

# Sparse Power Efficient Topology for Wireless Networks

Andreas Seelinger

Rheinisch-Westfälische Technische Hochschule Aachen (RWTH)  
Lehrstuhl für Informatik I  
Ahornstr. 55, 52074 Aachen, Germany

## Inhaltsverzeichnis

<b>1</b>	<b>Motivation</b>	<b>2</b>
<b>2</b>	<b>Einführung</b>	<b>2</b>
<b>3</b>	<b>Techniken</b>	<b>6</b>
3.1	Yao und Gabriel Graph . . . . .	6
3.2	Yao und Sink Graph . . . . .	7
3.3	Yao plus entgegengesetzter Yao Graph . . . . .	7
3.4	Ergebnisse . . . . .	8
<b>4</b>	<b>Ausblick</b>	<b>8</b>

## 1 Motivation

In der heutigen Zeit, erhalten kabellose Technologien und Netzwerke eine immer größere Bedeutung, weil sie sich gegen die traditionellen, kabelgebundenen Netzwerke immer öfter durchsetzen. Schon jetzt werden mehr kabellose als kabelgebundene Netzwerke installiert. Deswegen sollte man der Konstruktion von energieeffizienten und gesundheitsneutralen kabellosen Netzwerken mehr Aufmerksamkeit widmen. Ein kabelloses Netzwerk, bei dem Energie- und Gesundheitsaspekte eine wichtige Rolle spielen, ist das mobile Kommunikations-Netzwerk. Denn sowohl hier, als auch im Allgemeinen führt die Erhöhung der Sendeleistung auf der einen Seite automatisch zu einer erhöhten Reichweite dieses Gerätes. Auf der anderen Seite jedoch führt es zu einem nicht linearen Energieverbrauchsanstieg und möglichen Gesundheitsfolgen wegen langfristiger Einwirkung stärkerer Radiostrahlung. Außerdem ist in einer Zeit des Klimawandels und der hohen Strompreise die Energieeffizienz und Kontrolle ein wichtiges Thema.

Das Ziel ist es also ein vorhandenes kabelloses Netzwerk so zu modifizieren, dass es zum Schluss eine bessere Energieeffizienz aufweist. In meiner Ausarbeitung stelle ich einige Techniken vor, die eine solche Modifikation gewährleisten, indem sie die Topologie des Netzwerkes verändern. Dabei ist es wichtig, dass die Techniken zur Erzeugung ihrer Strukturen lokal und effizient auf den einzelnen, kabellosen Knoten ausgeführt werden können. Denn eine wichtige Eigenschaft eines kabellosen Netzwerkes ist es, dass kein Knoten die Information über das gesamte Netzwerk hat. Zur Vereinfachung wird ein kabelloses Netzwerk auf eine zwei-dimensionale Ebene abgebildet. Ein Knoten in diesem Graphen stellt ein kabelloses Gerät im dazugehörigen Netzwerk dar und eine Kante zwischen zwei Knoten existiert genau dann, wenn die beiden dazugehörigen kabellosen Geräte auf direktem Wege miteinander kommunizieren können. Die Techniken erzeugen dabei stets aus einem gegebenen Graphen einen dünnen Teilgraphen mit linearer Anzahl von Kanten und dennoch existierenden energieeffizienten Routen für zwei beliebige Knoten.

## 2 Einführung

Sei  $V$  eine nichtleere Menge von  $n$  kabellosen Knoten, die sich auf einer zwei-dimensionalen Ebene befinden und sei  $G$  der resultierende Graph daraus. Dann ist  $\|uv\| \in G$  der euklidische Abstand zwischen den beiden Knoten  $u$  und  $v$ , und  $\|uv\|^\beta$  ist die Energie, die aufgebracht werden muss, um eine Nachricht von  $u$  nach  $v$  zu transportieren und  $\beta$  ist dabei eine reelle Konstante zwischen 2 und 4. Die Konstante  $\beta$  spiegelt dabei die Schwierigkeit wieder, mit der sich die Radiowellen durch ein bestimmtes Medium bewegen, z.B. ist bei klaren Sichtverhältnissen  $\beta$  kleiner als bei Nebel. Man kann jetzt schon erkennen, dass eine Verdopplung der Sendereichweite im besten Falle zu einer Vervierfachung der benötigten Sendeenergie führt. Im folgenden stelle ich mehrere verschiedene Teilgraphen vor:

**Unit Disk Graph  $UDG(G)$**  : Hier existiert genau dann eine Kante zwischen zwei Knoten  $u$  und  $v$ , wenn der euklidische Abstand zwischen den beiden Knoten höchstens eins

## 2 Einführung

ist, d.h.  $\|uv\| \leq 1$ . Siehe Abbildung 1.

