

Routing, Anycast and Multicast for Mesh and Sensor Networks

Florian Merz

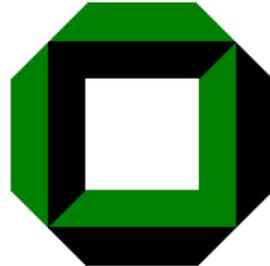
Ausarbeitung zum Seminar „Algorithmen in Sensornetzen“ im SS 07

Institut für theoretische Informatik

Prof. Dr. Dorothea Wagner

Betreuer: Steffen Mecke

1. Oktober 2007



Universität Karlsruhe (TH)

1 Einleitung

Sensornetzen wurde in der letzten Zeit viel Aufmerksamkeit gewidmet, vor allem auch wegen der zahlreichen Anwendungsmöglichkeiten wie zum Beispiel Wildtier-Überwachung, Brandwarnungen oder Umweltüberwachung. Da die Qualität der Netze in den meisten Anwendungsszenarien mit zunehmender Knotenanzahl zunimmt, wird günstige Hardware benötigt was aber eingeschränkte Leistung mit sich bringt, wie zum Beispiel kleine Speicher, langsame Prozessoren und eine begrenzte Energiequelle. Dieser Trend zu größeren Netzen stellt aber auch Anforderungen an die Programmierung der Geräte, es bedarf skalierbarer Algorithmen, die auch auf grossen Netzen performant sind.

In der Veröffentlichung [1] von Flury und Wattenhofer haben die Autoren sich einem der wichtigsten Teile eines solchen begrenzten Netzes zugewendet, dem Informationsaustausch zwischen Netzknoten. In den meisten Fällen ist das einzige Mittel dies zu tun Routing, entweder durch Unicast¹, Anycast², Multicast³ oder Broadcast⁴. Für letzteres gibt es viele Ansätze, jedoch ist es wesentlich herausfordernder ein effizientes Punkt-zu-Punkt Routing, Anycast sowie Multicast zu implementieren. Ein normaler Ansatz für Punkt-zu-Punkt Routing speichert allerdings den optimalen Pfad zu jedem Knoten des Netzes, was $O(n)$ Routing-Einträge mit sich bringt und die Speicherkapazität der einzelnen Knoten in grossen Netzen schnell übersteigt.

Es existierten zwar schon Routingalgorithmen, die gar keine Routingtabellen benötigen, wie zum Beispiel geographisches Routing. Jedoch benötigt zum Beispiel dieses die genaue Position jedes Knotens, und der Weg den eine Nachricht zurück legen muss, ist bis zu quadratisch gegenüber der optimalen Strecke. Auch liegt diesen Algorithmen das *Unit Disk Graph Model* zu Grunde, welches im Allgemeinen zu praxisfern ist.

Der in [1] vorgestellte Algorithmus überwindet diese Probleme und bietet mit kleinen begrenzten Routing-Tabellen nahezu optimales Unicasting und konstante Approximationen für Anycast und Multicast. Das verwendete abstrakte Verbindungsnetz trifft auf fast alle drahtlosen Netze zu, zusätzlich sind die Anforderungen an die einzelnen Knoten minimal: Lediglich eine eindeutige ID und die Kommunikationsmöglichkeit mit den direkten Nachbarn wurden vorausgesetzt.

Die Verbindungen eines jeden Netzwerkes können durch einen Graph beschrieben werden. Berücksichtigt man die begrenzte Reichweite drahtloser Geräte, kann man beobachten, dass Knoten vor allem mit Knoten in ihrer Umgebung verbunden sind, die Anzahl paarweise unabhängiger⁵ Nachbarn ist deshalb im Allgemeinen sehr gering. Diese Eigen-

¹Senden von einem Knoten zu einem anderen Knoten

²Senden von einem Knoten zu einem beliebigen Knoten aus einer bestimmten Knotenmenge

³Senden von einem Knoten zu einer Knotenmenge

⁴Senden von einem Knoten zu allen anderen Knoten des Netzes

⁵Zwei Knoten nennt man unabhängig wenn sie keine direkten Nachbarn sind

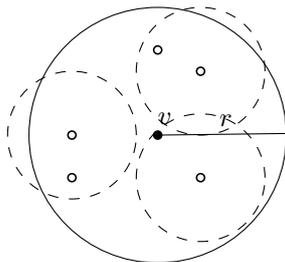


Abbildung 1.1: Kugel $\mathcal{B}_v(r)$ die von drei Kugeln mit Radius $\frac{r}{2}$ abgedeckt wird.

schaft wird zur Multi-Hop Eigenschaft eines Netzes verallgemeinert: Eine gegebene h -Hop Nachbarschaft zu einem Knoten v lässt sich in eine geringe Anzahl von $\frac{h}{2}$ -Hop Nachbarschaften unterteilen, welche alle deren Knoten abdecken.

Die Verbindungsdichte eines Netzes lässt sich durch \mathcal{A} beschreiben, und zwar sei \mathcal{A} die maximale Anzahl an Nachbarschaften halber Größe, die nötig sind um eine beliebige Nachbarschaft abzudecken. Diese Anzahl ist für die meisten drahtlosen Netze sehr klein, meist im niedrigen zweistelligen Bereich. Da \mathcal{A} den Aufbau eines Netzes weitestgehend beschreibt, können diese Erkenntnisse für nahezu alle drahtlosen Netze verwendet werden. Desweiteren ist es nicht nötig, dass der in [1] vorgestellte Routingalgorithmus den Wert von \mathcal{A} kennt, lediglich seine Laufzeit und der Aufwand hängen davon ab.

Im weiteren Verlauf wird ein Netz durch einen gewichteten Graph $G = (V, E)$ dargestellt werden, wobei jeder Knoten des Netzes durch einen Knoten im Graph repräsentiert wird, gewichtete Kanten verbinden jedes Knotenpaar, das direkt miteinander kommunizieren kann. Eine Entfernungsmetrik \mathcal{M} zu G weist über die Kostenfunktion $d_{\mathcal{M}}(a, b)$ jedem Knotenpaar $(a, b) \in V \times V$ die Kosten des kürzesten Pfades zwischen a und b zu.

Definition 1.1 (Kugel) Zu einem Knoten $v \in V$ bezeichnet die Kugel $\mathcal{B}_v(r)$ mit Radius r die Menge der Knoten, deren Entfernung von v höchstens r beträgt: $\mathcal{B}_v(r) = \{u \mid d_{\mathcal{M}}(a, b) \leq r\}$

Definition 1.2 (Constant Doubling Metric) Ein Graph $G = (V, E)$ besitzt die Eigenschaften einer Doubling Metric, wenn jede Kugel $\mathcal{B}_v(r)$ durch eine konstante Anzahl Kugeln $\mathcal{B}_v(\frac{r}{2})$, mit halbem Radius, abgedeckt werden kann. Für $r \geq 0$ und $\forall v \in V: \exists U \subseteq \mathcal{B}_v(r)$, so dass $|U| = O(1)$ und $\mathcal{B}_v(r) \subseteq \bigcup_{u \in U} \mathcal{B}_v(\frac{r}{2})$. Wenn $|U|$ durch 2^α für ein konstantes α begrenzt ist, besitzt die zu G dazugehörige Metrik eine sogenannte Doubling Dimension α .

Für die meisten drahtlosen Netze ist $\alpha = \lceil \log_2 \mathcal{A} \rceil$ ziemlich klein, meist 3 oder 4.

