle für mobile Ad-Hoc-Netzwerke


Abbildung 5. Durchsatz mit noCoCo in bidirektionaler Kettentopologie.

Abbildung 6. Protokolloverhead mit noCoCo in bidirektionaler Kettentopologie.
6. Evaluation schichtübergreifender Protokolle


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.

8. Zusammenfassung


Literatur


Kooperative schichtenübergreifende Überlastkontrolle für mobile Ad-Hoc-Netzwerke

1. Einführung und Projektziele


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


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 m² mit Fußgängerge-

2. Modulare P2P-basierte Gruppendienste

In diesem Abschnitt werden ausgewählte Konzepte, die im Rahmen des MA-MAS-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


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

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


– **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 Ereignis-basierter 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 Gruppenbeitrags 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.


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 entgegenzuwirken, bietet es sich an, die Broadcast-Fähigkeit des geteilten Mediums gezielt zur Datenübermittlung zu nutzen, wie beispielsweise anhand des in diesem Projekt entwickelten Konzepts Lokaler Broadcast Cluster (LBCs [2, 13]).


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
2. Modulare P2P-basierte Gruppendienste

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.


2.3.2. Ausfallerkennung von Gruppenmitgliedern


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

<table>
<thead>
<tr>
<th>Gruppenmitglieder</th>
<th>Aufgebauten Routen [1/s]</th>
<th>Kein Datenverkehr</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.5</td>
<td>0.1</td>
</tr>
<tr>
<td>10</td>
<td>1.0</td>
<td>0.2</td>
</tr>
<tr>
<td>20</td>
<td>1.5</td>
<td>0.3</td>
</tr>
<tr>
<td>30</td>
<td>2.0</td>
<td>0.4</td>
</tr>
<tr>
<td>40</td>
<td>2.5</td>
<td>0.5</td>
</tr>
<tr>
<td>50</td>
<td>3.0</td>
<td>0.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gruppenmitglieder</th>
<th>Medienzugriffszeit [s/s]</th>
<th>Kein Datenverkehr</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.05</td>
<td>0.1</td>
</tr>
<tr>
<td>10</td>
<td>0.1</td>
<td>0.2</td>
</tr>
<tr>
<td>20</td>
<td>0.15</td>
<td>0.3</td>
</tr>
<tr>
<td>30</td>
<td>0.2</td>
<td>0.4</td>
</tr>
<tr>
<td>40</td>
<td>0.25</td>
<td>0.5</td>
</tr>
<tr>
<td>50</td>
<td>0.3</td>
<td>0.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gruppenmitglieder</th>
<th>Zustellrate [%]</th>
<th>CBR-Anwendung (4 x 512 Bytes/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>80</td>
<td>88</td>
</tr>
<tr>
<td>10</td>
<td>85</td>
<td>89</td>
</tr>
<tr>
<td>20</td>
<td>90</td>
<td>92</td>
</tr>
<tr>
<td>30</td>
<td>95</td>
<td>94</td>
</tr>
<tr>
<td>40</td>
<td>100</td>
<td>98</td>
</tr>
<tr>
<td>50</td>
<td>100</td>
<td>99</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gruppenmitglieder</th>
<th>Zustellrate [%]</th>
<th>Chat-Anwendung</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>80</td>
<td>88</td>
</tr>
<tr>
<td>10</td>
<td>85</td>
<td>89</td>
</tr>
<tr>
<td>20</td>
<td>90</td>
<td>92</td>
</tr>
<tr>
<td>30</td>
<td>95</td>
<td>94</td>
</tr>
<tr>
<td>40</td>
<td>100</td>
<td>98</td>
</tr>
<tr>
<td>50</td>
<td>100</td>
<td>99</td>
</tr>
</tbody>
</table>
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 Gruppenmitgliedsausfall 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
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 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-
2. Modulare P2P-basierte Gruppendifenste

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


2.5. Praktische Umsetzungen und Demonstratoren


3. Zusammenfassung


Literaturverzeichnis


PBM – Positionsbasierter Multicast für mobile Ad-Hoc-Netze

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


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.