**Gerichteter relativer Nachbarschaftsgraph  $RNG(G)$**  : Hier existiert genau dann eine gerichtete Kante zwischen zwei Knoten  $u$  und  $v$ , wenn es keinen Knoten  $w \in V$  gibt, so dass  $u \rightarrow w$  und  $w \rightarrow v$  erreichen kann, siehe Abbildung 2. Mathematisch ist dies so definiert, dass  $\vec{uv} \in RNG(G)$ , wenn  $\|uw\| < \|uv\|$  und  $\|wv\| < \|uv\| \forall w \in G, w \neq u, v$ .

**Gerichteter Gabriel Graph  $GG(G)$**  : Beim Gabriel Graphen existiert genau dann eine gerichtete Kante zwischen  $u$  und  $v$ , wenn die Fläche des Kreises mit Durchmesser  $\|uv\|$  und dem Ursprung in der Mitte zwischen  $u$  und  $v$  keinen Knoten  $w \in G$  enthält, so dass  $u \rightarrow w$  und  $w \rightarrow v$  erreichen kann, siehe Abbildung 3. Mathematisch ist dies so definiert, dass  $\vec{uv} \in GG(G)$ , wenn  $\|uv\|^2 < \|uw\|^2 + \|wv\|^2 \forall w \in G, w \neq u, v$ .

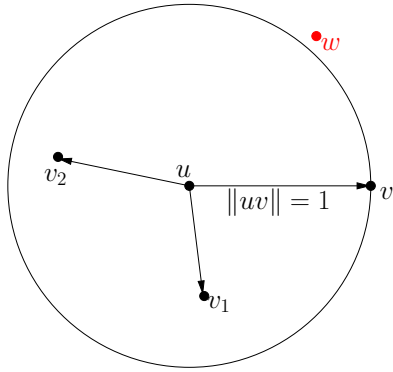
**Gerichteter Yao Graph  $YG_k(G)$**  : Der Yao Graphen von einem gegebenen Graphen  $G$  und einer festen ganzzahligen Konstante  $k \geq 6$  wird bezeichnet als  $\vec{YG}_k(G)$ . Dabei wird für jeden Knoten  $u$ , der Kreis, der seinen Ursprung in  $u$  hat und dessen Radius die Reichweite von  $u$  ist, in  $k$  gleichgroße Kegel zerlegt und in jeden Kegeln wird die kürzeste gerichtete Kante  $\vec{uv} \in G$  in  $\vec{YG}_k(G)$  aufgenommen, siehe Abbildung 4.

**Entgegengesetzter Yao Graph  $\overleftarrow{YG}_k(G)$**  : Die Auswahl der Knoten ist identisch mit der im Yao Graph, aber hier wird nicht die Kante  $\vec{uv}$  zum  $\vec{YG}_k(G)$  hinzugefügt, sondern die Kante  $\overleftarrow{uv}$ . Bei einem ungerichteten Graphen ist  $\vec{YG}_k(G) = \overleftarrow{YG}_k(G) = YG_k(G)$ .