2 Vorangegangene Arbeiten

Es gibt viele Veröffentlichungen zum Thema Routing in drahtlosen Netzen, die sich mit leistungsbegrenzter Hardware beschäftigen. Flury und Wattenhofer haben sich speziell den skalierbaren Routingalgorithmen für leistungsbegrenzte Hardware, mit dem Ziel immer noch eine sehr gute Routing-Leistung zu erreichen. Peleg und Upfal [2] haben in diesem Bereich einen Meilenstein gesetzt, indem sie die Zusammenhang zwischen der Effizienz und dem Speicheraufwand von Routingalgorithmen untersucht haben. Sie stellen einen Routingalgorithmus mit Ausdehnungsfaktor k vor, die durchschnittliche Größe der Routing-Tabellen beträgt hierbei $O(k^3 n^{\frac{1}{k}} \log n)$ Bits, für eine Knoten-Markierung werden $O(\log n)$ Bits benötigt. Knoten-Markierungen sind hierbei Bezeichner für die einzelnen Knoten, die neben einer Identifikation noch weitere Informationen beinhalten, in den Routing-Tabellen sind die einzelnen Routing-Informationen hinterlegt, wie ein Knoten zu erreichen ist.

17 Jahre später stellten Abraham, Gavoille, Goldberg und Malkhi in [5] ein bahnbrechendes Routing Schema vor, sie benötigten $\lceil \log n \rceil$ Bits für die Knoten-Markierungen und je $\varepsilon^{-O(\alpha)} \log n \log(\min(\Theta, n))$ Bits für die Routing-Tabellen.

Der Ansatz von Flury und Wattenhofer [1] erreicht nicht ganz die Schranken aus [5], sie benötigen $O(\frac{1}{\varepsilon})^\alpha (\log \Theta)(O(\alpha) + \log \Delta)$ Bits für die Routing-Tabellen und die Knoten-Markierungen benötigen $2\alpha \lceil \log \Theta \rceil + \lceil \log n \rceil$ Bits, wobei mit Θ der Durchmesser des Netzes bezeichnet wird. Dieser Overhead ermöglicht ihnen jedoch ein Routing Schema vorzuweisen, das neben Unicasting (Abschnitt 5) auch noch konstante Approximationen für Anycasting (Abschnitt 7) und Multicasting (Abschnitt 6) liefert. Zudem beschreiben sie einen verteilten Algorithmus um die Routing-Tabellen sowie die Knoten-Markierungen zu errechnen.

3 Definitionen

Um den Ansatz von Fury und Wattenhofer [1] verstehen zu können werden einige Bezeichnungen und Zustandsbeschreibungen für *Doubling Metriken* benötigt, die wichtigste Definition ist die des ρ -Netzes, welches nahe mit den Maximal Independent Set verwandt ist. Dieses ρ -Netz dient zur Unterteilung des Graphen, um einige Knoten hervorzuheben mit deren Hilfe alle Knoten abgedeckt werden können.

Definition 3.1 (ρ -Netz) *In einem Graph $G = (V, E)$ nennt man eine Teilmenge U einer Knotenmenge V ein ρ -Netz, falls jeder Knoten aus V höchstens eine Entfernung von ρ zu mindestens einem Knoten aus U besitzt, und der paarweise Abstand zweier Knoten aus U stets größer als ρ ist (vgl. Abbildung 3.1).*

Formal: Eine Menge $U \subseteq V$ ist ein ρ -Netz von $G = (V, E)$ wenn gilt:

$$\forall v \in V : \exists u \in U : d_{\mathcal{M}}(v, u) \leq \rho \text{ und}$$

$$\forall u_1, u_2 \in U : d_{\mathcal{M}}(u_1, u_2) > \rho.$$

Ein Knoten u aus einem ρ -Netz deckt alle Knoten in $\mathcal{B}_u(\rho)$ ab, somit ist jeder Knoten des Netzes von mindestens einem Knoten $u \in \rho$ -Netz abgedeckt. Die Knoten eines ρ -Netzes werden von nun an als **Netz-Zentren** des ρ -Netzes bezeichnet.

Ist eine Constant Doubling Metrik gegeben, so ist es für jede Kugel $\mathcal{B}_v(r)$ möglich, alle Knoten in $\mathcal{B}_v(r)$ mit 2^α Kugeln mit halbem Radius abzudecken. Dies liefert jedoch noch keine obere Schranke für die Anzahl an $\mathcal{B}_v(\frac{r}{2})$ die benutzt werden um $\mathcal{B}_v(r)$ abzudecken. Der folgende Satz gibt eine obere Schranke für die Anzahl an Netz-Zentren eines $\frac{r}{2}$ -Netzes die eine Kugel $\mathcal{B}_v(r)$ abdecken kann.

Satz 3.1 (Spärlichkeit) *In einem Graph $G = (V, E)$ mit einer Constant Doubling Dimension α gilt: Für $x \in \mathbb{N}^0$ deckt jede Kugel $\mathcal{B}_v(2^x \rho)$ höchstens $2^{(1+x)\alpha}$ Knoten eines beliebigen ρ -Netzes ab.*

Beweis. Wird die Definition der Doubling Metrik rekursiv angewandt, beobachtet man, dass es möglich ist $\mathcal{B}_v(2^x \rho)$ mit höchstens $2^{(1+x)\alpha}$ Kugeln mit Radius $\frac{\rho}{2}$ abzudecken. Angenommen C sei eine Menge solcher Kugeln die $\mathcal{B}_v(2^x \rho)$ abdeckt, wobei $|C| \leq 2^{(1+x)\alpha}$.

Um einzusehen, dass $\mathcal{B}_v(2^x \rho)$ höchstens $2^{(1+x)\alpha}$ Netz-Zentren $U = \{u_1, u_2, \dots, u_{|U|}\}$ von einem beliebigen ρ -Netz abdeckt, beobachtet man zunächst, dass der paarweise Abstand der Knoten in U stets grösser als ρ ist. Desweiteren ist nach Definition der Abdeckung C jedes $u_j \in U$ von mindestens einer Kugel aus C abgedeckt. Da die Kugeln in C einen Radius von $\frac{\rho}{2}$ haben und der paarweise Abstand zweier u_j stets größer als ρ ist deckt jedes $c \in C$ höchstens ein $u_j \in U$ ab. Daraus folgt: $|U| \leq |C| \leq 2^{(1+x)\alpha}$. \square

Mit Hilfe dieser oberen Schranke für die Anzahl der von einer Kugel abgedeckten Netz-Zentren kann nun eine obere Schranke für die Anzahl an Netz-Zentren eines gegebenen

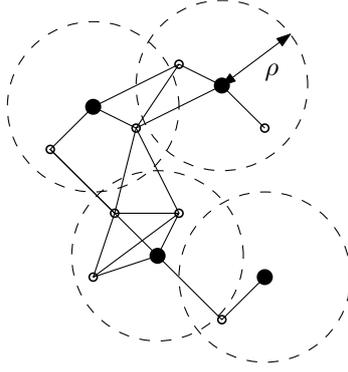


Abbildung 3.1: Die ausgefüllten Knoten bilden hier ein ρ -Netz auf dem abgebildeten Graphen.

ρ -Netzes, die einen beliebigen Knoten v abdecken, angegeben werden. Genauer betrachtet ist die Anzahl der Netz-Zentren eines ρ -Netzes interessant, die einen Knoten v abdecken, wenn jedes Netz-Zentrum eine erweiterte Kugel $\mathcal{B}_u(2^x \rho)$ abdeckt.

Satz 3.2 (Dominanz) *Ist ein ρ -Netz auf einem Graph $G = (V, E)$ mit einer Constant Doubling Metrik α gegeben, wobei jedes Netz-Zentrum $\mathcal{B}_u(2^x \rho)$ mit $x \in \mathbb{N}^0$ abdeckt, so wird jeder Knoten $v \in V$ von höchstens $2^{(1+x)\alpha}$ Netz-Zentren aus dem ρ -Netz abgedeckt.*

Beweis. Falls v von einem Netz-Zentrum u abgedeckt wird, so beträgt der Abstand zwischen u und v höchstens $2^x \rho$. Deshalb ist es hinreichend zu zeigen, dass die Kugel $\mathcal{B}_v(2^x \rho)$ um v höchstens $2^{(1+x)\alpha}$ Netz-Zentren enthält, dies folgt direkt aus Satz 3.1. \square

Korollar 3.1 (Dominanz) *Ist ein ρ -Netz auf einem Graph $G = (V, E)$ mit einer Constant Doubling Metrik α gegeben, wobei jedes Netz-Zentrum $\mathcal{B}_u(i\rho)$ mit $i \in \mathbb{R}^+, i > 1$, abdeckt, so wird jeder Knoten $v \in V$ von mindestens $2^{2\alpha} i^\alpha$ Netz-Zentren aus dem ρ -Netz abgedeckt.*

4 Das Dominanz-Netz

Da die Anforderungen an die einzelnen Knoten minimal waren (einzigartige ID und Kommunikationsfähigkeit mit direkten Nachbarn), werden noch zusätzliche Hilfsmittel benötigt, um das Netz näher zu beschreiben.