2. Das Position-Based Multicast Routing Protocol


3. Scalable Position-Based Multicast Routing


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 Quadrat 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.

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 Zielquadrat e, so haben die Knoten dort dank der hierarchischen Gruppenverwaltung nähere Information über die Verteilung der Empfänger innerhalb dieses Quadrates und können die Ziellinformationen 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 variable 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.

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.


4. Contention-Based Multicast Forwarding

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


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-
5. Implementierung


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 Sendewarteschlangen 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 Sendewarteschlangen 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

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

6. Greedy Routing with Abstract Neighbor Table


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,
6. Greedy Routing with Abstract Neighbor Table

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-Strategie 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)).

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 Basisalgorith-


Greedy Routing with Abstract Neighbor Table

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.

Abb. 6. Varianten der Abstract Neighbor Table

PBM – Positionsbasierter Multicast für mobile Ad-Hoc-Netze

verwendet werden, um die Position des Repräsentanten zu bestimmen. Sollte der Abstand zwischen dem Repräsentant und \( D \) kleiner sein als \( \text{metricVal} \), so wird der Abstand in \( \text{metricVal} \) 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 \( ED \) 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

\[
\text{Algorithmus 1: Berechnung des Eignungswerts eines potentiellen Weiterleiters}
\]

\[
\text{if } E = D \text{ then}
\]
\[
\text{  } \_ \text{ return } -\infty
\]
\[
\text{metricVal } \leftarrow d(E, D)
\]
\[
\text{maxLen } \leftarrow \text{metricVal}
\]
\[
\text{for } h \in [0, a - 1] \text{ do}
\]
\[
\text{  if } \text{ANT}[h].encLen > 0 \text{ then}
\]
\[
\text{    len } \leftarrow \text{decEstLen}(\text{ANT}[h].encLen)
\]
\[
\text{    ang } \leftarrow \text{decEstAngle}(h, \text{ANT}[h].encAngle)
\]
\[
\text{    if } \text{maxLen } < \text{len} \text{ then}
\]
\[
\text{      len } \leftarrow \text{maxLen}
\]
\[
\text{    len } \leftarrow \text{len } \ast \text{DISCOUNT}
\]
\[
\text{    NH}_h \leftarrow \text{getEstNGPos}(E, ang, len)
\]
\[
\text{    if } d(\text{NH}_h, D) < \text{metricVal} \text{ then}
\]
\[
\text{      } \_ \_ \text{metricVal } \leftarrow d(\text{NH}_h, D)
\]
\[
\text{return } \text{metricVal}
\]

7. Zusammenfassung


Literaturverzeichnis


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


PBM – Positionsbasierter Multicast für mobile Ad-Hoc-Netze
Selbstkonfiguration adaptiver Anwendungen in ubiquitären Systemen

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


1. Einleitung


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
Selbstkonfiguration adaptiver Anwendungen in ubiquitären Systemen

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 an-
passen.

Abb. 1. Adaption einer beispielhaften Messaging-Anwendung

Das hier beschriebene Forschungsprojekt beschäftigte sich mit der Konfigurati-
on und Adaption verteilter Anwendungen. Von besonderer Bedeutung hinsicht-
lich 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 kön-
nen. Um dieses Ziel zu erreichen, mussten neue Algorithmen entwickelt werden,
welche sowohl die zur Verfügung stehenden Ressourcen als auch die Anforderun-
gen der Anwendungen sowie Nutzerziele berücksichtigen.

Im Rahmen dieses Projekts sollte außerdem sowohl die Heterogenität der Geräte
als auch die Heterogenität der Kommunikationstechnologien berücksichtigt wer-
den. Das Gesamtziel dieses Projekts war es, unterstützende Systemsoftware zu kon-
2. Stand der Forschung bei Projektbeginn


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.


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.


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 Gesamtförderdauer des Projekts vier Jahre.


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


Die entsprechend entwickelten Konzepte und Mechanismen werden in den folgenden Abschnitten detaillierter diskutiert.
Selbstkonfiguration adaptiver Anwendungen in ubiquitären Systemen

3.1. Entwicklung eines Algorithmus zur initialen Konfiguration von Anwendungen


3. Wissenschaftliche Fortschritte


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.