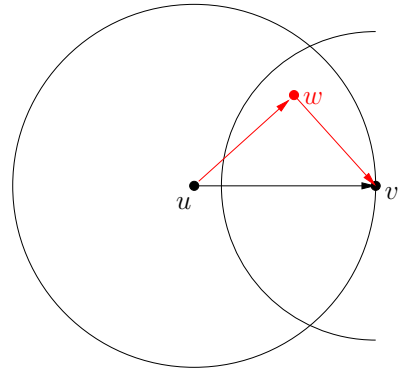
Die Eigenschaften eines kabellosen Netzwerkes werden am besten durch den  $UDG(G)$  wiedergegeben, denn ein kabelloser Knoten hat im Allgemeinen eine omnidirektionale Antenne und deswegen eine kreisförmige Signalausbreitung. Im weiteren Verlauf wird für  $G$  stets von einem Unit Disk Graphen ausgegangen. Für die vorgestellten Teilgraphen gelten die folgenden Aussagen. Für weiter Information verweise ich auf [4], [1], [2] und [9].

- Wenn  $UDG(G)$  zusammenhängend ist, dann enthält er auch den euklidischen minimalen Spannbaum ( $EMST(G)$ ).
- Wenn  $G$  ein zusammenhängender  $UDG(G)$  ist, dann sind  $RNG(G), GG(G)$  und  $YG_k(G)$  ebenfalls zusammenhängend und enthalten den  $EMST(G)$  als Teilgraphen.
- Wenn  $G$  ein  $UDG(G)$  ist, dann sind  $RNG(G), GG(G)$  und  $YG_k(G)$  planar. Unter anderem heißt das, dass viele Routingprotokolle eine sehr gute Laufzeit auf diesen Graphen haben.

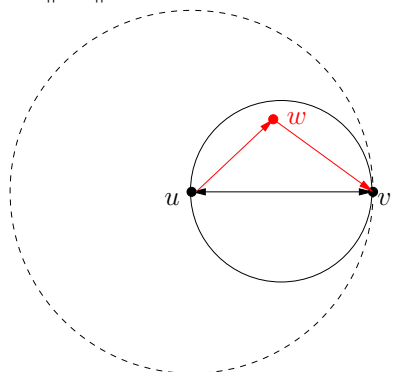
Des Weiteren erzeugen die Graphen  $RNG(G)$ ,  $GG(G)$  und  $YG(G)$  einen dünnen Teilgraphen. Dies bedeutet, dass die Kantenanzahl linear zur Knotenanzahl steigt und somit in  $O(n)$  liegt. Dabei gibt  $n$  die Anzahl der Knoten im Graphen an, siehe Abbildung 1. Eine weitere wichtige Eigenschaft erfüllen diese Graphen aber nicht, denn in einem Graphen sollte der Durchschnittsgrad im Graphen durch eine feste Konstante beschränkt sein. Der maximale Eingangsgrad beim  $RNG(G)$ ,  $GG(G)$  und  $YG(G)$  ist im schlechtesten Fall aber



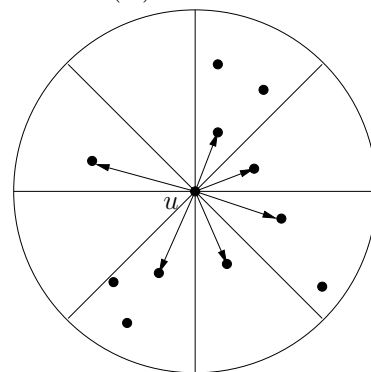
**Abbildung 1:** UDG. Kante  $uw$  liegt nicht im  $UDG(G)$ , da  $\|uw\| > 1$



**Abbildung 2:** RNG. Sollte ein solches  $w$  existieren, ist  $uv \notin RNG(G)$



**Abbildung 3:** GG. Sollte ein solches  $w$  existieren, ist  $uv \notin GG(G)$



**Abbildung 4:**  $YG_8$ : Der nächste Knoten in jedem Kegel ist  $\in YG_8$

$n - 1$ , wie das Abbildung 5 zeigt. Dies führt dazu, dass jegliche Kommunikation zwischen zwei nicht benachbarten Knoten über den Knoten  $u$  abgewickelt wird. Knoten  $u$  muss also den größten Teil der Kommunikation übernehmen und verbraucht daher auch sehr viel Energie. Im Allgemeinen kann man sagen, dass ein großer maximaler Eingangsgrad in einem Graphen zu einem großen Overhead an einigen Knoten führt. Daher sollte man Techniken entwickeln, die sowohl den Durchschnittsgrad als auch den maximalen Grad durch eine Konstante beschränken.