Diese Informationen werden durch ein *Dominanz-Netz* gewonnen, welches eine hierarchische Aufteilung des Netzes darstellt. Im Wesentlichen besteht das Dominanz-Netz aus mehreren Schichten von ρ -Netzen mit exponentiell wachsendem Radius ρ . Jede dieser Schichten deckt hierbei das gesamte Netz ab. Ein Dominanz-Netz soll folgende Eigenschaften vorweisen können:

1. Die Anzahl der Schichten beträgt höchstens $1 + \lceil \log \Theta \rceil$, Θ bezeichnet hier den Durchmesser des Netzes.
2. Jeder Knoten wird von höchstens $O(\log \Theta)$ Knoten dominiert.
3. Das Dominanz-Netz gewährleistet eine Entfernungskennzeichnung mit einer Markierungsgröße (Labelgröße) $O(\log \Theta)$ und einem Ausdehnungsfaktor¹ von höchstens $\sqrt{6}$.
4. Die Einführung von Routing-Tabellen der Größe $O(1/\varepsilon)^\alpha \log \Delta \log \Theta$ Bits für alle Knoten, ermöglicht ein Routing Schema mit Ausdehnungsfaktor $1 + \varepsilon$, Δ bezeichnet hierbei den Maximalgrad von G .

Im weiteren Verlauf wird o.B.d.A. davon ausgegangen, dass der kleinste Abstanz zweier Knoten 1 beträgt, der Durchmesser des Netzes sei durch Θ gegeben und $\vartheta = 1 + \lceil \log \Theta \rceil$. Desweiteren bezeichnet $G = (V, E)$ den Graph eines Netzes mit einer Constant Doubling Dimension α .

4.1 Dominanz

Um ein Dominanz-Netz aufzubauen, wird eine Hierarchie von ρ -Netzen auf G erstellt, mit $\rho = 2^i$ und $i \in \{0, 1, 2, 3, \dots, \vartheta - 1\}$. Das 2^i -Netz wird als Ebene- i der Heirarchie bezeichnet, und die Netz-Zentren dieser Ebene mit der Menge Γ_i . Auf Ebene- $(\vartheta - 1)$ gibt es nur noch ein Netz-Zentrum, dieser Knoten wird als Wurzel der Hierarchie bezeichnet. Der Algorithmus aus [1] um diese ρ -Netze verteilt zu errechnen wird hier in Abschnitt 8 beschrieben.

¹Der Faktor um den die approximierete Entfernung von der tatsächlichen abweicht.

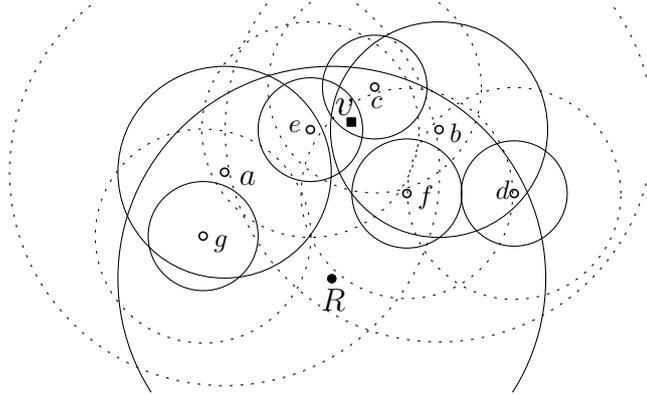


Abbildung 4.1: Ein Dominanz-Netz mit 3 Ebenen. Mit R ist hier die Wurzel bezeichnet. R ist das einzige Netz-Zentrum der Ebene-2, auf Ebene-1 sind $\Gamma_1 = \{a, b, R\}$ die Netz-Zentren, auf Ebene-0 $\Gamma_0 = \{c, d, e, f, g, a, b, R\}$. Die durchgezogenen Kreise stellen die die normalen Abdeckungsgebiete der Netz-Zentren dar, die Gepunkteten die Abdeckungsgebiete im erweiterten Dominanz-Netz.

Dominanz ist für diese ρ -Netz Hierarchie wie folgt definiert: Ein Netz-Zentrum $\gamma_i \in \Gamma_i$ wird von einem Netz-Zentrum $\gamma \in \Gamma_{i+1}$ dominiert falls $\gamma_i \in \mathcal{B}_\gamma(2^{i+1})$. So wird jedes Netz-Zentrum, ausgenommen die Wurzel, von mindestens einem Netz-Zentrum der nächst höheren Ebene dominiert.

4.2 Namensgebung

Mit Hilfe des Dominanz-Netzes werden die Netz-Zentren wie folgt benannt: Jedes Netz-Zentrum v mit Ausnahme der Wurzel, wählt genau ein Netz-Zentrum, von dem es dominiert wird, als Elternteil $\mathcal{P}(v)$ aus. Hieraus resultiert ein Dominanz-Baum der Höhe $\vartheta - 1$. Jedes Elternteil nummeriert seine Kinder durch und teilt ihnen ihre zugewiesene Nummer mit. Das Schema zur Namensgebung wird nun rekursiv definiert: Die Wurzel besitzt einen leeren Namen, jedes weitere Netz-Zentrum erhält seinen Namen, indem es seine Nummerierung an den Namen seines Elternteils anhängt.

Aus dem Dominanz-Netz aus Abbildung 4.1 und dem dazugehörigen Dominanz-Baum aus Abbildung 4.2 ergibt sich zum Beispiel für das Netz-Zentrum c der Name $R : 1 : 2$ oder für das Netz-Zentrum c $R : 2 : 3$.

Da jedes Netz-Zentrum höchstens Elternteil von $2^{2\alpha}$ Netz-Zentren ist (Satz 3.1), kann die Nummerierung mit höchstens 2α Bits dargestellt werden. Hierdurch steigt der Speicheraufwand eines Namens um höchstens 2α Bits pro Ebene, welcher für ein Netz-Zentrum der Ebene-0 mit einer Größe von $2\alpha \lceil \log \Theta \rceil = O(\log \Theta)$ Bits sein Maximum erreicht.

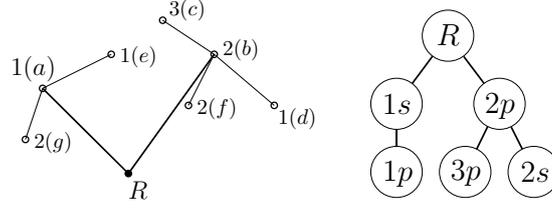


Abbildung 4.2: Die linke Abbildung zeigt einen Dominanz-Baum zu dem Dominanz-Netz aus Abbildung 4.1. Die rechte Abbildung zeigt den Abdeckungsbaum des Knotens v aus Abbildung 4.1. In der Tiefe i enthält der die Nummerierung der Netz-Zentren aus Ebene- $(\vartheta - i - 1)$, die v abdecken. Die Endungen p und s zeigen hierbei ob es sich dabei um ein primäres oder sekundäres Netz-Zentrum handelt.