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 Rechnungskosten für eine neue Konfiguration ergab sich die Forderung, dass diese im Adapptionsfall die Kosten für die Berechnung einer initialen Konfiguration nicht übersteigen sollten. Daher konnten die Adapptionskosten nur heuristisch optimiert werden, weil die Berechnung einer optimalen Anwendungskonfiguration im Allgemeinen die Berechnung aller möglichen Anwendungskonfigurationen voraus-
Selbstkonfiguration adaptiver Anwendungen in ubiquitären Systemen

setzt. Auf der Grundlage des Kostenmodells für die Migration von Anwendungs- 
zuständen wurde eine einfache Greedy-Heuristik in den bestehenden Konfigurations- 
algorithmus integriert, welche Adaptionen gemäß dem im Bereich der Bedings- 
erfüllungsprobleme häufig verwendeten Value Ordering-Verfahren durchführt. Bei 
diesem Verfahren wird derjenige Wert einer Variablen als nächstes gewählt, wel- 
cher die meisten Optionen für folgende Variablen offen läßt, wodurch die Effizienz 
von Adaptionen durch eine geringere Anzahl nötiger Adaptionen deutlich 
verbessert wird. In Evaluationen wurden dementsprechend in den meisten Szenari- 
en gute Ergebnisse mit geringen Adaptionsschritten 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 Quali- 
itätskriterien bei der automatischen Konfiguration und Adaption von Anwendun- 
gen durchgeführt. Hierdurch wurde der zuvor entwickelte Algorithmus erweitert, 
um mögliche Konfigurationen hinsichtlich vorgegebener Benutzerpräferenzen zu 
bewerten.

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

Ausgehend von diesen Werkzeugen sowie der bereits bestehenden Modellierung 
nicht-funktionaler Eigenschaften von Komponenten wurde dann ein Modellier- 
ungskonzept entwickelt, mit welchem Anwender ihre Präferenzen bezüglich unter- 
schiedlicher 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-
3. Wissenschaftliche Fortschritte

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 aufgespaltet 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 AssemblerImplementierungen 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-
Selbstkonfiguration adaptiver Anwendungen in ubiquitären Systemen

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 Konfigurationslatzen führen können.

3.5. Entwicklung eines effizienten Algorithmus für heterogene Umgebungen


Die proaktive Adoptionsvermeidung 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 Adoptionsvorgänge 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 Adptionsvorgang angepasst werden. Hierbei muss berücksichtigt werden, dass die Adaption einer Komponente zusätzlich die Adaption anderer Komponenten nach sich ziehen kann, um weiterhin sämtliche strukturelle 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 Adoptionsaufwands 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 signifikant 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 Konfigurationssalgorithmus 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 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-
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.

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 Grad 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 durchführte.

3.8. Einbindung der Ergebnisse voriger Konfigurationsvorgänge in den Konfigurationsprozess


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 Teilkonfigurationen per Broadcast an die Geräte in der Umgebung gesendet.

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

Selbstkonfiguration adaptiver Anwendungen in ubiquitären Systemen

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.


Literaturverzeichnis


4. Zusammenfassung und Ausblick


Selbstkonfiguration adaptiver Anwendungen in ubiquitären Systemen
COCOS – Coordinated Communicating Sensors

Reinhardt Karnapke und Jörg Nolte
Brandenburgische Technische Universität Cottbus


1. Einleitung


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

2. Das COCOS Projekt


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.

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

2.1. Cocos


Es wurden diverse Objekträumekonzepte untersucht, implementiert und evaluiert [12] [9] [10]. Ein wesentliches Untersuchungskriterium war die topologische Stabilität der Objekträume in Abhängigkeit von den vorstellbaren Anwendungsszen-
Szenarien. Die topologische Stabilität eines Objektraumes ist im wesentlichen abhängig von der Stabilität der Mitgliedschaft (der Häufigkeit der Beim- 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 Änderungsraten des jeweiligen physikalischen Phänomens.