### Power stretch Faktor

Der Power stretch Faktor repräsentiert das maximale Verhältnis des Energieverbrauches bei der Kommunikation zweier beliebiger Knoten, im Teilgraphen  $H$  und im gesamten Graphen  $G$  [2]. Dabei ist eine Kommunikation von zwei Knoten  $u$  und  $v$  als unicast  $\pi(u, v)$  definiert mit  $u, v \in G$  und  $\pi(u, v) = v_0v_1 \cdots v_{h-1}v_h$ , wobei  $u = v_0$  und  $v = v_h$  ist. Der Energieverbrauch, der dabei entsteht, wird mit  $p(\pi)$  bezeichnet und ist die Summe über



### 3 Techniken

	Length $\lambda(G)$	Power $\rho_H(G)$	In-Grad	Out-Grad	$ G ^{-1}$
$RNG(G)$	$n - 1$	$n - 1$	$n - 1$	n.a.	$\leq 3n - 10$
$GG(G)$	$\frac{(4\pi\sqrt{2n-4})}{3}$	1	$n - 1$	n.a.	$\leq 3n - 8$
$YG_k(G)$	$\frac{1}{1-2\sin\frac{\pi}{k}}$	$\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}$	$n - 1$	$k$	$\leq kn$
$GYG_k(G)$	$\frac{1}{1-2\sin\frac{\pi}{k}}$	$\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}$	$n - 1$	$k$	$\leq kn$
$YGG_k(G)$	$\frac{1}{1-2\sin\frac{\pi}{k}}$	$\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}$	$n - 1$	$k$	$\leq 3n$
$YG_k^*(G)$	$\left(\frac{1}{1-2\sin\frac{\pi}{k}}\right)^2$	$\left(\frac{1}{1-2\sin\frac{\pi}{k}}\right)^\beta$	$(k + 1)^2 - 1$	$k$	$\leq kn$
$YY_k(G)$	$\frac{1}{1-2\sin\frac{\pi}{k}}$	$\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}$	$k$	$k$	$\leq kn$

**Tabelle 1:** Zusammenfassung wichtiger Erkenntnisse. <sup>1</sup> Kantenanzahl des Graphen

## 3 Techniken

Es werden jetzt Techniken vorgestellt, die ein dünnes Netzwerk erzeugen, mit sowohl einem beschränkten Knotengrad als auch einem beschränkten Power stretch Faktor. Ein beschränkter Knotengrad ist deswegen wichtig, weil ein beschränkter Ausgangsgrad für viele Routingalgorithmen von Vorteil ist, wie z.B. für den FACE-Algorithmus[4]. Ein beschränkter Eingangsgrad garantiert, dass bei keinem Knoten ein zu großer Overhead entsteht. Als Faustregel kann man sagen, dass ein zu großer durchschnittlicher Knotengrad zu erhöhten Interferenzen und Kollisionen führt und damit das Netzwerk langsamer und energieverschwendender macht. Ein zu geringer durchschnittlicher Knotengrad führt dazu, dass das Netzwerk eine zu geringe Ausfallstoleranz hat, d.h. bei Ausfall eines Knotens steigt der Length stretch Faktor rapide und damit auch der Power stretch Faktor.

### 3.1 Yao und Gabriel Graph

Der Gabriel Graph hat den Vorteil, dass er die Kantenanzahl auf weniger als  $3n$  verringert und dabei immer noch einen Power stretch Faktor von 1 hat, d.h. der Energiebedarf verschlechtert sich nicht trotz geringerer Kantenanzahl. Der Yao Graph hat als Vorteil, dass er einen in  $k$  beschränkten Ausgangsgrad garantiert. Nun gibt es zwei Methoden diese beiden Techniken zu vereinen. Auf der einen Seite kann man zuerst den Yao Graphen entwickeln und auf diesem nochmal die Technik des Gabriel Graphen anwenden. Und auf der anderen Seite kann man zuerst den Gabriel Graphen entwickeln und diesen durch den Yao Graphen verfeinern. Diese beiden Teilgraphen haben fast die gleichen Eigenschaften. Beide sind im Ausgangsgrad durch  $k$  beschränkt und weisen einen Power stretch Faktor von  $\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}$  auf. Ein wesentlicher Unterschied zwischen  $GYG_k(G)$  und  $YGG_k(G)$  ist der, dass in  $GYG_k(G)$  die Kantenanzahl durch  $kn$  beschränkt ist und in  $YGG_k(G)$  durch  $3n$ . Des Weiteren ist in [8] bewiesen, dass beide Graphen zusammenhängend bleiben, wenn  $G$  ein zusammenhängender Unit Disk Graph ist.