4.3 Dominanz-Markierung

Mit Hilfe des Dominanz-Netzes und der Namensgebung errechnet sich nun jeder Knoten v des Graphen G eine Markierung $\mathcal{L}(v)$ der Größe $O(\log \Theta)$, diese wird effizientes Routing, Multicast, Anycast und Entfernungsabschätzungen ermöglichen. Um dies zu zeigen wird nun zunächst folgender Satz veranschaulicht:

Satz 4.1 (Kompakte Markierungen) *Den Knoten von G können einzigartige Markierungen der Größe $O(\log \Theta)$ zugewiesen werden. Dies ermöglicht Entfernungsabschätzungen mit einem Ausdehnungsfaktor $\sqrt{6}$ zwischen zwei beliebigen Knoten.*

Für die Namensgebung wird ein *erweitertes Dominanz Netz* verwendet, in diesem deckt ein Netz-Zentrum $\gamma \in \Gamma_i$ alle Knoten in $\mathcal{B}_\gamma(2^{i+1})$ ab. Durch diese Erweiterung ergibt sich eine Überlappung der Abdeckungen, dies verhindert Randeffekte bei denen zwei nahe beieinanderliegende Knoten nicht von gemeinsamen Netz-Zentren abgedeckt werden würden. In diesem erweiterten Dominanz Netz ermittelt nun jeder Knoten von welchen Netz-Zentren er abgedeckt wird, eine Liste derer Namen muss in der Markierung des Knotens enthalten sein.

Um die angestrebte Markierungsgröße für einen Knoten v zu erreichen, können jedoch nicht einfach alle Namen der abdeckenden Netz-Zentren gespeichert werden. Mit höchstens ϑ Ebenen mit je höchstens $2^{2\alpha}$ Netz-Zentren die v abdecken (Satz 3.2) und einer Namensgröße von $2\alpha(\vartheta - i - 1)$ Bits auf Ebene- i ergibt sich eine Markierungsgröße von $\sum_{i=0}^{\vartheta} 2^{2\alpha} 2\alpha(\vartheta - i - 1) = O(\vartheta^2) = O(\log^2 \Theta)$ Bits. Um aber eine Markierungsgröße in $O(\log \Theta)$ zu erreichen, hilft das folgende Lemma, welches zeigt, dass nur Teile der Namen der Netz-Zentren pro Ebene gespeichert werden müssen.

Lemma 4.2 (Abdeckung der Eltern) *Deckt ein Netz-Zentrum $c \in \Gamma_i$ einen Knoten v im erweiterten Dominanz-Netz ab, so deckt auch das Elternteil $\mathcal{P}(c) \in \Gamma_{i+1}$ den Knoten v ab.*

Beweis. Ein Netz-Zentrum c auf Ebene- i deckt im erweiterten Dominanz-Netz die Knoten in $\mathcal{B}_c(2^{i+1})$ ab. Daher gilt: $d_{\mathcal{M}}(v, c) \leq 2^{i+1}$. Da das Elternteil $\mathcal{P}(c) \in \Gamma_{i+1}$, gilt: $d_{\mathcal{M}}(c, \mathcal{P}(c)) \leq 2^{i+1}$. Durch die Dreiecksungleichung erhält man: $d_{\mathcal{M}}(v, \mathcal{P}(c)) \leq d_{\mathcal{M}}(v, c) + d_{\mathcal{M}}(c, \mathcal{P}(c)) \leq 2^{i+1}$. Hieraus folgt die Behauptung, da im erweiterten Dominanz-Netz $\mathcal{P}(c)$ die Kugel $\mathcal{B}_{\mathcal{P}(c)}(2^{i+1})$ abdeckt, welche v beinhaltet. \square

4.4 Abdeckungs-Baum

Die Markierung eines Knotens v ist die flache Abbildung eines Abdeckungsbaums, dieser beinhaltet alle Netz-Zentren die v abdecken. Der Aufbau eines Abdeckungsbaum ist rekursiv definiert: Der Baum enthält auf der Höhe i alle Netz-Zentren $\gamma \in \Gamma_{\vartheta-i-1}$, die v abdecken. Jedes γ im Abdeckungs-Baum ist mit seinem Elternteil $\mathcal{P}(\gamma)$ verbunden (Lemma 4.2).

Da der Name eines Netz-Zentrums durch das rekursive Aneinanderhängen der Nummerierungen seiner Vorfahren definiert ist, und der Abdeckungs-Baum dem selben Stammbaum zu Grunde liegt, reicht es aus in jedem Knoten des Abdeckungs-Baumes nur die Nummerierung des dazugehörigen Netz-Zentrums zu speichern. In einem Abdeckungsbaum kann der Name eines darin enthaltenen Netz-Zentrums c ebenfalls rekursiv gewonnen werden, da wie zuvor erwähnt alle Vorfahren von c ($\mathcal{P}(c), \mathcal{P}(\mathcal{P}(c)), \dots$) ebenfalls im Abdeckungsbaum enthalten sind. Somit werden in jedem Knoten des Abdeckungs-Baumes höchstens 2α Bits gespeichert und aus Satz 3.2 wissen wir, dass ein Knoten v pro Ebene von höchstens $2^{2\alpha}$ Netz-Zentren abgedeckt wird. In Verbindung mit der maximalen Tiefe $\vartheta - 1$ des Abdeckungs-Baumes umfasst dieser also höchstens $\alpha 2^{2\alpha+1}\vartheta$ Bits. Um den Baum in eine flache Struktur umzuwandeln werden noch 2 zusätzliche Bits pro Netz-Zentrum benötigt, hieraus resultiert eine Markierungsgröße von $(2\alpha + 2)2^{2\alpha}\vartheta = O(\log \Theta)$ Bits.

Für ihre Zwecke mussten Flury und Wattenhofer [1] die Markierungen allerdings noch um eine weitere Information erweitern, und zwar speichern sie nicht nur von welchen Netz-Zentren ein Knoten v abgedeckt wird, sie unterteilen diese Netz-Zentren noch in zwei Kategorien. Ein Netz-Zentrum $c \in \Gamma_i$ wird als Primär bezeichnet wenn es v im normalen Dominanz-Netz abdeckt, sprich $v \in \mathcal{B}_c(2^i)$, und als Sekundär wenn es v nur im erweiterten Dominanz-Netz abdeckt. Das hierdurch zusätzlich benötigte Bit pro Netz-Zentrum ändert den Speicheraufwand nur minimal und er ist immernoch in $O(\log \Theta)$.

4.5 Entfernungs-Abschätzung

Um die Entfernung zwischen zwei Knoten a und b mit den dazugehörigen Markierungen $\mathcal{L}(a)$ und $\mathcal{L}(b)$ abzuschätzen, wird zunächst die niedrigste Ebene- i ermittelt, in der a und b von dem selben Netz-Zentrum abgedeckt werden. Sei $\rho = 2^i$ der Radius der Netz-Zentren in Ebene- i und $C \subseteq \Gamma_i$ die Menge der gemeinsamen Netz-Zentren aus Ebene- i . So ergibt sich unmittelbar eine untere Schranke für die Entfernung zwischen a und b , da wenn $d_{\mathcal{M}}(a, b) \leq \rho/2$ gelten würde, gäbe es ein Netz-Zentrum $c \in \Gamma_{i-1}$ das sowohl a

als auch b abdecken würde. Deshalb gilt: $d_{\mathcal{M}}(a, b) > \rho/2$. In diesem Szenario können die folgenden drei Fälle auftreten:

1. Existiert ein Netz-Zentrum $c \in C$, für das gilt: c ist für a und b primäres Netz-Zentrum, so gilt $d_{\mathcal{M}}(a, b) \leq 2\rho$ und somit $d_{\mathcal{M}}(a, b) \in (\rho/2, 2\rho]$. Dies folgt aus der Dreiecksungleichung: $d_{\mathcal{M}}(a, c) \leq \rho$ und $d_{\mathcal{M}}(b, c) \leq \rho$ da c primär für a und b , daraus folgt $d_{\mathcal{M}}(a, b) \leq d_{\mathcal{M}}(a, c) + d_{\mathcal{M}}(b, c) \leq 2\rho$.
2. Alle Netz-Zentren $c \in C$ sind sekundär für a und b , so gilt: $d_{\mathcal{M}}(a, b) \in (\rho, 4\rho]$. Die obere Schranke ergibt sich ebenfalls aus der Dreiecksungleichung, die untere liegt darin begründet, dass falls $d_{\mathcal{M}}(a, b) \leq \rho$ jedes primäre Netz-Zentrum von a auch b abdecken würde, und umgekehrt.
3. Existiert mindestens ein Netz-Zentrum $c \in C$, welches entweder für a oder aber für b primär ist, so gilt: $d_{\mathcal{M}}(a, b) \in (\rho/2, 3\rho]$. O.B.d.A. sei c für a primär, dann gilt $d_{\mathcal{M}}(a, c) \leq \rho$ und $d_{\mathcal{M}}(b, c) \leq 2\rho$. Und somit folgt $d_{\mathcal{M}}(a, b) \leq d_{\mathcal{M}}(a, c) + d_{\mathcal{M}}(b, c) \leq 3\rho$.