Objekträume mit einer stabilen Mitgliederzahl und einer stabilen 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.


2.2. Chips

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

2.3. Copra


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


### 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.


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

**Abb. 3. Ein kombiniertes WSN/MANET**
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


2.7. Ein netzübergreifendes Routingprotokoll


Abb. 5. Eine gewichtete Routingmetrik

2.8. Erweiterung des OMNET++ Simulators

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


Literaturverzeichnis

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


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


Middleware für mobile spontan vernetzte Sensornetzwerke

Jan Blumenthal und Dirk Timmermann

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


1. Ausgangslage und Motivation


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


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].


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.


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.


5. Die **Energieeffizienz** von Mechanismen, Algorithmen und Protokollen ist entscheidend für die Lebensdauer des Netzwerkes.


### 3.2. Design von Middleware- und Dienstarchitekturen


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-
Middleware für mobile spontan vernetzte Sensornetzwerke


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


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
3. Erzielte Ergebnisse


3.4. Entwicklung von Algorithmen zur dezentralen Lokalisierung von ressourcenarmen Sensorskonten

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, Speichergöße und Kommunikationsaufwand) hohe Anforderungen stellen und zumeist einen erheblichen Energieverbrauch aufweisen.

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 Speichermehraufwand 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 Triilateration erreichen können [1].


3.5. Gruppenbildung in Sensornetzwerken


Nach der Ermittlung der Gruppengrenzen wird durch die Gruppenverwaltung definiert, wie die Datenübertragung erfolgt bzw. wie die Daten aggregiert werden sollen. Diese Verfahren bilden die Voraussetzung zu Untersuchungen der dezentralen Phänomen-detektion 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


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 Software-System, 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-
Middleware für mobile spontan vernetzte Sensornetzwerke

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


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.


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
Ausblick


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


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

Carsten Buschmann¹, Stefan Fischer¹, Norbert Luttenberger² und Jochen Koberstein²

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


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 Schematata 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.


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

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]:


- 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 mit 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 geschickte 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 Netzes 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.


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
Das Grundkonzept von SWARMS ist 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., neue 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 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 Sensus netz 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.

4. Lösungen und Umsetzungen

4.1. dvSIS


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 Methodenrampen 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 beispielsweise um eigene Messergebnisse von nahen Knoten bestätigen zu
4. Lösungen und Umsetzungen

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.


Abb. 1. Nachbarschaftüberschneidung bei verschiedenen Distanzen und Winkeln.


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


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.


4. Lösungen und Umsetzungen

4.3. Sensor Web Enablement


Abb. 3. Architektur des eSOS
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.


4.4. Entwicklung, Test und Evaluation von Software


4.4.1. SpyGlass


Abb. 4. Architektur von SpyGlass.

4.4.2. Simulation mit OMNeT++


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 Bewegungsmodul 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


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
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.


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 Dienstgüte-Modells.

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 (⊙) oder funktionaler (f(x)) Natur sein.

Unterschiedliche Einflussgrößen dieses Modells 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-

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.

![Graphik](image.png)

(a) Paketankunftsrate in Abhängigkeit von der Distanz.  
(b) Shawn-Modell.

*Abb. 7. Experimentelle Ergebnisse und Simulatormodellierung der Paketankunftsrate.*


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


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 I²C-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.

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

![Abb. 8. WiSeBEES Visualisierungswerkzeug](image-url)

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 Straßenetz 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 widerruft. So ermöglichen die gesammelten Referenzdaten eine Anpassung der Simulationsumgebung an die zu simulierende reale Umgebung.

![Graph des Straßenmodells](image)

(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 Vielzahl von Arbeiten in Deutschland und Europa, die auf diesem Wissen aufbauen und weitere interessante Fragestellungen rund um das Thema Sensornetze bearbeiten.

Literaturverzeichnis


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


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 Schwerpunktprogramm 1140 einen essentiellen Beitrag zur und Fortschritt in der aktuellen Forschung und erschafft gleichzeitig ein solides Fundament zur Förderung und Entwicklung zahlreicher Anwendungen.