### 3.2 Yao und Sink Graph

Alle bisher vorgestellten Methoden haben den Nachteil, dass sie keinen beschränkten Eingangsgrad garantieren. Li, Wan und Wang präsentierten in [7] eine Methode, mit der man einen Graphen erzeugen kann; mit sowohl einem beschränkten Power stretch Faktor als auch einem beschränkten Eingangs- und Ausgangsgrad. Zuerst wird dabei die  $\overrightarrow{YG}_k(G)$  Struktur berechnet. Diese kann aber immernoch einen hohen Eingangsgrad haben und deswegen werden aus jedem Kegel des Yao Graphen alle Kanten, die  $u$  als Endknoten haben, zu einem gerichteten Baum  $T(u)$  mit Senke  $u$  zusammengefasst. Dies ist in Abbildung 6 illustriert. Die Vereinigung über alle Bäume  $T(u)$  nennt man die Sink Struktur und der ermittelte Teilgraph wird mit  $\overrightarrow{YG}_k^*(G)$  bezeichnet. Die Ergebnisse aus [8] und [7] sind in Tabelle 1 zusammengefasst.

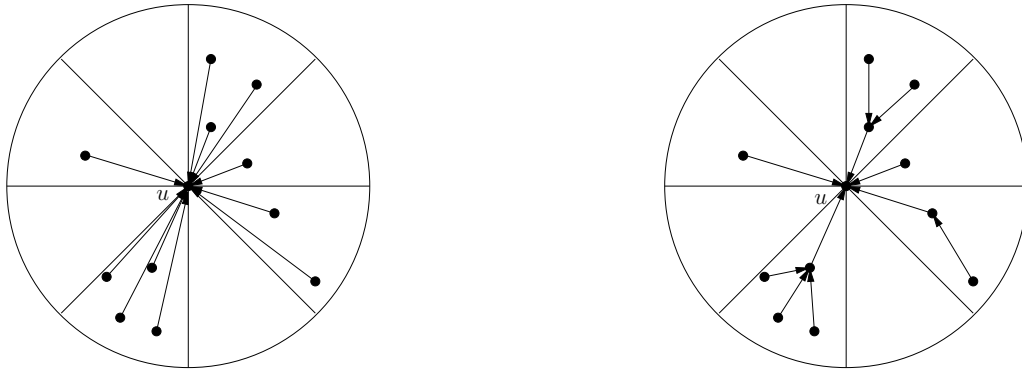


Abbildung 6: links: Alle gerichteten Kanten im Yao Graphen, die  $u$  als Endknoten haben; rechts: Der gerichtete Baum  $T(u)$  mit Senke  $u$ .

### 3.3 Yao plus entgegengesetzter Yao Graph

Die Technik des Yao und Sink Graphen hat den Nachteil, dass die Konstruktion des Sink Graphen sehr kompliziert ist und kaum Verbesserungen gegenüber der Kombination aus Yao und Gabriel Graphen bietet. Deswegen wurde in [8] eine ganz neue Technik entwickelt. Dafür bekommt jeder Knoten  $v_i \in V$  eine individuelle Kennung  $ID(v_i) = i$ . Und die Kennung jeder gerichteten Kante  $\overrightarrow{uv} \in G$  wird definiert als  $ID(\overrightarrow{uv}) = (\|uv\|, ID(u), ID(v))$ . Die Kanten werden mit Hilfe ihrer Kennung geordnet, so dass  $ID(\overrightarrow{uv}) > ID(\overrightarrow{pq})$  gilt, wenn:

1.  $\|uv\| > \|pq\|$  oder
2.  $\|uv\| = \|pq\|$  und  $ID(u) > ID(p)$  oder
3.  $\|uv\| = \|pq\|$ ,  $u = p$  und  $ID(v) > ID(q)$ .