Eine Abschätzung kann durch das geometrische Mittel der nun bekannten Intervallgrenzen gewonnen werden, somit ist $d_{\mathcal{M}}(a, b) = \sqrt{r_1 r_1}$. Die maximale Abweichung dieser Abschätzung von der tatsächlichen Entfernung beträgt $\sqrt{r_2/r_1}$ welche ihr Maximum im dritten Fall erreicht: $\sqrt{3\rho/\frac{\rho}{2}} = \sqrt{6}$. Somit beträgt der maximale Ausdehnungsfaktor dieser Markierungs Methode $\sqrt{6}$.

5 Routing

Der von Flury und Wattenhofer in [1] vorgestellte Punkt zu Punkt Routingalgorithmus (single destination Routingalgorithm (SDR)), liefert die Grundlagen für deren Multicast- und Anycast-Ansätze. Das Ziel beim SDR ist es eine Nachricht vom Senderknoten s zum Empfängerknoten t zu leiten. Der Ausdehnungsfaktor $S_{\mathcal{A}}$ eines Routingalgorithmus wird durch $S_{\mathcal{A}} = \max_{s,t \in V} \frac{d_{\mathcal{A}}(s,t)}{d_{\mathcal{M}}(s,t)}$, wobei $d_{\mathcal{A}}(s,t)$ die Länge des durch den Routingalgorithmus gefundenen Pfad bezeichnet. Diesen Ausdehnungsfaktor gilt es so klein wie möglich zu halten, jedoch ist er in diesem Ansatz an die Größe der Routing-Tabellen gekoppelt.

Satz 5.1 *Für ein festes ε , $0 < \varepsilon \leq 2$, beträgt der Ausdehnungsfaktor des SDR aus [1] $(1+\varepsilon)$. Der Header jeder Nachricht benötigt $2\alpha \lceil \log \Theta \rceil + \lceil \log n \rceil$ Bits, die Routing Tabellen der einzelnen Knoten $O(\frac{1}{\varepsilon})^\alpha (\log \Theta)(O(\alpha) + \log \Delta)$ Bits.*

Das Routing Schema baut auf den Markierungen der Knoten auf: Der Senderknoten s entnimmt der Markierung $\mathcal{L}(t)$ den Namen eines beliebigen Netz-Zentrums $\gamma \in \Gamma_0$, das den Zielknoten t abdeckt. Dieser Name \mathcal{N} und die ID von t bilden die Headerinformation der Nachricht, und benötigt höchstens $2\alpha \lceil \log \Theta \rceil + \lceil \log n \rceil$ Bits. Da \mathcal{N} eine Verknüpfung aus dem Nummerierungswerten von γ und deren Vorfahren ist, beschreibt \mathcal{N} , pro Ebene, genau ein Netz-Zentrum, das t abdeckt. Diese Netz-Zentren seien durch $\{c^0, c^1, c^2, \dots, 2^{\theta-2}\}$ dargestellt, wobei c^i das Netz-Zentrum auf in Ebene- i ist. Jeder Knoten, der die Nachricht erhält, sucht nun das Netz-Zentrum c^i der niedrigsten Ebene- i , für das er einen Routing-Eintrag besitzt, und leitet die Nachricht in dessen Richtung weiter. Dieses Verfahren wird wiederholt, bis die Nachricht einen direkten Nachbarknoten von t erreicht, welcher die Nachricht dann direkt an t sendet.

Um diese Nachrichtenweiterleitung zu unterstützen, muss jeder Knoten wissen, wie er einige nahegelegene Netz-Zentren erreichen kann. Deshalb stellt sich jedes Netz-Zentrum $\gamma \in \Gamma_i$ allen Knoten in aus $\mathcal{B}_\gamma(\eta 2^i)$, $\eta = \frac{8}{\varepsilon} + 6$, vor. Anstatt den optimalen Pfad zu γ zu speichern, reicht es aus den nächsten Knoten auf optimalen Pfad zu v speichern. Um den optimalen Pfad zu errechnen, kann eine verteilte Implementierung des Algorithmus von Dijkstra angewendet werden. Die Routing-Tabelle eines jeden Knotens v speichert sich also wie dieser jedes Netz-Zentrum $\gamma \in \Gamma_i$, für das $d_{\mathcal{M}}(v, \gamma) \leq \eta 2^i$, erreichen kann.

5.1 Routing mit einem Ausdehnungsfaktor von $(1 + \varepsilon)$

Um zu verstehen dass dieser Ansatz gute Routing Pfade liefert, sieht man zunächst, dass, sobald eine Nachricht ein Netz-Zentrum das t abdeckt, sie zu einem Netz-Zentrum weitergeleitet werden kann, welches t auf einer niedrigeren Ebene abdeckt.

Lemma 5.2 (Netz-Zentrum Sprünge) *Die Routing-Tabelle eines Netz-Zentrums $c^i \in \Gamma_i$, welches den Zielknoten t abdeckt, enthält die Routing Einträge zu allen Netz-Zentren $c^{i-x} \in \Gamma_{i-x}$, die t abdecken. Dies gilt für $x \leq \lceil \log_2 \frac{\eta-2}{2} \rceil$.*

Beweis. Angenommen $c^j \in \Gamma_j$, $j < i$ sei ein Netz-Zentrum, das t abdeckt. Durch den Aufbau der Routing-Tabellen besitzt jeder Knoten aus $\mathcal{B}_\gamma(\eta 2^i)$ einen Routing-Eintrag für c^j . Mit Hilfe der Dreiecksungleichung ergibt sich $d_{\mathcal{M}}(c^j, c^i) \leq 2^{j+1} + 2^{i+1}$. Dies stellt sicher, dass c^i einen Routing-Eintrag für c^j besitzt, falls $d_{\mathcal{M}}(c^j, c^i) \leq \eta 2^j$, dies gilt für $x = i - j \leq \lfloor \log_2 \frac{\eta-2}{2} \rfloor$. \square

Der folgende Beweis aus [1] für den Ausdehnungsfaktor des Routings beschreibt das Worst Case Szenario, in welchem eine Nachricht über alle Netz-Zentren $\{c^i, c^{i-x}, c^{i-2x}, \dots\}$ gesendet wird, c^i sei hierbei das Netz-Zentrum zu welchem s die Nachricht geschickt hat.

Beweis (1 + ε Routing). Sei c^i das Netz-Zentrum der niedrigsten Ebene für welches s einen Routing-Eintrag besitzt. Dem Routing Schema zufolge leitet s die Nachricht zunächst an c^i weiter, von dort kann sie dann an c^{i-x} , $x = \lfloor \log_2 \frac{\eta-2}{2} \rfloor$, geschickt werden (Lemma 5.2). Dies kann wiederholt werden bis die Nachricht ein Netz-Zentrum in der Ebene-0 erreicht. Von dort aus wird die Nachricht dann direkt an t übermittelt. Somit ist der Abstand $d_{SDR}(s, t)$ durch

$$d_{\mathcal{M}}(s, c^i) + d_{\mathcal{M}}(c^i, c^{i-x}) + d_{\mathcal{M}}(c^{i-x}, c^{i-2x}) + \dots + 1$$

begrenzt. Die Dreiecksungleichung liefert hier $d_{\mathcal{M}}(s, c^i) \leq d_{\mathcal{M}}(s, t) + 2^{i+1}$, und für $j \geq 0$: $d_{\mathcal{M}}(c^{i-jx}, c^{i-(j+1)x}) \leq 2 \cdot 2^{i-jx} + 2 \cdot 2^{i-(j+1)x}$. Folglich gilt:

$$d_{SDR}(s, t) \leq d_{\mathcal{M}}(s, t) + 4 \sum_{j=0}^{\infty} 2^{i-jx} \tag{5.1}$$

$$= d_{\mathcal{M}}(s, t) + 4 \sum_{j=0}^{\infty} \frac{2^i}{2^{\lceil \log_2 \frac{\eta-2}{2} \rceil j}} \tag{5.2}$$

$$\leq d_{\mathcal{M}}(s, t) + 2^{i+2} \sum_{j=0}^{\infty} \left(\frac{4}{\eta-2}\right)^j \tag{5.3}$$

$$\leq d_{\mathcal{M}}(s, t) + 2^{i+2} \frac{\eta-2}{\eta-6} \tag{5.4}$$

\square

Da $2^{\lceil \log_2 \frac{\eta-2}{2} \rceil j} \geq 2^{j \log_2 \frac{\eta-2}{4}} = ((\eta-2)/4)^j$ folgt 5.3 aus 5.2. Hieraus ergibt sich 5.4, da die Summe in (3) eine Geometrische Reihe ist, $\frac{4}{\eta-2} = \frac{\varepsilon}{2+\varepsilon} < 1$.

Weil s keinen Routing-Eintrag für ein Netz-Zentrum c^j , $j < i$ besitzt ist $d_{\mathcal{M}}(s, t) > (\eta-2)2^{i-1}$, da $d_{\mathcal{M}}(c^{i-1}, t) \leq 2 \cdot 2^{i-1}$ und weil alle Knoten in $\mathcal{B}_{c^{i-1}}(\eta 2^{i-1})$ einen Routing Eintrag für c^{i-1} besitzen. Deshalb hätte s einen Routing-Eintrag für c^{i-1} , wenn $d_{\mathcal{M}}(s, t) \leq (\eta-2)2^{i-1}$.

Aus diesen Schranken für $d_{\mathcal{M}}(s, t)$ und $d_{SDR}(s, t)$ ergibt sich folgender Ausdehnungsfaktor für den SDR:

$$\begin{aligned} S_{SDR} &\leq \frac{d_{SDR}(s, t)}{d_{\mathcal{M}}(s, t)} \leq 1 + \frac{2^{i+2} \frac{\eta-2}{\eta-6}}{2^{i-1}(\eta-2)} \\ &= 1 + \frac{8}{\eta-6} = 1 + \varepsilon \end{aligned}$$

Im Gegensatz zu diesem Worst Case Szenario, kann man im Average Case einen beachtlichen Performancezuwachs beobachten, wenn man nach jedem Schritt überprüft, ob ein t näher gelegenes Netz-Zentrum bekannt ist. Ist ein solch näheres Netz-Zentrum c^k , $k < j$, bekannt, während eine Nachricht in Richtung c^j weitergeleitet wird, so wird die Nachricht nun in Richtung c^k umgeleitet, so entstehen Abkürzungen in Richtung t . Hierdurch wird der Pfad und somit auch den Ausdehnungsfaktor reduziert.

5.2 Kompakte Routing-Tabellen

Der wesentliche Faktor des Speicherbedarfs des Routingschemas ist die Größe der Routing-Tabellen. Hierbei ist zwar die Menge der Einträge die in jedem Knoten gespeichert werden ausschlaggebend, jedoch fällt auch die Größe der einzelnen Einträge ins Gewicht. Durch die vorgestellten Strukturen können nun kompakte Routing-Tabellen gebildet werden. Deren Aufbau ist dem der Abdeckungsbäume sehr ähnlich, der Hauptunterschied liegt im größeren Abdeckungsradius der Netz-Zentren, welche $\mathcal{B}_{\gamma}((\frac{8}{\varepsilon} + 6)2^i)$ abdecken. Deshalb muss Lemma 4.2 erweitert werden:

Lemma 5.3 (Abdeckung der Eltern II) *Besitzt ein Knoten v einen Routing-Eintrag zu einem Netz-Zentrum $\gamma \in \Gamma_i$ so besitzt es ebenfalls einen Routing Eintrag zu $\mathcal{P}(\gamma)$, ausgenommen γ ist die Wurzel.*

Beweis. Der Knoten v hat einen Routing-Eintrag zu $\gamma \in \Gamma_i$ genau dann wenn $v \in \mathcal{B}_{\gamma}(\eta 2^i)$. Ebenfalls gilt $d_{\mathcal{M}}(\gamma, \mathcal{P}(\gamma)) \leq 2^{i+1}$ und $d_{\mathcal{M}}(\gamma, v) \leq \eta 2^i$. Mit Hilfe der Dreiecksungleichung folgt: $d_{\mathcal{M}}(v, \mathcal{P}(\gamma)) \leq (\eta + 2)2^i$. Somit genügt es zu zeigen dass $\leq (\eta + 2)2^i \leq \eta 2^{i+1}$ gilt um folgern zu können, dass $v \in \mathcal{B}_{\mathcal{P}(\gamma)}(\eta 2^{i+1})$. Diese Ungleichung ist erfüllt da $\eta = \frac{8}{\varepsilon} + 6 \geq 2$. \square

Wie schon in Abschnitt 4.4 kann gezeigt werden, dass alle Netz-Zentren, zu denen ein Knoten v Routing Einträge speichern muss, in einer Baumstruktur gespeichert werden können, in der jeder Knoten des Baumes ein Netz-Zentrum darstellt. In jedem Knoten des Baumes werden lediglich der Nummerierungswert des dazugehörigen Netz-Zentrums γ und der Routing-Eintrag für γ gespeichert. Wie schon zuvor benötigt der Nummerierungswert 2α Bits und als Routing Information reicht der Knoten auf dem Optimalen Pfad zu γ aus. Also kann die Routing Information mit $\log \Delta$ Bits dargestellt werden, wobei Δ der höchste Grad eines Knotens ist.

In jedem Knoten müssen die Routing-Informationen für höchstens $2^{2\alpha}(\frac{8}{\varepsilon} + 6)^\alpha$ Netz-Zentren pro Ebene gespeichert werden (Korollar 3.1). Da es höchstens ϑ Ebenen gibt, enthält der Baum höchstens $2^{2\alpha}(\frac{8}{\varepsilon} + 6)^\alpha \vartheta$ Knoten, in denen je $2\alpha + \log_2 \Delta$ Bits gespeichert sind. Dieser Baum wird ebenfalls durch 2 zusätzliche Bits pro Knoten in einer flachen Datenstruktur gespeichert, was zu einer Größe der Routing-Tabellen von je $(2\alpha + \log_2 \Delta + 2)2^{2\alpha}(\frac{8}{\varepsilon} + 6)^\alpha \vartheta = O(\frac{1}{\varepsilon})^{alpha}(\log \Theta)(O(\alpha) + \log \Delta)$ Bits führt.