Der Algorithmus berechnet die Struktur des Graphen nun wie folgt: Zuerst wird die  $\overrightarrow{YG}_k(G)$  Struktur berechnet. Danach wählt jeder Knoten  $v$  aus jedem seiner Kegel die

Kante  $\overrightarrow{uv}$  mit der kleinsten  $ID(\overrightarrow{uv})$ . Die anderen, gerichteten Kanten werden verworfen. So entsteht ein Graph mit in  $k$  beschränktem Eingangs- und Ausgangsgrad. Dass dieser Graph zusammenhängend ist, wird in [8] bewiesen. Die entstehende Struktur wird mit  $\overrightarrow{YY}_k(G)$  bezeichnet.

### 3.4 Ergebnisse

Die Ergebnisse aller hier vorgestellten Methoden wurden in [8] verglichen. Dabei wurden zwei Testfelder aufgebaut mit 100 Knoten und 1000 Kanten. Für alle Techniken, die die Yao Technik verwenden, wurde  $k = 8$  gesetzt. Außerdem wurde zur Berechnung der Kommunikationsenergie  $\beta$  auf 2 gesetzt. Es zeigte sich, dass die  $EMST(G)$  und  $RNG(G)$  Strukturen einen zu geringen durchschnittlichen Knotengrad erzeugen und daher zu Ausfallsintoleranz neigen. Die  $YGG_k(G)$  Struktur ist fast identisch zu der  $GG(G)$  Struktur und hat den gleichen durchschnittlichen Knotengrad. Jedoch hat die  $YYG_k(G)$  Struktur ein etwas besseres Broadcast-Energieverhalten, wohingegen die  $GG(G)$  Struktur ein besseres Unicast-Energieverhalten aufweist. Es hat sich auch herausgestellt, dass die  $YYG_k(G)$ ,  $GYG_k(G)$ ,  $YG_k^*(G)$  und  $YY_k(G)$  Strukturen, die auf der  $YG_k(G)$  Struktur basieren, einen besseren durchschnittlichen Knotengrad als diese aufweist und zudem ein besseres Unicast- und Broadcast-Energieverhalten haben. Man kann folgende Ordnung beim Unicastenergieverhalten erkennen, wobei diese stark von der Ausgangsstruktur abhängt:

$$UDG(G) = GG(G) < YG_k(G) = YGG_k(G) = GYG_k(G) < YG_k^*(G) < YY_k(G) < RNG(G) < EMST(G)$$

Und folgende Ordnung beim Broadcastenergieverhalten:

$$EMST(G) < RNG(G) < YGG_k(G) < GG(G) < GYG_k(G) < YY_k(G) < YG_k^*(G) < YG_k < UDG(G)$$

In Abbildung 7 sind alle Versuchsergebnisse nochmal graphisch Dargestellt.

Beim durchschnittlichen Knotengrad lässt sich keine Ordnung erkennen. Jedoch liegen  $GYG_k(G)$ ,  $YGG_k(G)$ ,  $YG_k^*(G)$  und  $YY_k(G)$  nah beieinander, wobei  $YY_k(G)$  und  $YG_k^*(G)$  normalerweise einen geringeren maximalen Knotengrad besitzen, da diese Strukturen einen in  $k$  beschränkten Eingangsgrad haben und deswegen gegenüber den anderen Techniken zu bevorzugen sind.