6 Multicasting

Mit dem SDR aus 5 als Grundbaustein wird in [1] zunächst Multicasting betrachtet. Das Ziel ist es hierbei eine Nachricht von dem Senderknoten s an eine Menge U von Empfängerknoten zu senden. Im Wesentlichen errechnet der Senderknoten s einen approximierten minimalen Spannbaum auf der Menge $s \cup U$, und sendet die Nachricht entlang dieses Baumes zu den Empfängerknoten, zur Berechnung des MST wurde der Algorithmus von Kruskal verwendet. Folgendes Ergebnis wird hierdurch erzielt:

Satz 6.1 *Ist ein Netz $G = (V, E)$, zu dem ein Dominanz-Netz und die dazugehörigen Routing-Tabellen und Knoten-Markierungen errechnet wurden, gegeben, so kann jeder Knoten $s \in V$ an jede Knotenmenge $U \subset V$ eine Nachricht mit Multicasting senden, wobei der Ausdehnungsfaktor konstant ist. Die mit diesem Multicasting verbundenen Kosten belaufen sich auf das höchstens $12(1+\varepsilon)$ -fache der Kosten eines optimalen Multicast Algorithmus, der die gesamte Netztopologie kennt.*

Beweis. Es gilt zu zeigen, dass der Pfad entlang des approximierten MST höchstens $12(1+\varepsilon)$ mal so lang ist, als der optimale Pfad, welcher durch einen minimalen Steiner-Baum (MStT) auf $s \cup U$ gegeben ist. Die übrigen Knoten $V \setminus (s \cup U)$ dienen hierbei als Steiner-Punkte. Der Ausdehnungsfaktor setzt sich aus 3 Komponenten zusammen: Ersteinmal ist ein MST auf $s \cup U$ eine 2-Approximation gegenüber dem korrespondierenden MStT [3]. Desweiteren fließt der Ausdehnungsfaktor $(1+\varepsilon)$ des SDR mit ein, über den die Nachrichten gesendet werden. Zuletzt kommt noch hinzu, dass der Aufbau des MST auf die Entfernungs-Markierungen mit Ausdehnungsfaktor $\sqrt{6}$ basiert. Der Algorithmus von Kruskal wählt somit im schlechtesten Fall eine Kante aus, die um den Faktor $\sqrt{6}$ mal größer ist als die Abschätzung für diese Kante, und die tatsächlich kürzeste Kante ist um den Faktor $\sqrt{6}$ mal kleiner als deren Abschätzung. Deshalb ist es möglich, dass der Algorithmus von Kruskal in diesem Falle in jedem Schritt eine bis zu 6-mal längere Kante, gegenüber der optimalen, auswählt, wodurch der approximierte MST bis zu 6 mal so lang wie der eigentliche MST sein kann. \square

7 Anycast

Für den Spezialfall Anycast, bei dem eine Nachricht zu genau einem Knoten aus einer Empfängerknotenmenge U geroutet werden muss, liefern Flury und Wattenhofer [1] ebenfalls einen Algorithmus mit konstanten Ausdehnungsfaktor:

Satz 7.1 *Der SDR aus Abschnitt 5 zusammen mit den Knoten-Markierungen aus Abschnitt 4.3 ermöglicht eine $6(1 + \varepsilon)$ - Approximation für Anycasting.*

Beweis. Mit Hilfe der Entfernungsabschätzung durch die Markierungen, wird der Knoten $u \in U$ mit der scheinbar kürzesten Entfernung zu s gesucht und anschließend die Nachricht an u geschickt. Durch den Ausdehnungsfaktor der Entfernungsabschätzung wählt dieser Ansatz im schlimmsten Falle einen 6-fach weiter entfernten Knoten, als den optimalen Empfängerknoten, aus. Hinzu kommt noch der Ausdehnungsfaktor $(1 + \varepsilon)$ des SDR. \square

8 Verteilter Aufbau des Dominanz-Netzes

Abschließend wird in [1] ein verteilter Algorithmus vorgestellt um die ρ -Netze zu berechnen und somit das Dominanz-Netz zu bilden. Wie in Abschnitt 4.1 beschrieben besteht ein Dominanz-Netz aus ϑ ρ -Netzen mit exponentiell zunehmenden Abdeckungsradius $\rho \in \{1, 2, 4, \dots, 2^{\vartheta-1}\}$.

Eine zentrale Lösung um ein ρ -Netz zu berechnen ist einfach zu implementieren: Mit einem Greedy-Algorithmus einen beliebigen Knoten $v \in V$ auswählen, v zu Γ hinzufügen, und alle Knoten aus $\mathcal{B}_v(\rho)$ aus V entfernen. Dies wiederholt man bis V die Leere Menge ist. Dieser Aufbau gewährleistet, dass der Abstand zweier Netz-Zentren stets größer als ρ ist und ebenso, dass jeder Knoten $v \in V$ von mindestens einem Netz-Zentrum abgedeckt wird.

Der Ansatz der verteilten Lösung, nutzt aus, dass ein ρ -Netz ein *Maximal Independent Set* (MIS) auf dem ρ -metrischen Abschluss¹ $G^\rho = (V, E^\rho)$ ist. So kommt man zu einem einfachen verteilten Algorithmus um ein ρ -Netz zu berechnen: Erstelle einen ρ -metrischen Abschluss bei dem jeder Knoten $v \in V$ alle Knoten aus $\mathcal{B}_v(\rho)$ als seine direkten Nachbarn hat. Berechne nun eine MIS mit Hilfe eines verteilten MIS Algorithmus und wähle die Knoten des errechneten MIS als Netz-Zentren des ρ -Netzes aus. Es gibt eine breite Auswahl an verteilten MIS Algorithmen, wie zum Beispiel der randomisierte Algorithmus von Luby [4] mit einer Laufzeit in $O(\log n)$.

Dieser verteilte Algorithmus ist ein guter Ansatz für kleine Werte für ρ , wenn ρ jedoch Werte um Θ annimmt, entfallen die Nachbarschaften der Knoten nahezu alle Knoten des Netzes. Hierdurch werden die Speicherkapazitäten der einzelnen Knoten schnell überschritten, vor allem in großen Netzen. Um diesem Problem vorzubeugen stellen Flury und Wattenhofer in [1] einen verteilten Algorithmus vor, der nicht nur relativ schnell, sondern auch speichersparend ist.

8.1 Sequentielle ρ -Netze

Algorithmus 1 beschreibt die einzelnen Schritte, die jeder Knoten v des Netzes ausführt um das Dominanz-Netz zu errechnen. Nach Ausführung des Algorithmus weiß jeder Knoten in welchen ρ -Netzen er ein Netz-Zentrum ist, ebenso besitzt er bereits die vollständige Routing-Tabelle für den SDR aus Abschnitt 5. Kurz gesagt handelt es sich um einen rekursiven Algorithmus, der folgende zwei Schritte ausführt um ein ρ -Netz der Ebene- i zu berechnen:

¹Der n -metrische Abschluss eines Graphen $G = (V, E)$ ist der Graph $G^n = (V, E^n)$ mit $E^n = \{(u, v) | u, v \in V \wedge u \neq v \wedge d_{\mathcal{M}}(u, v) \leq n\}$

Algorithmus 1 : Dominanz-Netz

```
1 status = aktiv
2 Main()
3    $N = \text{Knoten in } \mathcal{B}_v(1)$ 
4   wenn  $v \notin \text{MIS}$  dann
5     state = passiv
6   sonst
7     Benachrichtige  $\mathcal{B}_v(1)$ 
8     Füge in  $\mathcal{B}_v(1)$  Routing Einträge für  $v$  hinzu
9      $\text{VereineMIS}(1)$ 
10  Ende
11
12 VereineMIS(i)
13    $\rho = 2^i$ 
14    $N = \text{angeregte Knoten mit max. Abstand } \rho$ 
15   Berechne MIS mit der Nachbarschaft  $N$ 
16   wenn  $v \notin \text{MIS}$  dann
17     status = passiv
18     solange  $\exists w \in \mathcal{B}_v(\frac{\rho}{2})$ , so dass  $w$  nicht abgedeckt ist tue
19        $P = \text{beliebiger nicht abgedeckter Knoten aus } \mathcal{B}_v(\frac{\rho}{2})$ 
20       Sende  $\text{EXC}(i)$  an  $P$  und warte auf Antwort
21     Ende
22     Sende  $\text{ACK}(i)$  an  $N$ 
23   sonst
24     Benachrichtige  $\mathcal{B}_v(\rho)$ 
25     Füge in  $\mathcal{B}_v(\eta\rho)$  Routing Einträge für  $v$  hinzu
26     Sammle  $\text{ACK}(i)$  von allen Nachbarn  $N$  ein
27      $\text{VereineMIS}(i + 1)$ 
28   Ende
29
30 EmpfangeNachricht(EXC(i) from u)
31   status = angeregt,  $\rho = 2^i$ 
32   Füge in  $\mathcal{B}_v(\eta\rho)$  temporäre Routing Einträge für  $v$  hinzu
33    $N = \text{angeregte Knoten mit max. Abstand } \rho$ 
34   Berechne MIS mit der Nachbarschaft  $N$ 
35   wenn  $v \in \text{MIS}$  dann
36     status = aktiv
37     Benachrichtige  $\mathcal{B}_v(\rho)$ 
38     Füge in  $\mathcal{B}_v(\eta\rho)$  Routing Einträge für  $v$  hinzu
39      $\text{VereineMIS}(i + 1)$ 
40   sonst
41     status = passiv
42     Entferne temporäre Routing Einträge für  $v$  aus  $\mathcal{B}_v(\eta\rho)$ 
43   Ende
44   Sende  $\text{ACK}(\text{state})$  an  $u$ 
45
```

1. Ein ρ -Netz wird zu gegebenem $\frac{\rho}{2}$ -Netz approximiert, zum Beispiel kann ein MIS auf einem ρ -metrischen Abschluß von Γ_{i-1} berechnet werden, welches dann Γ_i bildet. Hierzu sei noch bemerkt, dass ein so gewonnenes MIS nicht zwangsläufig alle Knoten des Netzes abdeckt, sondern lediglich die Abdeckung von Γ_{i-1} gewährleistet.
2. Zusätzliche Netz-Zentren werden zu Γ_i hinzugefügt bis alle Knoten $v \in V$ von je mindestens einem $\gamma \in \Gamma_i$ abgedeckt werden.

Die erstrebte Speicherersparnis wird erzielt, indem die ρ -Netze sequenziell aufeinander aufgebaut werden. Als erstes wird mit Hilfe aller Knoten ein MIS auf dem 1-metrischen Abschluss des Netzes berechnet, woraus sich das 1-Netz ergibt. Nun wird der ρ -metrische Abschluss G_i^ρ , für jede Ebene- i , nur aus den Netz-Zentren Γ_{i-} gebildet und somit besitzt ein Knoten $\gamma \in G_i^\rho$ höchstens $2^{2\alpha}$ Nachbarn, unabhängig von der Netzgröße (Satz 3.1). Die Kehrseite dieses Ansatzes liegt in der längeren Laufzeit und darin, dass die errechneten MIS nicht zwingend alle Knoten des Netzes abdecken. Deshalb wird eine zweite Berechnungsphase benötigt in der dem MIS zusätzliche Netz-Zentren hinzugefügt werden, bis eine volle Abdeckung erreicht ist.

Bei der rekursiven Berechnung des MIS der nächst höheren Ebene (Zeilen 12-14) ist zu beachten, dass lediglich die Netz-Zentren aus Ebene- $(i-1)$ an der Berechnung des MIS für Ebene- i beteiligt sind. Ist ein Knoten $v \in \Gamma_{i-1}$ nicht im MIS der Ebene- i enthalten, so verfällt er in einen passiven Zustand und ist dafür verantwortlich, dass alle Knoten in $\mathcal{B}_v(2^{i-1})$ durch Γ_i abgedeckt werden (Zeilen 17-21). Dieser Abdeckungs-Algorithmus benötigt einige Durchgänge um die entsprechenden Knoten die den MIS ergänzen zu bestimmen, der Knoten v ist bei diesem Prozess so lange beteiligt, bis alle Knoten in $\mathcal{B}_v(2^{i-1})$ abgedeckt werden. In jedem Durchgang wählt jedes Netz-Zentrum v , für das $\mathcal{B}_v(2^{i-1})$ unabgedeckte Knoten enthält, einen dieser unabgedeckten Knoten aus und versetzt ihn in den Zustand *angeregt*, indem er an ihn eine *EXC* Nachricht schickt. Die Menge der angeregten Knoten berechnet nun auf ihrem ρ -metrischen Abschluß ein MIS (Zeilen 32-33), und die in diesem MIS enthaltenen Knoten werden nun dem (2^i) -Netz hinzugefügt (Zeilen 35-38). Die in Zeile 31 eingeführten temporären Routing Einträge dienen zur Kommunikation während der Berechnung des MIS auf dem entsprechenden metrischen Abschluss der angeregten Knoten und werden falls der Knoten nicht im MIS enthalten sein sollte in Zeile 41 wieder entfernt.

Aus Gründen der Überschaubarkeit werden folgende Probleme im Pseudocode zu Algorithmus 1 nicht näher erläutert:

- Empfängt ein Knoten v mehrere *EXC* Anfragen im selben Durchgang, so werden die Zeilen 30-42 nur einmal ausgeführt, jedoch werden danach in Zeile 43 an alle Knoten die eine Anfrage geschickt haben auch eine Bestätigung geschickt.
- Die Netz-Zentren müssen ihre Nummerierungswerte (siehe Abschnitt 4.2) erhalten haben um ihre Markierungen erstellen zu können. Da die Markierungen unvollständig sind, solange nicht alle ρ -Netz berechnet wurden, wurden die Routing Einträge (erstellt in den Zeilen 7, 24, 31, 37) aus Teilen der Markierungen erstellt.

- Es wird eine grobe Synchronisierung zwischen den aufeinander folgenden MIS Berechnungen benötigt, so daß eine MIS Berechnung erst dann startet wenn die vorangegangene auch sicher beendet wurde. Dies kann erreicht werden indem während der Berechnung des metrischen Abschlusses für das folgende MIS (Zeilen 13 und 33), jeder Knoten die anderen Knoten bitten kann zu warten. Um dies Synchronisierung zu unterstützen wartet jeder Knoten der dem ρ -Netz zugehört, bis all seine Nachbarn (des metrischen Abschlusses) die Abdeckungs-Überprüfung abgeschlossen haben (Zeile 25).
- Wenn in Zeile 13 gilt: $N = \emptyset$, dann gehört v selbstverständlich zum MIS und prüft nach ob er alle Knoten des Netzes abdeckt. Ist dem so, dann wird v zur Wurzel des Netzes und beendet die Rekursion.
- Bevor das erste MIS berechnet werden kann müssen die Knoten die niedrigsten Kosten einer Kante des Netzes ermitteln, um so den Normierungsfaktor berechnen zu können, der die Kosten dieser Kante auf 1 skaliert.

9 Schluss

In dieser Ausarbeitung wurde die Arbeit von Flury und Wattenhofer aus [1] näher erleutert, ein Ansatz der Lösungen für Routing, Anycast und Multicast, in drahtlosen Sensornetzen, liefert. Dies liefert den Grundstein, um, darauf aufbauend, leistungsorientierte Programme zu entwickeln. Erweiterungen, wie zum Beispiel Sicherheit, Unterstützung mobiler Sensoren, oder Fehlertoleranz werden zu diesem Zeitpunkt offen gelassen und in zukünftigen Veröffentlichungen behandelt werden.

Literaturverzeichnis

- [1] R. Flury und R. Wattenhofer. Routing, Anycast, and Multicast for Mesh and Sensor Networks. Proc. INFOCOM. 2007. [1](#), [2](#), [3](#), [4.1](#), [4.4](#), [5](#), [5.1](#), [5.1](#), [6](#), [7](#), [8](#), [9](#)
- [2] D. Peleg and, E. Upfal. A Trade-Off between Space and Efficiency for Routing Tables. J.ACM, 36(3):510–530, 1989. [2](#)
- [3] L. Kou ,G. Markowsky, and L. Berman. A Fast Algorithm for Steiner Trees. Acta Informatica, 15:141–145, 1981. [7](#)
- [4] M. Luby. A Simple Parallel Algorithm for the Maximal Independent Set Problem. SIAM Journal on Computing, 15:1036–1053, 1986. [8](#)
- [5] I. Abraham, C. Gavoille, A. Goldberg und D. Malkhi. Routing in Networks with Low Doubling Dimension. In Proc. ICDCS, 2006. [2](#)