## 4 Ausblick

Die in [8] vorgestellten Techniken erzeugen stets eine bessere Struktur als die einfachen Techniken  $GG(G)$  und  $YK_k(G)$ . Denn alle in Kapitel 3 vorgestellten Techniken haben sowohl einen beschränkten Power stretch Faktor, als auch einen beschränkten Ausgangsgrad.  $YY_k^*(G)$  und  $YY_k(G)$  garantieren sogar einen beschränkten Eingangsgrad, wobei hier  $YY_k(G)$  stets zu bevorzugen ist, denn diese Technik ist einfacher zu berechnen und liefert

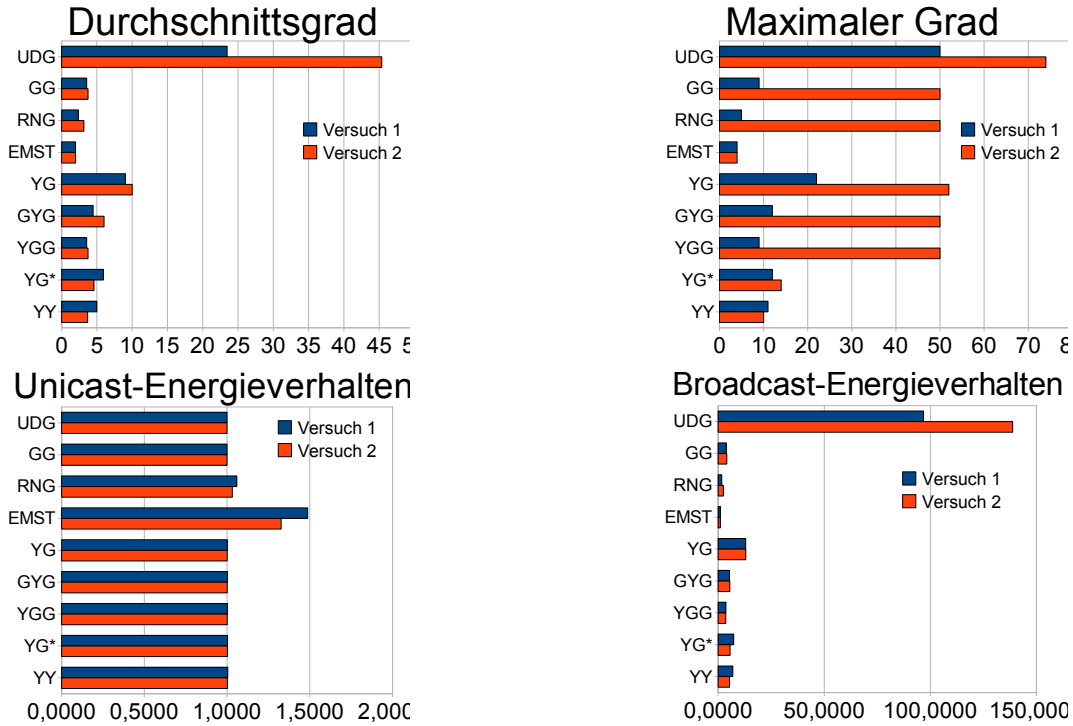


Abbildung 7: Zusammenfassung der Versuchsergebnisse

auch insgesamt ein besseres Energieverhalten. Die  $YY_k(G)$  Struktur ist auch gegenüber den  $YYG_k(G)$  und  $GYG_k(G)$  Strukturen zu bevorzugen, da diese Struktur garantiert, dass bei keinem Knoten ein zu großer Overhead entsteht, denn dies führt besonders in kabellosen Netzwerken zu erheblichen Energieverlusten.

Allerdings ist mir in dieser Arbeit aufgefallen, dass bei der Berechnung der Kommunikationsenergie lediglich der Abstand und das Übertragungsmedium eine Rolle spielen. Es ist aber mittlerweile so, dass kabellose Knoten meist auf mehreren Channels senden und empfangen können. In der hier vorgestellten Kommunikationsenergie-Berechnung wird aber angenommen, dass alle Knoten den selben Channel benutzen. Dies ist weder energieeffizient noch realistisch, denn wenn alle Knoten den selben Channel benutzen, führt das zu unnötigen Interferenzen und folglich enormem Energieverlust. Es wurde in [3] gezeigt, dass ein dünnes Netzwerk Interferenzen nicht reduziert, deswegen muss meines Erachtens dies in der Kommunikationsenergie-Berechnung mit einkalkuliert werden. Des Weiteren wurde auch schon in [8] erwähnt, dass das Speichern und Aufrufen der berechneten Struktur nicht mitberücksichtigt wurde. Die Kommunikationsenergieformel müsste also eigentlich so lauten:  $\|uv\|^\beta + c$ . Würde man Interferenzen aufgrund gleicher Channel noch mitberücksichtigen wollen, müsste die Formel so lauten:  $\|uv\|^{\beta+\gamma} + c$ , wobei  $\gamma$  abhängig von Interferenzwahrscheinlichkeit zwischen den Knoten  $u$  und  $v$  ist.

#### 4 Ausblick

Ein anderer Punkt ist mir ebenfalls negativ aufgefallen. Alle vorgestellten Techniken operieren auf festen kabellosen Knoten, dabei ist der Vorteil eines kabellosen Netzwerkes gerade seine Mobilität. Nur für Knoten die keinen festen Stromzugang haben, ist ein gutes Energieverhalten von Vorteil. Feste Knoten, wie z.B. Router, haben aber meist diesen Stromzugang, und für diese Netzwerke wäre eine Verbesserung des Throughput die bessere Methode, um Energie zu sparen. Diese Techniken entfalten ihre Potenziale erst bei Knoten, die über keine ständige Stromversorgung verfügen. Ein gutes Beispiel ist hierfür das mobile Kommunikations-Netzwerk. Meines Erachtens sind die vorgestellten Techniken und die aus ihnen resultierenden Strukturen in der Lage, bei solchen Netzwerken ein verbessertes Energieverhalten zu erzielen. Doch diese Strukturen müssten in festen Abständen aktualisiert werden. Je nachdem wie man diesen Abstand wählt, kann es zu einem verbesserten Energieverhalten des gesamten Netzwerkes führen. Dieses müsste diese jedoch mit Messdaten belegen.

## Abkürzungsverzeichnis

$UDG(G)$  Unit Disk Graph  
 $RNG(G)$  Relative Neighborhood Graph  
 $EMST(G)$  Euklidischer minimaler Spannbaum  
 $GG(G)$  Gabriel Graph  
 $YG_k(G)$  Yao Graph  
 $GYG_k(G)$  Erst Yao dann Gabriel Graph  $GG(YG_k(G))$   
 $YGG_k(G)$  Erst Gabriel dann Yao Graph  $YG_k(GG(G))$   
 $YY_k(G)$  Yao plus entgegengesetzter Yao graph  
 $YG_k^*(G)$  Yao mit Sink Graphen  
 $\rho_H(G)$  Power stretch Faktor oder Unicast-Energieverhalten  
 $\lambda(G)$  Length stretch Faktor

## Literatur

- [1] J.L. Bentley H.N. Gabow and R.E. Tarjan. Scaling and related techniques for geometry problems. *ACM Symposium on Theory of Computing*, 1984.
- [2] J. Katajainen. The region approach for computing relative neighborhood graphs in the  $l_p$  metric. *Computing* 40, 1988.
- [3] Roger Wattenhofer und Aaron Zollinger Martin Burkhart, Pascal von Rickenbach. Does topology control reduce interference? *Department of Computer Science ETH Zürich*, 2004.
- [4] I. Strojmenovic P. Bose, P. Morin and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *3rd Int. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, 1999.
- [5] W. Evans P. Bose, L. Devroye and J. Kirk-Parick. On the spanning ratio of gabriel graphs and  $\beta$ -skeletons. *Submitted to SIAM Journal on Discrete Mathematics*, 2001.
- [6] J.M. Keil und C.A. Gutwin. Classes of graphs which approximate the complete euclidean graph. *Discrete Computational Geometry* 7, 1992.
- [7] P.-J. Wan X.-Y. Li and Y. Wang. Power efficient and sparse spanner for wireless ad hoc networks. *IEEE IC3N*, 2001.
- [8] Yu Wang Xiang-Yang Li, Peng-Jun Wan and Ophir Frieder. Sparse power efficient topology for wireless networks. *Journal of Parallel and Distributed Computing*, 2002.
- [9] A. C.-C. Yao. On constructing minimum spanning trees in  $k$ -dimensional spaces and related problems. *SIAM J. Computing* 11, 1982.